MCDAN: a Multi-scale Context-enhanced Dynamic Attention Network for Diffusion Prediction

Xiaowen Wang, Lanjun Wang*, Yuting Su, Yongdong Zhang, An-An Liu* *

Abstract

Information diffusion prediction aims at predicting the target users in the information diffusion path on social networks. Prior works mainly focus on the observed structure or sequence of cascades, trying to predict to whom this cascade will be infected passively. In this study, we argue that user intent understanding is also a key part of information diffusion prediction. We thereby propose a novel Multi-scale Context-enhanced Dynamic Attention Network (MCDAN) to predict which user will most likely join the observed current cascades. Specifically, to consider the global interactive relationship among users, we take full advantage of user friendships and global cascading relationships, which are extracted from the social network and historical cascades, respectively. To refine the model's ability to understand the user's preference for the current cascade, we propose a multi-scale sequential hypergraph attention module to capture the dynamic preference of users at different time scales. Moreover, we design a contextual attention enhancement module to strengthen the interaction of user representations within the current cascade. Finally, to engage the user's own susceptibility, we construct a susceptibility label for each user based on user susceptibility analysis and use the rank of this label for auxiliary prediction. We conduct experiments over four widely used datasets and show that MCDAN significantly overperforms the state-of-the-art models. The average improvements are up to 10.61% in terms of Hits@100 and 9.71% in terms of MAP@100, respectively.

Keywords: User intention understanding, information diffusion prediction, context interaction, graph neural networks

*Xiaowen Wang, Lanjun Wang, Yuting Su, and An-An Liu are with the Tianjin University, Tianjin 300072, China. Xiaowen Wang and An-An Liu are also with the Institute of Artificial Intelligence, Hefei Comprehensive National Science Center, Hefei 230088, China. Yongdong Zhang is with the University of Science and Technology of China, Hefei 230026, China. (Corresponding authors: 1. Lanjun Wang, E-mail: wang.lanjun@outlook.com; 2. An-An Liu, E-mail: anan0422@gmail.com).

1 Introduction

Information diffusion prediction, which is also known as cascade prediction, is a challenging but critical task in many real-world application domains, such as influence maximization [1], influential user detection [2], online advertising [3], recommendation [4–7] and rumor detection [8]. Recent works on diffusion prediction rely on the achievements of deep neural networks, jointly learning the structure of the social graph and dynamic diffusion graph through Graph Neural Networks (GNN) [9–11].
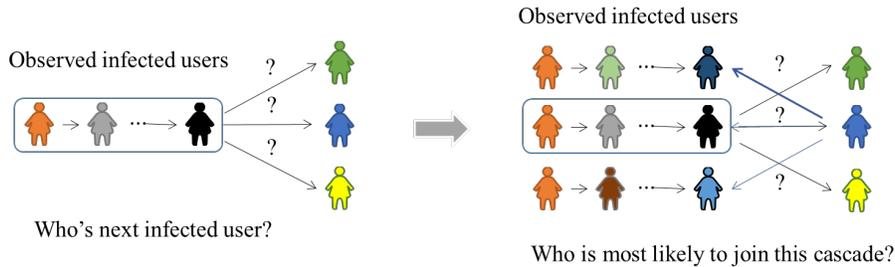

Figure 1: Illustrative examples for traditional next infected user prediction (left) and user-aware next infected user prediction (right).

However, as shown in Fig. 1(left), existing works always focus on predicting to whom this cascade will be infected [9]. In this context, there is no difference between information diffusion and virus transmission, where users are always passively infected. This is not consistent with the fact that information diffusion is a two-way selection process. Fig. 1(right) shows a realistic scenario on social networks, in which a user receives a lot of information every day, but he/she only selects a few of them to forward and/or to reply. That is to say, from the user's perspective, when the cascade appears within the user's visible range, the user makes a decision on whether to join or not based on their own intention for this cascade. As a result, instead of predicting who is the next infected user, the task is more appropriately expressed as 'who is most likely to join this cascade'. Apart from unidirectional information infection, we argue that user intent understanding is also a key part of information diffusion prediction.

The challenges of understanding user intention in information diffusion prediction lie in accurately portraying the target user portrait, which consists of user interaction, cascade memory, and user susceptibility. First, user interaction refers to interactive relationships between the target user and others. Since information diffusion is built on social networks, most of the existing methods take advantage of the social connections [9–11]. Although the who-follow-who relations can represent friendship, it is not enough or exact to represent global dependencies. Second, cascade memory refers to the representation of the current cascade. Previous works capture users' dynamic preference through time series slicing and look up the memory for each cascade [10,11]. Regrettably, these models suffer from information loss due to their reliance on a single time scale. Meanwhile, the context interaction within the cascade also needs to be captured, which cannot be learned only by looking up the stored memory. Third, user susceptibility refers to the personal attribute of the target user regarding whether he/she is susceptible to infection. Although some works consider the social role [12] or social influence [13], they are not straightforward to describe user susceptibility. To the

best of our knowledge, no previous works consider the concept of user susceptibility.

To address these challenges, we propose a Multi-scale Context-enhanced Dynamic Attention Network (MCDAN) to achieve the user-aware next infected user prediction. First, to more accurately represent the global dependencies of users, we do not only take advantage of their friendships from the social network but also consider global cascading relationships, which come from historical cascades. The intention behind this is that people are more likely to interact with friends as well as people who have interacted with them before. Second, to refine the model's ability to understand the user's preferences for the current cascade, we propose a multi-scale sequential hypergraph attention module to capture the dynamic preference of users at different time scales and design a contextual attention enhancement module to strengthen the interaction of user representations within the current cascade. Third, to present the user's own susceptibility, we construct a susceptibility label for each user based on user susceptibility analysis and use the rank of this label for the final auxiliary prediction.

In summary, the main contributions are as follows:

- We propose a user-aware framework named MCDAN which aims at predicting the target user in the information diffusion path by understanding the user's intention. Through this framework, we fully utilize user portraits composed of user interaction, cascade memory, and user susceptibility.

- We build a global cascading graph based on historical cascades, which patches the friendship graph and learns the global user interaction.

- We propose a multi-scale sequential hypergraph attention module to encode the multi-scale cascade memory.

- To enhance the context dependency of the current cascade, a contextual attention enhancement module is also proposed after memory look-up.

- We assign susceptibility labels to users based on historical cascades. The labels ranking helps lock the target user.

- Extensive experiments demonstrate the superiority of our method. We outperform state-of-the-art baseline methods on four public datasets with average improvements of 10.61% in Hits@100 and 9.71% in MAP@100, respectively. In addition, we conduct ablation studies to demonstrate the effectiveness of each part and parameter analysis experiments to discuss the sensitivity of different key parameters.

The rest of this paper is organized as follows. In Section 2, we briefly review the related works including traditional models and deep learning models for information diffusion prediction. In Section 3, we define the problem and introduce the proposed MCDAN model. In Section 4, we report all the results of comparative experiments, ablation study experiments, and parameter analysis experiments. Finally, we summarize the paper in Section 5.

2 Related Work

2.1 Information Diffusion Prediction

Information diffusion prediction is to predict the trajectories as well as the participants in information spreading in the future based on observed cascades and relevant known information. So far, there are many methods for modeling and predicting information

3

cascading and data types related to cascading [10, 11, 14–20]. Here we categorize them into two categories: traditional models and deep learning models.

Traditionally, earlier works mainly focus on feature-based models [14–16] and generative models [21–23]. In feature-based models, different features can be extracted from given information through feature engineering, and classical machine learning methods are used for prediction [24, 25]. Most features are manually constructed, focusing on extracting features from information content [26, 27]. In generative models, the spreading of information is widely characterized by probabilistic statistical generative approaches such as epidemic models [23] and stochastic point processes [21, 22]. However, feature-based models are not generalizable to different scenarios [28], which are inefficient in large-scale networks. Although generative models are easily applicable to modeling information diffusion, they mainly help process modeling but are less powerful in making accurate predictions.

