Information Diffusion Prediction with Temporal and Structural Contrastive Learning Augmented Graph Neural Network

Anonymous ACL submission

Abstract

Information diffusion prediction aims to forecast how information propagates through social networks. Current works have explored users' preferences from dynamic diffusion structures and social relations. Despite recent advances, they generally share two natural deficiencies. First, they generally fail to identify users' critical preferences hidden in noisy and complex user structures. In addition, existing works primarily extract users' dynamic preferences within localized sub-graph structures, struggling to filter relevant preferences for the current cascade. Thus, we propose **T**emporal and **S**tructural **C**ontrastive **L**earning **A**ugmented Graph Neural Network (**TSCLA**). Specifically, we split the diffusion process into discrete periods and introduce a temporal contrastive learning module to extract users' diversified preferences across the diffusion process. Furthermore, we introduce a hierarchical adaptation module that dynamically filters relevant preferences in each diffusion period. In addition, we construct a heterogeneous graph to extend users' preferences and design a structural contrastive learning module for discerning critical user relations from noisy connections. Experimental results on four real-world datasets demonstrate the superior performance of our model compared to state-of-the-art baselines.

1 Introduction

In recent years, online social networks (OSN) have become powerful platforms for information dissemination. These networks trigger large-scale online information diffusion cascades, facilitating the rapid dissemination of online content. To understand the patterns of information diffusion and identify potential participants, researchers have formulated the task of information diffusion prediction. As a fundamental problem in social networks analysis, the task has been applied in many downstream social applications, such as fake news detection (Wei et al., 2021; Yuan et al., 2019), and personalized recommendation (Wang et al., 2020; Liu et al., 2023b).

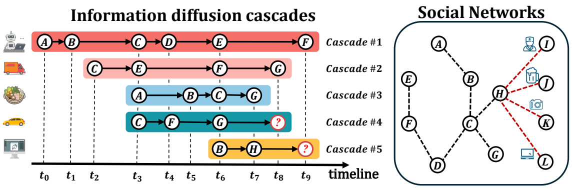


Figure 1: A toy example of diffusion cascades and social network. The diffusion process is split into three diffusion periods $[t_0, t_3)$, $[t_3, t_6)$, and $[t_6, t_9)$. Icons near social relations reflect users' personal preferences.

Generally, existing works on information diffusion prediction are encapsulated into the following categories. 1) *Feature engineering-based models* (Yang et al., 2015; Bourigault et al., 2016) assume that the diffusion process is governed by an underlying diffusion function. However, they hardly generalize to real-world diffusion cascades. 2) *Sequential-based models* (Islam et al., 2018; Yang et al., 2021; Wang et al., 2018a) employ sequence models to extract user correlations within the diffusion path. Despite their effectiveness, these methods focus solely on sequential patterns, overlooking user interactions beyond sequential structures. 3) *Graph-based models* (Wang et al., 2018b, 2022) believe that users' preferences drive the diffusion process and introduce different graph structures to learn users' preferences. Recently, studies (Yuan et al., 2020; Sun et al., 2022; Jiao et al., 2024) further explore the impact of users' dynamic preferences, achieving promising performance.

Although existing works extract various preferences from different user relations or structures, they still suffer from two key limitations. Firstly, existing research typically incorporates all social relations to extend user preferences without distinguishing critical relationships hidden in complex social structures. Users in social networks maintain numerous social connections, each of which reflects different personal preferences. Therefore, indiscriminately incorporating all of a user's social relationships for prediction may introduce substantial noise, reducing predictive accuracy. As illustrated in Figure 1, when predicting the future participants of Cascade #5 after timestamp t_7, considering all of User H's social relations introduces more noise than useful information, since only User L shares a preference relevant to the current cascade. Secondly, current studies describe users' dynamic preferences from separated sub-graph structures, failing to identify relevant preferences that shape the current cascade from these structure snapshots. User interactions in social platforms follow habitual behavioral patterns, resulting in similar user structures among different snapshots. Inferring user preferences solely from these structures may introduce massive amounts of irrelevant information. For instance, when predicting the diffusion trend of Cascade #4, multiple similar diffusion structures with minor variations can be observed at different periods (*e.g.*, diffusion structure $C \rightarrow D \rightarrow E$, $C \rightarrow E \rightarrow F$, and $C \rightarrow G$), making it challenging for models to discern preferences that are informative for the current cascade.