With advances in deep neural networks, prior works utilize or expand related networks to learn potential information [10, 11, 17–20], which can be classified into cascades diffusion based models and social graph based models. For the cascades diffusion based models, typically, DeepDiffuse combines the LSTM network and attention mechanism to learn time information [18]. SNIDSA incorporates a structure attention module and gating mechanism into a recurrent neural network (RNN) for integrating the structural and sequential information [19]. For the social graph based models, they are with an intuition behind that people have common interests with their friends. FOREST introduces social relationships through GNN [9]. Inf-VAE embeds social homophily into the prediction model [20]. With a deeper understanding of information diffusion, DyHGCN jointly learns the structure of the social graph and dynamic diffusion graph [10]. MS-HGAT further introduces diffusion hypergraphs into user representation learning, among which designs a sequential hypergraph attention network to learn user preference dynamically [11]. However, existing works neglect the complete portrayal of user portraits, resulting in information loss in the global dependency of users, the integrity of information, and the user's own susceptibility.

2.2 User Portrait Modeling

User portrait is a concept first proposed by Alan Cooper, and it is interpreted as a concrete representation of the target user [29]. The modeling of user portraits aims to reflect user intention through data analysis, including behaviour patterns and interest preference [30].

Researches on user portrait mainly focus on three directions [31], which are user attribute annotation [32], user preference understanding [33–35], and user behavior analysis [36]. For user attribute annotation, existing works collect some feature information through social annotation systems [32]. User portrait is then built on the extracted features [37, 38]. For user preference understanding, existing models include user preference prediction [34] and similar user mining [4, 39]. The user preference prediction model builds user portraits based on historical information in dynamic social networks [34]. The similar user mining model searches for similar users by mining the same user habits from mobile devices [39]. For user behaviour analysis, the existing methods learn from historical behaviours, establish user behaviour profiles, and analyze potential relationships among users [38, 40, 41]. The methods are mainly applied in fields such as marketing [42] and recommendation [43].

In our study, we are the first to apply the concept of user portrait to information diffusion prediction. The purpose of user portrait modeling is to understand user

intention in information diffusion scenarios, with key elements being user susceptibility, information portrayal, and the relationships between users.
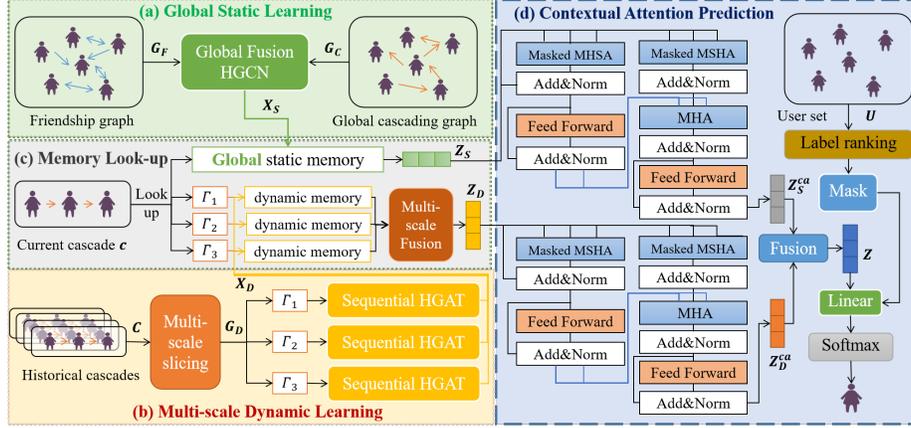
3   Method

Figure 2: An overview of the proposed MCDAN framework. It comprises four components: (a) Global Static Learning, which aims at learning the global users' dependencies; (b) Multi-scale Dynamic Learning, which aims at storing the dynamic users' preference memory based on hypergraphs over M different time scales(we set $M = 3$ for example in Fig. 2); (c) Memory Look-up, which aims at depicting the representations for the observed current cascade and (d) Contextual Attention Prediction, which aims at encoding the cascade with a CAE module and predicting the next user based on user susceptibility analysis.

3.1   Problem Formulation

Since information diffusion prediction aims at predicting the future diffusion process based on the current cascades and relevant knowledge [11], the task is defined as follows. Suppose that a collection of diffusion cascades C is propagated among a set of users U. The user set is recorded as $U = \{u_1, u_2, ..., u_N\}$, where N denotes the maximum user number. The collection of diffusion cascades is recorded as $C = \{c_1, c_2, c_3, ..., c_H\}$, where H denotes the maximum historical cascade number. Given a friendship graph $G_F = (U, E_F)$ where E_F means the friendship edges and an observed current cascade $c = \{(u_i, t_i)|u_i \in U\}$ where t_i means the time u_i join the current cascade. The target is to estimate the probability $\hat{y}$ of each user from the U joining this cascade c at the next step and predict the candidate by ranking all the probabilities.

3.2   Framework

In this study, we apply the social network as a friendship graph G_F and historical cascades C to construct a global cascading graph G_C. Meanwhile, we construct diffusion

hypergraphs G_D which are based on different time scales.

The overall framework of the proposed MCDAN is depicted in Fig. 2. As shown, the model consists of four modules, namely Global Static Learning, Multi-scale Dynamic Learning, Memory Look-up and Contextual Attention Prediction.

(a) Global Static Learning module aims at learning the user interaction and storing the global static memory X_S from the friendship graph G_F and the global cascading graph G_C.

(b) Multi-scale Dynamic Learning module aims at constructing multi-scale hypergraphs G_D and storing the dynamic users' preference memory X_D based on the hypergraphs.

(c) Memory look-up module is used to depict the representations for the current cascade from static memory X_S and dynamic memory X_D, which learned from Global Static Learning module and Multi-scale Dynamic Learning module, respectively. The output of this module is two embeddings named global static user embedding Z_S and multi-scale dynamic user embedding Z_D.

(d) Contextual Attention Prediction module then encodes global static user embedding Z_S and multi-scale dynamic user embedding Z_D with a Contextual Attention Enhancement(CAE) module and applies an insusceptibility mask to obtain the final output.

We then introduce each module in detail in the following sections.

3.3 Global Static Learning

Since the social network represents the friendship and cascading graph from historical cascades represents the historical interaction, we learn the global users' dependencies from the related prior knowledge. Specifically, the global cascading graph is recorded as $G_C = (U, E_C)$, where E_C means the historical cascading edges. Given the friendship graph G_F and global cascading graph G_C, we feed them into the Global Fusion Heterogeneous Graph Convolutional Networks(HGCN) as shown in Fig. 3.

Specifically, we use two multi-layer graph convolutional networks (GCN) [44] to learn user friendship representation and user cascading representation from the two graphs, respectively. The layer-wise propagation rule can be defined as follows:

$$X_F(l+1) = ReLU(\widetilde{D}_F^{-\frac{1}{2}} \widetilde{A}_F \widetilde{D}_F^{-\frac{1}{2}} X_F(l) W_F) \qquad (1)$$

where $X_F(0) \in \mathbb{R}^{N \times d}$ is randomly initialized user friendship embedding with normal distribution, $\widetilde{A}_F = A_F + I$ is built based on the adjacency matrix A_F of the given graph G_F, $\widetilde{D}_F$ is the corresponding degree matrix, and W_F is a trainable weight matrix.

We can obtain the user cascading representation X_C through GCN in a similar layer-wise propagation rule as follows:

$$X_C(l+1) = ReLU(\widetilde{D}_C^{-\frac{1}{2}} \widetilde{A}_C \widetilde{D}_C^{-\frac{1}{2}} X_C(l) W_C) \qquad (2)$$

where $X_C(0) \in \mathbb{R}^{N \times d}$ is randomly initialized user cascading embedding with normal distribution, $\widetilde{A}_C = A_C + I$ is built based on the adjacency matrix A_C of G_C, $\widetilde{D}_C$ denotes degree matrix, and W_C is a trainable weight matrix.

Finally, we fuse user friendship representation X_F and user cascading representation X_C to obtain the final global static memory X_S. The gated fusion module is depicted as follows:

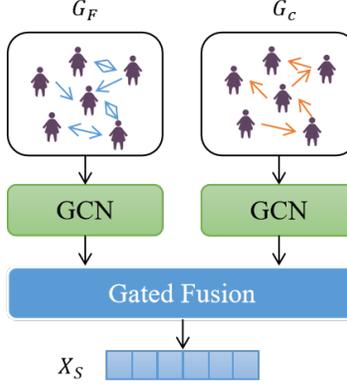$$X_S = \alpha X_F + (1 - \alpha) X_C \qquad (3)$$

Figure 3: Illustration of the proposed Global Fusion HGCN. The friendship graph and global cascading graph are fed into GCN to obtain user friendship representation and user cascading representation, respectively. Then, the representations are fused to obtain the final global static user representation.

$$\alpha = \frac{exp(W_S^T \sigma(W_1 X_F))}{exp(W_S^T \sigma(W_1 X_F)) + exp(W_S^T \sigma(W_1 X_C))} \tag{4}$$

where $X_S \in \mathbb{R}^{N \times d}$, $\sigma(\cdot)$ represents the $tanh$ activation function, W_1 denotes the transformation matrix and W_S is the vector of attention which both W_1 and W_S are trainable.

3.4 Multi-scale Dynamic Learning

Although the static memory describes the users' dependencies globally, it is not enough to reflect the temporal user interaction relationships. As shown in Fig. 2(b), we further construct hypergraphs based on all the historical cascades over different time scales. Then, we apply the Sequential Hypergraph Attention Network(HGAT) module to learn dynamic user interactions from the hypergraphs.

3.4.1 Multi-scale slicing

We construct the diffusion hypergraph set G_D based on historical cascades C over M different time scales. In each time scale, we arrange the cascades in chronological order and split the historical diffusion timeline into Γ time intervals. The multi-scale hypergraphs are constructed as follows:

$$\begin{aligned} G_D &= \{G_D^\Gamma | \Gamma = \Gamma_1, \Gamma_2, ..., \Gamma_M\} \\ G_D^\Gamma &= \{G_D^\tau = (U^\tau, E_D^\tau) | \tau = 1, 2, 3, ..., \Gamma\} \end{aligned} \tag{5}$$

where M denotes the number of time scales, Γ denotes number of time intervals which can be set as $\{\Gamma_1, \Gamma_2, ..., \Gamma_M\}$ and the total timeline is cut into Γ pieces of time intervals, G_D^τ denotes the diffusion hypergraph of the τ-th time interval, U^τ and E_D^τ denote the corresponding users and hyperedges of the users, respectively.

Here we go through this process with a brief example. Suppose that M is 2 and corresponding Γ are $\{4, 8\}$, we have 100 cascades with 8 minutes. Firstly, for $\Gamma = 4$, we divide the cascades into 4 subsets with 2 minutes each in chronological order. Here the length of the time interval is 2 minutes. Note that since there are 100 cascades in total, each subset contains 100 subsequences at most. Secondly, we construct the hypergraph based on the subsequences of each time interval. In each hypergraph, once the users of the same cascade appear in the same time interval, a hyperedge is built. Also note that since there are 100 subsequences at most, the maximum number of hyperedges in one hypergraph is 100. Thirdly, we repeat the above steps with $\Gamma = 8$, in which the number of subsets is 8 and the length of the time interval is 1 minute. Finally, we obtain a single-scale set of 4 hypergraphs and a single-scale set of 8 hypergraphs corresponding to $M = 2$ and $\Gamma = \{4, 8\}$.

3.4.2 Sequential HGAT

Given the diffusion hypergraph set G_D with a size M, we apply M Sequential HGAT modules to learn M single-scale dynamic user memories. The final output of the modules is depicted as follows:

$$
\begin{aligned}
X_D &= \{X_D^\Gamma | \Gamma = \Gamma_1, \Gamma_2, ..., \Gamma_M\} \\
X_D^\Gamma &= Sequential\ HGAT(G_D^\Gamma)
\end{aligned}
\tag{6}
$$

where X_D^Γ denotes the single-scale dynamic memory learned from a single-scale set of diffusion hypergraphs G_D^Γ through a Sequential HGAT module.

Specifically, a Sequential HGAT module consists of Γ blocks. Each block corresponds to a hypergraph for each time interval. In each block, as shown in Fig. 4, we
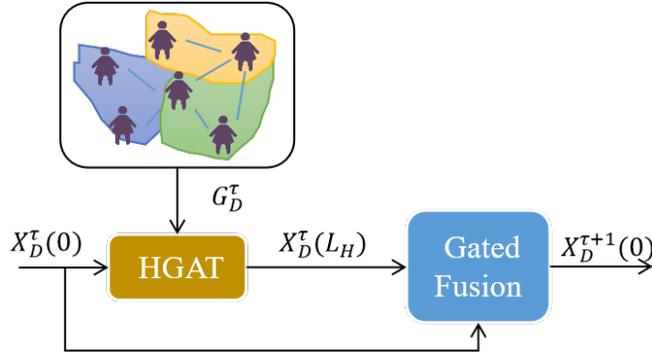

Figure 4: Illustration of a Sequential HGAT block. In each block, an L_H-layer HGAT module and a gated fusion module are applied.

first apply an L_H-layer HGAT to model the high-order interaction among users from a hypergraph for each time interval, the process can be formulated as follows:

$$
X_D^\tau(L_H) = HGAT(X_D^\tau(0), G_D^\tau) \quad \tau = 1, 2, 3, .., \Gamma
\tag{7}
$$

where G_D^τ denotes the corresponding diffusion hypergraph in the time interval τ, $X_D^\tau(0)$ denotes the initial user representation, and the output $X_D^\tau(L_H)$ is stored in single-scale memory X_D^Γ described in Equation(6).

Then, as shown in Fig. 4, we connect $X_D^\tau(0)$ and $X_D^\tau(L_H)$ with a gated fusion module in chronological order to obtain the initial user representation of the next time interval as follows:

$$X_D^{\tau+1}(0) = g_f X_D^\tau(0) + (1 - g_f) X_D^\tau(L_H) \tag{8}$$

$$g_f = \frac{exp(W_{DH}^T \sigma(W_g X_D^\tau(0)))}{exp(W_{DH}^T \sigma(W_g X_D^\tau(0))) + exp(W_{DH}^T \sigma(W_g X_D^\tau(L_H)))} \tag{9}$$

where W_g denotes the transformation matrix and W_{DH} is the vector of attention. Note that we use the friendship representation X_F stored in global static memory as the initial user representation of HGAT in the first time interval to warm up the Sequential HGAT module, i.e. $X_D^1(0) = X_F$.

Furthermore, we illustrate the process of HGAT in detail. Given the diffusion hypergraph G_D^τ, in each L_H-layer HGAT, we have two steps to obtain the high-order interaction among users. That is, (i) nodes to single hyperedge and (ii) hyperedges to nodes. Fig. 5 shows the learning process of a single hyperedge e_j^τ in the diffusion hypergraph G_D^τ.
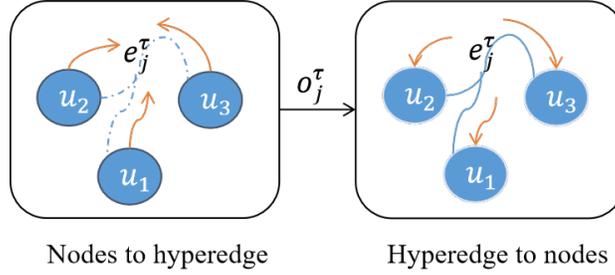

Figure 5: Illustration of an L_H-layer HGAT block in the hyperedge e_j^τ of G_D^τ.

(i) Nodes to single hyperedge: The first step of HGAT aims to learn the representation o_j^τ of hyperedge e_j^τ by aggregating the initial user representation x_i^τ of all the connected nodes u_i^τ:

$$o_j^\tau(l+1) = ReLU(\sum_{u_i^\tau \in e_j^\tau} W_{h_1} x_i^\tau(l)) \tag{10}$$

where $o_j^\tau(l+1)$ denotes the learned representation of the hyperedge e_j^τ, W_{h_1} denotes the trainable parameter, and $x_i^\tau(l)$ denotes the initial user representation.

(ii) Hyperedges to nodes: The second step is based on the learned representations of hyperedges, we integrate all the hyperedges $E_{D,i}^\tau$ joined by u_i^τ in the time interval τ to update the user representation x_i^τ:

$$x_i^\tau(l+1) = ReLU(\sum_{e_j^\tau \in E_{D,i}^\tau} W_{h_2} o_j^\tau(l+1)) \tag{11}$$

where $x_i^\tau(l+1)$ denotes the updated user representation, and W_{h_2} denotes the trainable parameter. Note that all the final updated user representations i.e. $x_i^\tau(L_H)$ constitute the output $X_D^\tau(L_H)$ described in Equation(7).

3.5 Memory Look-up

In this section, we depict the current cascade c with the global static memory X_S obtained from Equation(3) and multi-scale dynamic memory X_D obtained from Equation(6).