To address the above problems, we propose a **T**emporal and **S**tructural **C**ontrastive **L**earning **A**ugmented graph neural network (short for **TSCLA**). Specifically, we discretize the diffusion process into periods and apply GCN at each period to learn user preferences at each period. Moreover, we introduce a temporal contrastive learning module to avoid extracting redundant preferences from similar structures and diversify user preferences in the diffusion process. Furthermore, we design a hierarchical adaptation module, which selects relative preferences at each period based on the user context of the current cascade. In addition, we integrate three different user relations to extend users' preferences and design a structural contrastive learning module to filter critical user relations in the heterogeneous graph. We conduct extensive experiments on four real-world datasets. Empirical results demonstrate that our model outperforms state-of-the-art baselines, validating the effectiveness of TSCLA.

In summary, the main contributions of this paper are three-fold:

- We propose a temporal contrastive learning module to extract users' diversified preferences at each period and design a hierarchical adaptation module to select preferences relevant to the current cascade.
- We incorporate three types of user relations to extend users' preferences and design a structural contrastive learning module for discerning critical user relations within noisy structures.
- We conduct extensive experiments on four real-world datasets, demonstrating the effectiveness of TSCLA in information diffusion prediction.

2 Related Work

2.1 Information Diffusion Prediction

Information diffusion prediction has attracted significant research interest for decades. Early feature-engineering models assume the diffusion process adheres to specific diffusion models (Kempe et al., 2003). However, their assumptions constrain their ability to characterize complex diffusion patterns.

Sequential-based models have emerged as a promising approach with the recent advancement of deep learning. They (Wang et al., 2017; Islam et al., 2018; Yang et al., 2021) transform diffusion cascades into user sequences and incorporate various sequential models (Hochreiter and Schmidhuber, 1997; Vaswani et al., 2017) to capture user correlations within the sequence. However, these sequential-based models focus on strictly ordered user sequences and overlook users' non-sequential correlations.

To address this problem, various graph structures, primarily social graphs, have been utilized to extend user associations. Researchers (Wang et al., 2018b, 2022; Cheng et al., 2023) further found that users' dynamic preferences play a vital role in facilitating information diffusion and extracted various user preferences from graph structures. Most recently, some studies (Wang et al., 2021; Yuan et al., 2020) consider that users' preferences evolve as time passes; they model the diffusion process as a series of structure snapshots and employ graph neural networks to learn users' dynamic preferences.

However, current works incorporate all social relations for prediction without distinguishing critical user relations from noisy connections. Moreover, they only describe users' preferences from separated sub-graph structures, failing to identify relevant preferences within each diffusion period.

2.2 Graph Contrastive Learning

Graph neural networks (GNNs) have demonstrated strong capabilities in learning node and edge representations from graph-structured data (Yao et al., 2019; Velickovic et al., 2018). However, traditional graph methods rely heavily on labeled data. Inspired by the success of self-supervised learning in various domains (Zhou et al., 2021; Jain et al., 2021), recent studies have introduced graph contrastive learning (GCL) to mitigate dependence on labeled data and improve generalization performance (Wu et al., 2023; Liu et al., 2023a).

Existing GCL methods can be broadly categorized into two types based on the contrasting scales, same-scale contrastive learning and cross-scale contrastive learning. Same-scale contrastive learning, where the contrastive objectives are set to discriminate graph views generated on the same scale. Cross-scale contrastive learning generates samples at multiple scales and maximizes the mutual information between these scales. Both strategy primarily based on mutual information (MI) maximization, where model is optimized to maximize agreement between positive pairs while pushing negative pairs apart.

Inspired by their outstanding performance, we leverage both contrastive learning frameworks in our model to assist prediction.