As shown in Fig. 2(c), for the static memory, we arrange it with the order of user in the given cascade $c = \{(u_i, t_i)|u_i \in U\}$, thus obtaining global static user embedding: $Z_S = \{x_S^i|u_i \in U, \ x_S^i \in X_S\}$, where $Z_S \in \mathbb{R}^{|c| \times d}$.

For the dynamic memory, we first obtain each single-scale dynamic user embedding and then fuse them with a multi-scale fusion module.

To obtain the single-scale dynamic user embedding from each single-scale dynamic memory X_D^Γ, we query the corresponding users' representation at the nearest time interval based on the given cascade $c = \{(u_i, t_i)|u_i \in U\}$ as well. Since t_i denotes the time that u_i join and if it is divided into the time interval τ, the representation can be represented as: $Z_D^\Gamma = \{x_D^{i,\tau}|u_i \in U, t_i \in \tau, \ x_D^{i,\tau} \in X_D^\Gamma\}$, where $Z_D^\Gamma \in \mathbb{R}^{|c| \times d}$.

After obtaining M single-scale dynamic user embeddings, we propose a multi-scale fusion module to obtain the final multi-scale dynamic user embedding $Z_D \in \mathbb{R}^{|c| \times d}$:

$$
\begin{aligned}
Z_D &= \sum_{\Gamma \in \{\Gamma_1, \Gamma_2, ..., \Gamma_M\}} m_\Gamma Z_D^\Gamma \\
m_\Gamma &= \frac{exp(W_D^T \sigma(W_m Z_D^\Gamma))}{\sum_\Gamma exp(W_D^T \sigma(W_m Z_D^\Gamma))}
\end{aligned}
\tag{12}
$$

where W_m denotes the transformation matrix and W_D is the vector of attention.

3.6 Contextual Attention Prediction

In order to further capture the context information within the current cascade, as shown in Fig. 2(d), we propose a Contextual Attention Enhancement(CAE) module to enhance the context dependency. Based on the user susceptibility analysis, we then feed the context-enhanced embeddings to obtain the final predicted user. We illustrate the process in detail in the following sections.

3.6.1 CAE module

Instead of using an RNN [9] or a single masked Multi-Head Self-Attention(MHSA) module [10, 11] to decode the given user embeddings, we design an encode-decode module to learn the context information.

(i) Self-attention encode: Given the static user embedding Z_S, we first apply a masked MHSA module to learn the hidden embeddings h_S:

$$
\begin{aligned}
MaskedAtt(Q, K, V) &= softmax(\frac{QK^T}{\sqrt{d_\Omega}} + \mathbb{M})V, \\
h_\omega &= MaskedAtt(Z_S W_\omega^Q, Z_S W_\omega^K, Z_S W_\omega^V), \\
h_S &= [h_1; h_2; ...; h_\Omega]W^O
\end{aligned}
\tag{13}
$$

where W_ω^Q, W_ω^K, W_ω^V and W^O are trainable parameters, $d_\Omega = d/\Omega$, d is the dimension of the embedding and Ω denotes the number of heads of attention. The mask matrix $\mathbb{M}$ is defined as $\mathbb{M}_{ij} = 0$ if $i \leq j$ otherwise $-\infty$, which is used to avoid label leakage.

Then, we obtain the attentive representation as context embedding Z_S^c through the two layers fully-connected neural network named Feed Forward network:

$$Z_S^c = ReLU(h_S W_{E_1} + b_1)W_{E_2} + b_2 \tag{14}$$

where W_{E_1} and W_{E_2} are trainable matrices, b_1 and b_2 are trainable bias parameters. Note that residual connection is realized by the Add & Norm layer.

(ii) Context-attention decode: The context-attention decoder has three steps:

First, we apply a masked MHSA module to obtain the masked hidden embedding h_S^c, $h_S^c = h_S$ where h_S is from Equation(13).

Second, we feed the masked hidden embedding h_S^c and context embedding Z_S^c into a multi-head attention(MHA) module to obtain the context-enhanced hidden embedding h_S^{ca}. Specifically, we set h_S^c as Q, Z_S^c as the K and V. The process can be formulated as follows:

$$
\begin{aligned}
Att(Q, K, V) &= softmax(\frac{QK^T}{\sqrt{d_\Omega}})V, \\
h_\omega^{ca} &= Att(h_S^c W_\omega^{Q^{ca}}, Z_S^c W_\omega^{K^{ca}}, Z_S^c W_\omega^{V^{ca}}), \\
h_S^{ca} &= [h_1^{ca}; h_2^{ca}; ...; h_\Omega^{ca}]W^{O^{ca}}
\end{aligned}
\tag{15}
$$

where $W_\omega^{Q^{ca}}$, $W_\omega^{K^{ca}}$, $W_\omega^{V^{ca}}$ and $W^{O^{ca}}$ are trainable matrices.

Third, we apply a Feed Forward network to learn the context-enhanced user embedding Z_S^{ca}:

$$Z_S^{ca} = ReLU(h_S^{ca} W_{E_3} + b_3)W_{E_4} + b_4 \tag{16}$$

where W_{E_3} and W_{E_4} are trainable matrices, b_3 and b_4 are trainable bias parameters.

Meanwhile, given the dynamic user embedding Z_D, we obtain the dynamic context-enhanced user embedding Z_D^{ca} in a similar way.

After then, we incorporate the static context-enhanced user embedding Z_S^{ca} and the dynamic context-enhanced user embedding Z_D^{ca} to obtain the final context-enhanced user embedding Z. We implement a new gated fusion module to realize:

$$Z = \beta Z_S^{ca} + (1 - \beta)Z_D^{ca} \tag{17}$$

$$\beta = \frac{exp(W_Z^T \sigma(W_2 Z_S^{ca}))}{exp(W_Z^T \sigma(W_2 Z_S^{ca})) + exp(W_Z^T \sigma(W_2 Z_D^{ca}))} \tag{18}$$

where W_2 denotes the transformation matrix and W_Z is the vector of attention.

3.6.2 Prediction based on user susceptibility

Before the final prediction, we construct a susceptibility label for each user based on all the historical cascades. Specifically, we calculate the frequency of infection for each user based on the given cascades. Then we match the user susceptibility score based on this frequency, thus obtaining the susceptibility label.

Based on the susceptibility label, we sort the user set U with the label ranking. We set an insusceptible threshold to put users who rank behind it as insusceptible users.

Table 1: Statistics of the preprocessed datasets in our experiments

Datasets	Twitter	Douban	Android	Christianity
# Users	12,627	12,232	2,927	1,651
# Fri. Links	309,631	198,496	24,459	21,955
# Cas. Links	73,036	51,797	23,958	11,328
# Cascades	3,442	3,475	678	589
Avg. Length	32.60	21.76	42.05	26.02

Finally, we compute the diffusion probabilities with the mask of the insusceptible users:

$$\hat{y} = softmax(W_p Z + Mask) \tag{19}$$

where W_p denotes the trainable parameter, and $Mask$ is used to mask users who are set as insusceptible.

During the training, we apply the cross-entropy loss as the objective function:

$$Loss(\theta) = -\sum_{t=2}^{|c|} \sum_{i=1}^{N} y_{ti} log(\hat{y}_{ti}) \tag{20}$$

where θ denotes all the parameters to be trainable. If the user u_i join in cascade c at the step t, $y_{ti} = 1$, otherwise $y_{ti} = 0$.

4 Experiments

In this section, we first illustrate the datasets, implementation details and baseline models used in our experiments, and then we report the results of comparative experiments, ablation study experiments, and parameter analysis experiments to demonstrate the effectiveness of our proposed MCDAN model.

4.1 Experiment Setting

4.1.1 Datasets

Following the previous work [11], we conduct our experiments on four publicly available datasets, i.e., Twitter [45], Douban [46], Android [20] and Christianity [20]. The first two datasets are collected from social media, the last two are collected from Stack-Exchanges.

For a fair comparison, we preprocess the four datasets according to the method in [11] by removing cascades with lengths beyond 200. The statistics of the preprocessed datasets are shown in Table 1. Specifically, $\#Fri.Links$ denotes the number of edges in the friendship graph from the social network. $\#Cas.Links$ denotes the number of edges in the global cascading graph which is built based on the historical cascades. Note that all isolated edges are not counted.

For each dataset, we randomly split the dataset by 8:1:1 for training, validation, and testing.