3 Problem Formulation

The information diffusion process is typically recorded as a cascade $c_m = \{(u_i^m, t_i^m)|i < L)\}$ in chronological order, where element (u_i^m, t_i^m) denotes that user u_i^m participates in cascade c_m at time t_i^m, *e.g.*, forwarding the message m at timestamp t_i^m. L is the maximum length of the cascade. The cascade c_m can be further decomposed into user sequence $c_m^u = \{u_i^m|i < L\}$ and timestamp sequence $c_m^t = \{t_i^m|i < L\}$. We collect all historical cascades and users in $\mathcal{C} = \{c_1, c_2, ..., c_{|\mathcal{C}|}\}$ and $\mathcal{U} = \{u_1, u_2, ..., u_{|\mathcal{U}|}\}$, respectively.

Moreover, we introduce three graph structures essential for prediction: social graph $\mathcal{G}_s$, diffusion graph $\mathcal{G}_d$, and bipartite graph $\mathcal{G}_b$, as illustrated on the upper part of Figure 2. The social graph $\mathcal{G}_s = \{\mathcal{U}, \mathcal{E}_s\}$ is an undirected graph that describes the social connections among users. Similarly, the diffusion graph $\mathcal{G}_d = \{\mathcal{U}, \mathcal{E}_d\}$ is a directed graph formed by users' diffusion connections. The bipartite graph $\mathcal{G}_b = \{\mathcal{V}_b, \mathcal{E}_b\}$ is a directed graph describing the co-occurrence correlation between users and cascades.

Based on the above formulations, we define the task of **information diffusion prediction** as follows: given the set of user $\mathcal{U}$, the set of historical cascades $\mathcal{C}$, user graph structures $\mathcal{G}_s, \mathcal{G}_d, \mathcal{G}_b$, and a current cascade $c_o = \{(u_i^o, t_i^o)|u_i^o \in \mathcal{U}\}$. Our objective is to compute the conditional probability $\hat{\boldsymbol{y}}_j = p(u_j|c_o)$ to show how likely user u_j will participate in cascade c_o at the next timestamp.

4 Method

In this section, we present the details of our proposed TSCLA framework. An overview of the architecture is illustrated in Fig. 2.

4.1 Temporal Contrastive Enhanced User Encoding

4.1.1 User Dynamic Embedding

To model the diffusion dynamics at different periods, we first discretize the diffusion graph into a discrete-time dynamic graph. Specifically, we create a set of chronological structural snapshots based on diffusion graph $\mathcal{G}_d$, denoted as $\mathcal{G}_D = \{\mathcal{G}_d^i\}, 0 \le i \le N$. For structure snapshot $\mathcal{G}_d^i = \{\mathcal{U}, \mathcal{E}_d^i\}$, its edge set $\mathcal{E}_d^i$ is constructed by selecting diffusion edges within the corresponding time interval, *i.e.*, $\mathcal{E}_d^i = \{e_d|e_d \in \mathcal{E}_d, t_i \le t_{e_d} < t_{i+1}\}$. Then, we apply an independent GNN for each structure snapshot to extract users' dynamic preferences at periods. Technically, we formulate the process in the i^{th} snapshot $\mathcal{G}_d^i$ as,

$$\mathbf{X}^{d,i(2)} = \mathbf{\Psi}_i^d \left(\mathcal{G}_d^i, \mathbf{X}^{d,i(0)}\right), \qquad (1)$$

where $\mathbf{X}^{d(0)}$ is the input node embedding matrix initialized with a normal distribution (Glorot and Bengio, 2010). $\mathbf{\Psi}_i^d(\cdot)$ is a two layer GNN. $\mathbf{X}^{d,i(2)}$ denotes user embedding matrix from the last GNN layer to represent the users' dynamic preference at the i^{th} period, abbreviated as $\mathbf{X}^{d,i}$.

4.1.2 Temporal Contrastive Enhancement

Although users' dynamic preferences continually evolve across periods, certain behavioral patterns persist. Inferring preferences solely from these structures without considering their temporal context will extract redundant preference. For example, as illustrated in Figure 1, we may extract the same preference for User A in period $[t_0, t_3)$ and $[t_3, t_6)$ due to similar user interaction $A \to B$ in these periods.