4.1.2 Implementation Details

We implement the proposed MCDAN via PyTorch [47]. Specifically, we apply Adam as the optimizer, and the learning rate is initialized as 0.01. The batch size of training and the dimension of embeddings are both set to 64.

For global static learning (Sec. 3.3), we apply two two-layer GCNs to learn the friendship graph and global cascading graph, respectively. For multi-scale dynamic learning(Sec. 3.4), the number of time scales M is set to 3 and the corresponding numbers of time intervals are set to $\{4, 8, 16\}$. During each time scale, one-layer HGAT is adopted to learn the high-order interaction from each hypergraph. For contextual attention prediction(Sec. 3.6), the number of heads in multi-head attention Ω is set to 14. The insusceptible label threshold $ratio_t$ is set to adapt to the datasets. Here we set it to 12% for Twitter, 6.9% for Douban, 0.17% for Android and 0.15% for Christianity.

4.1.3 Evaluation Metrics

Following previous works [9–11, 20], the evaluation metrics we use in this study are two ranking metrics, that is, Hits score on top k (Hits@k) and Mean Average Precision on top k(MAP@k), where k = $\{10, 50, 100\}$.

4.2 Baselines

To verify the effectiveness of the proposed MCDAN, we compare it with the following baselines in two categories.

4.2.1 Cascades Diffusion based

These methods infer future diffusion processes based on the given cascades without social information.

- DeepDiffuse [18] combines the LSTM network and attention mechanism to model the diffusion path.
- Topo-LSTM [48] extends standard LSTM to Topo-LSTM to model the cascade diffusion.
- NDM [17] models the cascades by self-attention mechanism and CNNs.
- SNIDSA [19] integrates structural attention modules and gating mechanisms into RNN for model learning.

4.2.2 Social Graph based

These methods utilize social information and given cascades for diffusion prediction.

- FOREST [9] incorporates the social connections through graph neural networks into RNN for prediction.
- Inf-VAE [20] learns social homophily by utilizing graph neural network architectures and integrates it into a variational autoencoder.
- DyHGCN [10] jointly learns the graph representations of the social graph and diffusion graph for dynamic diffusion modeling.
- MS-HGAT [11] proposes a memory-enhanced sequential hypergraph attention network on the basis of the social graph and diffusion hypergraphs.

4.3 Overall performance

Table 2: Overall results with Hits@k scores for k = 10, 50, 100 on four public datasets(%). The average improvement is up to 10.61% in terms of the Hits@100 score.

model	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	33.50	49.59	58.91	21.33	35.25	42.75	10.41	20.31	27.55	28.80	47.14	55.62
MCDAN(ours)	**38.45**	**55.78**	**64.25**	**49.39**	**58.58**	**62.81**	**11.89**	**25.10**	**32.79**	**35.49**	**56.92**	**67.41**

The comparison results over four datasets are reported in Table 2 and Table 3, which correspond to two metrics, Hits@k and MAP@k, respectively. Since MS-HGAT [11] is the SOTA model, all the experimental results of baselines reported are cited from it. From these two tables, we obtain the following observations:

1. Our MCDAN model achieves optimal predictive performance. Compared with the SOTA model MS-HGAT [11], we have achieved better performance on all four datasets with average improvements of 10.61% in the Hits@100 score and 9.71% in the MAP@100 score, respectively. Especially, for the Douban dataset, our MCDAN reaches up to 20.06% improvement in Hits@100 score and 28.59% in MAP@100 score.

2. Regarding the Hits@100 metric shown in Table 2, our method improves 5.34% in Twitter, 20.06% in Douban, 5.24% in Android and 11.79% in Christianity by comparing with the SOTA model, respectively. Since the Hits@100 metric refers to the hitting rate of the first 100 results, the reason for such improvement is our method presents more global interaction and context information at different scales, which helps learn more potential interactions from both historical cascades and the current cascade.

3. Regarding the MAP@100 metric shown in Table 3, our method improves 3.51%

Table 3: Overall results with MAP@k scores for k = 10, 50, 100 on four public datasets(%). The average improvement is 9.71% in terms of the MAP@100 score.

model	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	22.49	23.17	23.30	11.72	12.52	12.60	6.39	6.87	6.96	17.44	18.27	18.40
MCDAN(ours)	**25.89**	**26.69**	**26.81**	**40.70**	**41.13**	**41.19**	**7.47**	**8.04**	**8.15**	**22.88**	**23.78**	**23.94**

Table 4: Ablation study with Hits@k scores for k = 10, 50, 100 on four public datasets(%). Note that we use underlining to mark the results of the most effective component.

model	Twitter			Douban			Android			Christianity		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
MCDAN	**38.45**	**55.78**	**64.25**	**49.39**	**58.58**	**62.81**	**11.89**	**25.10**	**32.79**	**35.49**	**56.92**	**67.41**
w/o G	32.02	49.94	60.27	28.97	42.72	49.11	10.88	22.61	29.99	32.59	51.12	64.96
w/o M	38.16	54.96	63.17	39.65	52.43	57.58	11.58	22.92	30.61	31.47	52.46	62.05
w/o C	35.41	50.49	58.55	20.50	34.86	41.97	11.11	21.45	28.67	31.92	52.46	61.38
w/o L	34.94	51.64	60.58	45.77	55.64	60.05	11.34	21.06	28.44	32.14	52.90	66.29

in Twitter, 28.59% in Douban, 1.19% in Android and 5.54% in Christianity by comparing with the SOTA model, respectively. Since the MAP@100 metric refers to the mean average precision of the first 100 results, the reason for such improvement is our method of user susceptibility analysis as an auxiliary prediction can eliminate more erroneous candidates and improve the precision of retrieval. In addition, the enrichment of the potential interaction also improves prediction precision.

4.4 Ablation study

We conduct ablation studies on the different components of the proposed MCDAN model over the four public datasets. The variants of the model are designed as:

w/o Global cascading graph: remove global cascading graph. The Equation(3) is replaced by:

$$X_S = X_F \tag{21}$$

w/o Multi-scale diffusion hypergraphs: replace the multi-scale diffusion hypergraphs with single-scale diffusion hypergraphs. Here we set the number of time intervals to 8, which is the same as the previous models [10, 11]. The Equation(12) is replaced by:

$$Z_D = Z_D^\Gamma \tag{22}$$

where $\Gamma = 8$.

w/o CAE module: replace the contextual attention enhancement module with only a multi-head self-attention decoder for each user embedding. The Equation(17) is replaced by:

$$Z = \gamma Z_S^c + (1 - \gamma)Z_D^c \tag{23}$$

$$\gamma = \frac{exp(W_\gamma^T \sigma(W_3 Z_S^c))}{exp(W_\gamma^T \sigma(W_3 Z_S^c)) + exp(W_\gamma^T \sigma(W_3 Z_D^c))} \tag{24}$$

where Z_S^c denotes the user static embedding obtained through Equation(14), Z_D^c denotes the user dynamic embedding obtained in a similar way from Z_D, W_3 denotes the transformation matrix and W_γ is the vector of attention.

w/o Label ranking mask: remove the final insusceptible label mask. The Equation(19) is replaced by:

$$\hat{y} = softmax(W_p Z) \tag{25}$$

The ablation results are reported in Table 4 and Table 5 which are related to two metrics, Hits@k and MAP@k. From these two tables, we obtain the following observations:

Table 5: Ablation study with MAP@k scores for k = 10, 50, 100 on four public datasets(%). Note that we use underlining to mark the results of the most effective component.

model	Twitter			Douban			Android			Christianity		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
MCDAN	**25.89**	**26.69**	**26.81**	**40.70**	**41.13**	**41.19**	**7.47**	**8.04**	**8.15**	**22.88**	**23.78**	**23.94**
w/o G	<u>19.90</u>	<u>20.72</u>	<u>20.87</u>	19.68	20.30	20.39	<u>6.63</u>	<u>7.17</u>	<u>7.27</u>	20.34	21.23	21.42
w/o M	25.18	25.96	26.08	29.89	30.50	30.57	6.99	7.50	7.61	20.20	21.14	21.28
w/o C	23.59	24.29	24.40	<u>11.17</u>	<u>11.87</u>	<u>11.97</u>	6.85	7.29	7.39	<u>19.26</u>	<u>20.16</u>	<u>20.28</u>
w/o L	22.73	23.49	23.62	37.36	37.81	37.87	7.09	7.52	7.62	19.64	20.52	20.71

Table 6: Results of the impact of the number of time scales M on four public datasets(%). (Hits@k scores for k = 10, 50, 100)

M	Γ	Twitter			Douban			Android			Christianity		
		@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
1	8	38.16	54.96	63.17	39.65	52.43	57.58	11.58	22.92	30.61	31.47	52.46	62.05
3	**4,8,16**	**38.45**	**55.78**	**64.25**	**49.39**	**58.58**	**62.81**	**11.89**	**25.10**	**32.79**	**35.49**	**56.92**	**67.41**
5	2,4,8,16,32	34.86	52.04	61.25	45.11	54.96	59.88	10.65	22.92	31.24	32.37	50.89	65.63

1. Our MCDAN model achieves the best performance over all the datasets, which confirms that all components of our model are effective.

2. Regarding social media-based datasets i.e. Twitter and Douban, the requirements for different components in different datasets are not completely consistent. As the Hits@k metric shown in Table 4 and the MAP@k metric shown in Table 5, we observe that for the Twitter dataset, the most effective component is the global cascading graph, the least useful one is the multi-scale diffusion hypergraphs. However, for the Douban dataset, the most effective component is the CAE module, and the least powerful one is the masking of insusceptibility labels. The difference shows the different propagation tendencies of different datasets. Specifically, the Twitter dataset exhibits global propagation consistency, while the Douban dataset focuses more on the contextual information of the current sequence.

3. Regarding Stack-Exchanges-based datasets i.e. Android and Christianity, the conclusion is consistent with results on the social media-based datasets. Specifically, we observe that for the Android dataset, the least powerful component is the multi-scale diffusion hypergraphs. While for the Christianity dataset, the least effective one is the masking of insusceptibility labels. However, compared with social media-based datasets, the performance differences of variant models on these two datasets are not significant, which may be limited by the size of the datasets.

4.5 Parameter Analysis

In this part, we further conduct some sensitivity analysis experiments of key parameters on the four datasets to identify how they influence the prediction performance.

4.5.1 Impact of the historical cascades proportion

Since the global cascading graph is constructed based on the historical cascades, which is confirmed as an effective component. Here we further explore how they affect the

Table 7: Results of the impact of the number of time scales M on four public datasets(%). (MAP@k scores for k = 10, 50, 100)

M	Γ	Twitter			Douban			Android			Christianity		
		@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
1	8	25.18	25.96	26.08	29.89	30.50	30.57	6.99	7.50	7.61	20.20	21.14	21.28
3	**4,8,16**	**25.89**	**26.69**	**26.81**	**40.70**	**41.13**	**41.19**	**7.47**	**8.04**	**8.15**	**22.88**	**23.78**	**23.94**
5	2,4,8,16,32	22.90	23.68	23.82	34.79	35.26	35.33	6.88	7.44	7.56	20.27	21.22	21.43

Table 8: insusceptible label threshold setting.

t_{ratio}	Twitter				Douban				Android				Christianity			
	0.02	0.04	0.06	0.08	0.02	0.04	0.06	0.08	0.02	0.04	0.06	0.08	0.02	0.04	0.06	0.08
# insusceptible users	252	505	757	1010	244	489	734	978	58	117	175	234	33	66	99	132

prediction performance when the historical cascades proportion changes across the range of {0, 0.2, 0.4, 0.6, 0.8, 1}, where 0 denotes that no historical cascades are given, while 1 denotes that all the cascades of the training set are given to construct the graph.

The results are reported in Fig. 6. We observe that as the proportion increases, the prediction performance of the model improves. More specifically, for the social media-based datasets i.e. Twitter and Douban, there is a significant increase in performance between 0 and 0.2, and the trend gradually stabilizes afterward. For the Android dataset, the overall performance improvement is relatively slow. A similar trend is also reflected in Hits@k of the Christianity dataset, but another metric MAP@k shows a linear increase after 0.4.

4.5.2 Impact of the number of scales M

In this study, we propose a multi-scale diffusion hypergraphs learning structure and confirm its effectiveness through the ablation study. Here we further discuss how the number of time scales M and the corresponding numbers of time intervals Γ affect the prediction performance. We change the number M across the range of {1, 3, 5}. For the single scale, we set the corresponding number of time intervals Γ to 8. For the three scales we propose in MCDAN, we set the corresponding numbers of time intervals Γ to {4, 8, 16} as in Sec. 4.3. For the five scales, we set the corresponding numbers of time intervals Γ to {2, 4, 8, 16, 32}.

The results in Table 6 and Table 7 show that the method with the three scales we apply in the study ($M = 3$) achieves the best performance. It is also noted that excessive time scales may increase memory burden and may not necessarily result in optimal performance.

4.5.3 Impact of the insusceptible label threshold

Since the insusceptible label threshold may affect the performance, we thus construct experiments to explore the impact. We change the insusceptible label threshold t_{ratio} across the range of {0, 0.02, 0.04, 0.06, 0.08} and record the insusceptible user size of datasets in Table 8.

From Fig. 7, we find that it is not the higher threshold that leads to better performance. For example, for the Douban dataset, compared with 0.08, 0.06 has better

performance. This is because an excessively large threshold may cause some inactive users to lose access to information, leading to prediction failure. From the results of different datasets, we also find that the threshold should not be an invariant constant, which needs to be adapted to the dataset.

5 Conclusion

In this study, we propose a novel Multi-scale Context-enhanced Dynamic Attention Network (MCDAN) for diffusion prediction. Different from the previous studies which select the infected user unidirectionally, our model aims at predicting the target user by understanding the user's intention. We construct user representations based on global user dependency and context enhancement from three perspectives of user portrait, which are users' global relationships, multi-scale dynamic preference, and user susceptibility. Comprehensive experiments constructed on four public datasets demonstrate the superiority of the proposed model.

Acknowledgements

References

[1] S. Kumar, A. Mallik, A. Khetarpal, and B. Panda, "Influence maximization in social networks using graph embedding and graph neural network," *Information Sciences*, vol. 607, pp. 1617–1636, 2022.

[2] E. Oro, C. Pizzuti, N. Procopio, and M. Ruffolo, "Detecting topic authoritative social media users: a multilayer network approach," *IEEE Transactions on Multimedia*, vol. 20, no. 5, pp. 1195–1208, 2017.

[3] H. Wang, Q. Meng, J. Fan, Y. Li, L. Cui, X. Zhao, C. Peng, G. Chen, and X. Du, "Social influence does matter: User action prediction for in-feed advertising," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 34, no. 01, 2020, pp. 246–253.

[4] S. Huang, J. Zhang, L. Wang, and X.-S. Hua, "Social friend recommendation based on multiple network correlation," *IEEE Transactions on Multimedia*, vol. 18, no. 2, pp. 287–299, 2015.

[5] Z. Zhao, Q. Yang, H. Lu, T. Weninger, D. Cai, X. He, and Y. Zhuang, "Social-aware movie recommendation via multimodal network learning," *IEEE Transactions on Multimedia*, vol. 20, no. 2, pp. 430–440, 2017.

[6] L. Sang, M. Xu, S. Qian, M. Martin, P. Li, and X. Wu, "Context-dependent propagating-based video recommendation in multimodal heterogeneous information networks," *IEEE Transactions on Multimedia*, vol. 23, pp. 2019–2032, 2020.

[7] S. Feng, C. Xu, Y. Zuo, G. Chen, F. Lin, and J. XiaHou, "Relation-aware dynamic attributed graph attention network for stocks recommendation," *Pattern Recognition*, vol. 121, p. 108119, 2022.

[8] F. Zeng and W. Gao, "Early rumor detection using neural hawkes process with a new benchmark dataset," in *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, 2022, pp. 4105–4117.

[9] C. Yang, J. Tang, M. Sun, G. Cui, and Z. Liu, "Multi-scale information diffusion prediction with reinforced recurrent networks." in *IJCAI*, 2019, pp. 4033–4039.

[10] C. Yuan, J. Li, W. Zhou, Y. Lu, X. Zhang, and S. Hu, "Dyhgcn: A dynamic heterogeneous graph convolutional network to learn users' dynamic preferences for information diffusion prediction," in *Machine Learning and Knowledge Discovery in Databases: European Conference, ECML PKDD 2020, Ghent, Belgium, September 14–18, 2020, Proceedings, Part III*. Springer, 2021, pp. 347–363.

[11] L. Sun, Y. Rao, X. Zhang, Y. Lan, and S. Yu, "Ms-hgat: Memory-enhanced sequential hypergraph attention network for information diffusion prediction," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 4, 2022, pp. 4156–4164.

[12] Y. Yang, J. Tang, C. Leung, Y. Sun, Q. Chen, J. Li, and Q. Yang, "Rain: Social role-aware information diffusion," in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 29, no. 1, 2015.

[13] D. Li, Z. Xu, Y. Luo, S. Li, A. Gupta, K. Sycara, S. Luo, L. Hu, and H. Chen, "Modeling information diffusion over social networks for temporal dynamic prediction," in *Proceedings of the 22nd ACM international conference on Information & Knowledge Management*, 2013, pp. 1477–1480.

[14] D. Liao, J. Xu, G. Li, W. Huang, W. Liu, and J. Li, "Popularity prediction on online articles with deep fusion of temporal process and content features," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 33, no. 01, 2019, pp. 200–207.

[15] C. Gou, H. Shen, P. Du, D. Wu, Y. Liu, and X. Cheng, "Learning sequential features for cascade outbreak prediction," *Knowledge and Information Systems*, vol. 57, pp. 721–739, 2018.

[16] A. L. Jia, S. Shen, D. Li, and S. Chen, "Predicting the implicit and the explicit video popularity in a user generated content site with enhanced social features," *Computer Networks*, vol. 140, pp. 112–125, 2018.

[17] C. Yang, M. Sun, H. Liu, S. Han, Z. Liu, and H. Luan, "Neural diffusion model for microscopic cascade study," *IEEE Transactions on Knowledge and Data Engineering*, vol. 33, no. 3, pp. 1128–1139, 2019.

[18] M. R. Islam, S. Muthiah, B. Adhikari, B. A. Prakash, and N. Ramakrishnan, "Deepdiffuse: Predicting the'who'and'when'in cascades," in *2018 IEEE international conference on data mining (ICDM)*. IEEE, 2018, pp. 1055–1060.

[19] Z. Wang, C. Chen, and W. Li, "A sequential neural information diffusion model with structure attention," in *Proceedings of the 27th ACM international conference on information and knowledge management*, 2018, pp. 1795–1798.

[20] A. Sankar, X. Zhang, A. Krishnan, and J. Han, "Inf-vae: A variational autoencoder framework to integrate homophily and influence in diffusion prediction," in *Proceedings of the 13th international conference on web search and data mining*, 2020, pp. 510–518.

[21] Y. Wang, X. Ye, H. Zhou, H. Zha, and L. Song, "Linking micro event history to macro prediction in point process models," in *Artificial Intelligence and Statistics*. PMLR, 2017, pp. 1375–1384.

[22] B. Samanta, A. De, A. Chakraborty, and N. Ganguly, "Lmpp: A large margin point process combining reinforcement and competition for modeling hashtag popularity." in *IJCAI*, 2017, pp. 2679–2685.

[23] L. Zhao, J. Chen, F. Chen, F. Jin, W. Wang, C.-T. Lu, and N. Ramakrishnan, "Online flu epidemiological deep modeling on disease contact network," *GeoInformatica*, vol. 24, pp. 443–475, 2020.

[24] X. Lai, Y. Zhang, and W. Zhang, "Hyfea: winning solution to social media popularity prediction for multimedia grand challenge 2020," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 4565–4569.

[25] K. Wang, P. Wang, X. Chen, Q. Huang, Z. Mao, and Y. Zhang, "A feature generalization framework for social media popularity prediction," in *Proceedings of the 28th ACM International Conference on Multimedia*, 2020, pp. 4570–4574.

[26] W. Chen, C. Huang, W. Yuan, X. Chen, W. Hu, X. Zhang, and Y. Zhang, "and-tag contrastive vision-and-language transformer for social media popularity prediction," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 7008–7012.

[27] J. Wu, L. Zhao, D. Li, C.-W. Xie, S. Sun, and Y. Zheng, "Deeply exploit visual and language information for social media popularity prediction," in *Proceedings of the 30th ACM International Conference on Multimedia*, 2022, pp. 7045–7049.

[28] F. Zhou, X. Xu, G. Trajcevski, and K. Zhang, "A survey of information cascade analysis: Models, predictions, and recent advances," *ACM Computing Surveys (CSUR)*, vol. 54, no. 2, pp. 1–36, 2021.

[29] A. Cooper, *The inmates are running the asylum.* Springer, 1999.

[30] G. Li, W. Chen, X. Yan, and L. Wang, "Modeling and analysis of group user portrait through wechat mini program," *Wireless Communications and Mobile Computing*, vol. 2022, 2022.

[31] Y. Chen, J. He, W. Wei, N. Zhu, and C. Yu, "A multi-model approach for user portrait," *Future Internet*, vol. 13, no. 6, p. 147, 2021.

[32] H. Zeng and S. Wu, "User image and precision marketing on account of big data in weibo," *Modern Economic Information*, vol. 16, pp. 306–308, 2016.

[33] L. Xing, Z. Song, and Q. Ma, "User interest model based on hybrid behaviors interest rate," *Application Research of Computers*, vol. 33, no. 3, pp. 661–665, 2016.

[34] L. Wu, Y. Ge, Q. Liu, E. Chen, B. Long, and Z. Huang, "Modeling users' preferences and social links in social networking services: a joint-evolving perspective," in *Proceedings of the AAAI conference on artificial intelligence*, vol. 30, no. 1, 2016.

[35] Q. Gao, D. Fu, and X. Dong, "A context-aware mobile user behavior-based neighbor finding approach for preference profile construction," *Sensors*, vol. 16, no. 2, p. 143, 2016.

[36] J. Li, S. Pan, L. Huang *et al.*, "A machine learning based method for customer behavior prediction," *Tehnički vjesnik*, vol. 26, no. 6, pp. 1670–1676, 2019.

20

[37] J. Mueller and G. Stumme, "Gender inference using statistical name characteristics in twitter," in *Proceedings of the The 3rd Multidisciplinary International Social Networks Conference on SocialInformatics 2016, Data Science 2016*, 2016, pp. 1–8.

[38] H. Gu, J. Wang, Z. Wang, B. Zhuang, and F. Su, "Modeling of user portrait through social media," in *2018 IEEE international conference on multimedia and expo (ICME)*. IEEE, 2018, pp. 1–6.

[39] H. Ma, H. Cao, Q. Yang, E. Chen, and J. Tian, "A habit mining approach for discovering similar mobile users," in *Proceedings of the 21st international conference on World Wide Web*, 2012, pp. 231–240.

[40] F. Li, N. Clarke, M. Papadaki, and P. Dowland, "Behaviour profiling on mobile devices," in *2010 International conference on emerging security technologies*. IEEE, 2010, pp. 77–82.

[41] G. Zhao, X. Qian, and X. Xie, "User-service rating prediction by exploring social users' rating behaviors," *IEEE Transactions on Multimedia*, vol. 18, no. 3, pp. 496–506, 2016.

[42] J. R. Saura, "Using data sciences in digital marketing: Framework, methods, and performance metrics," *Journal of Innovation & Knowledge*, vol. 6, no. 2, pp. 92–102, 2021.

[43] F. Sun, J. Liu, J. Wu, C. Pei, X. Lin, W. Ou, and P. Jiang, "Bert4rec: Sequential recommendation with bidirectional encoder representations from transformer," in *Proceedings of the 28th ACM international conference on information and knowledge management*, 2019, pp. 1441–1450.

[44] T. N. Kipf and M. Welling, "Semi-supervised classification with graph convolutional networks," *arXiv preprint arXiv:1609.02907*, 2016.

[45] N. O. Hodas and K. Lerman, "The simple rules of social contagion," *Scientific reports*, vol. 4, no. 1, p. 4343, 2014.

[46] E. Zhong, W. Fan, J. Wang, L. Xiao, and Y. Li, "Comsoc: adaptive transfer of user behaviors over composite social network," in *Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining*, 2012, pp. 696–704.

[47] N. Ketkar and J. Moolayil, *Introduction to PyTorch*. Berkeley, CA: Apress, 2021, pp. 27–91. [Online]. Available: https://doi.org/10.1007/978-1-4842-5364-9_2

[48] J. Wang, V. W. Zheng, Z. Liu, and K. C.-C. Chang, "Topological recurrent neural network for diffusion prediction," in *2017 IEEE international conference on data mining (ICDM)*. IEEE, 2017, pp. 475–484.
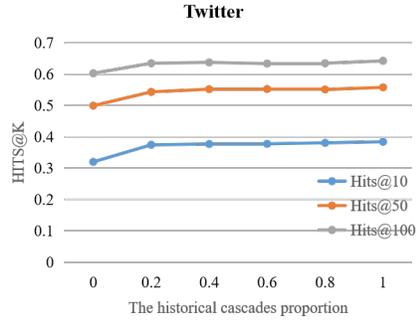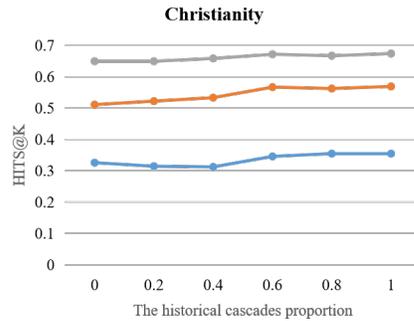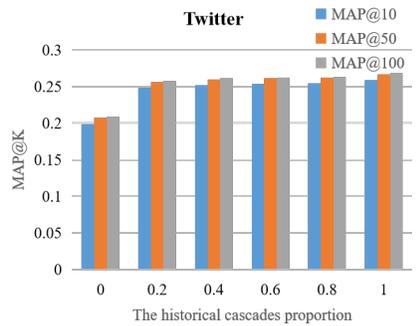
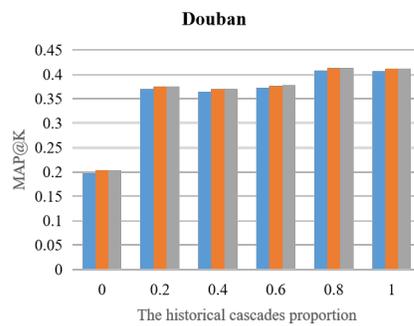(a) Hits@k on Twitter
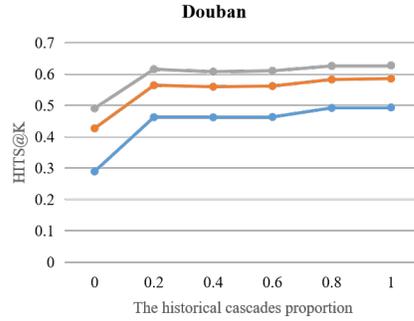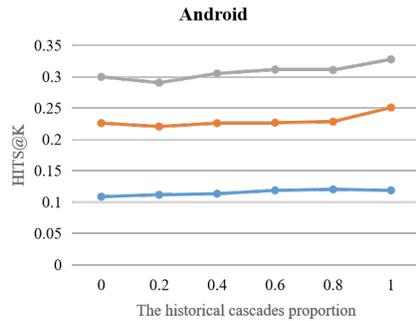
(b) Hits@k on Douban
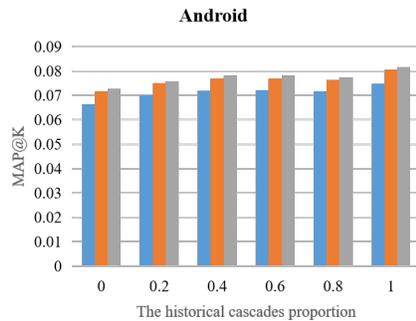
(c) Hits@k on Android
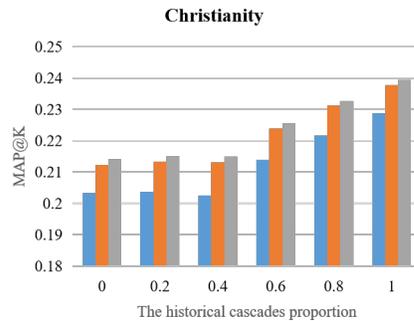
(d) Hits@k on Christianity

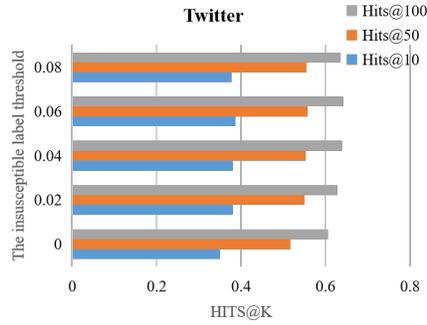(e) MAP@k on Twitter

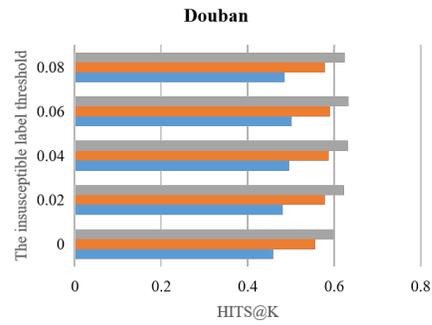(f) MAP@k on Douban

(g) MAP@k on Android
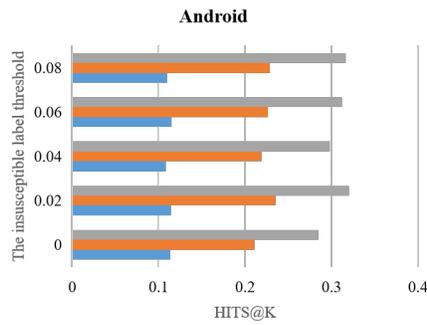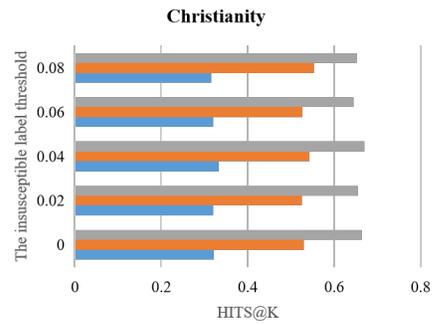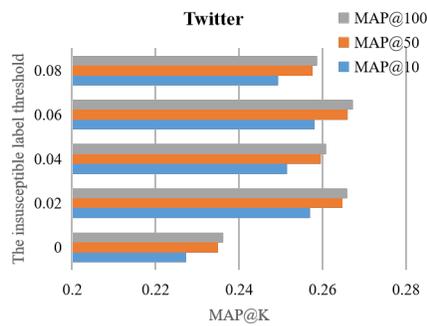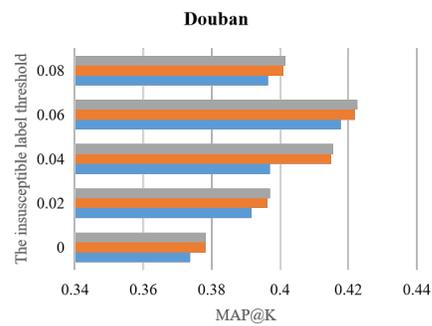
(h) MAP@k on Christianity

Figure 6: Results of the impact of the historical cascades proportion on the four public datasets.

(a) Hits@k on Twitter
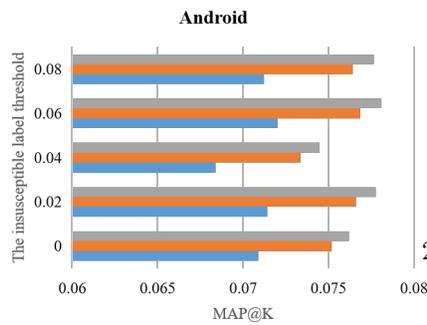
(b) Hits@k on Douban

(c) Hits@k on Android
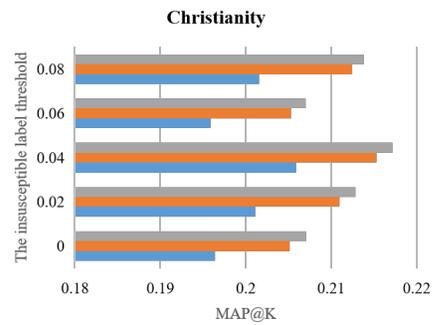
(d) Hits@k on Christianity

(e) MAP@k on Twitter

(f) MAP@k on Douban

(g) MAP@k on Android

(h) MAP@k on Christianity

Figure 7: Results of the impact of the insusceptible label threshold on the four public datasets.

23