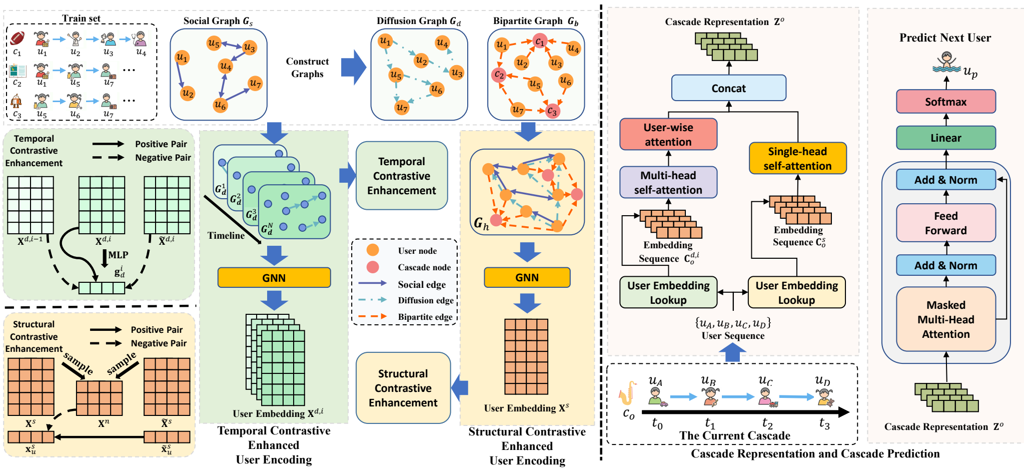


Figure 2: The overview architecture of TSCLA.

To address this, we propose a temporal contrastive learning task for adjacent subgraphs in $\mathcal{G}_D$. As illustrated in the left part of Fig. 2, the core objective of this contrastive framework is to align user embeddings with their corresponding period-specific graph-level structural summaries for each period while distancing them from summaries of adjacent periods, encouraging our model to extract diverse preferences from each snapshot. Specifically, we first employ a readout function to obtain the graph-level representation $\mathbf{g}_d^i$ for each subgraph $\mathcal{G}_d^i$, formulated as, $\mathbf{g}_d^i = \mathbf{MLP}(\frac{1}{|\mathcal{U}|}\sum_{i=1}^{|\mathcal{U}|}\mathbf{X}^{d,i})$, where $\mathbf{MLP}$ is a projection head that maps the aggregated user embeddings to a shared latent space. Then, we define positive and negative samples in our temporal contrastive learning framework. For structure summary $\mathbf{g}_d^i$ at i^{th} period, its positive samples are corresponding user embeddings, *i.e.*, $\mathbf{x}_k^{d,i} \in \mathbf{X}^{d,i}$. Its negative samples consist of two components. The first part is generated by perturbing the input user matrix $\mathbf{X}^{d,i(0)}$ with row-wise and column-wise shuffling and feeding it to the same GNN layer $\Psi_i^d(\cdot)$, producing $\tilde{\mathbf{X}}^{d,i(0)}$ and subsequently $\tilde{\mathbf{X}}^{d,i}$. The second part is the user embeddings from the preceding period, denoted as $\mathbf{x}_k^{d,i-1} \in \mathbf{X}^{d,i-1}$.

Finally, we leverage Jensen-Shannon Divergence (JSD) (Peng et al., 2020) to estimate the distance between positive and negative samples, which is,

$$\mathcal{L}_t^i = 2\sum_k^{|\mathcal{U}|}\mathcal{D}_w\left(\mathbf{g}_d^i, \mathbf{x}_k^{d,i}\right) - E_{\tilde{P}}\left[\left(\mathcal{D}_w\left(\mathbf{g}_d^i, \tilde{\mathbf{x}}_k^{d,i}\right)\right)\right] - E_{\tilde{P}}\left[\left(\mathcal{D}_w\left(\mathbf{g}_d^i, \mathbf{x}_k^{d,i-1}\right)\right)\right], \tag{2}$$

where $\mathcal{D}_w : D \times D' \rightarrow R$ is a discriminator constructed by neural networks. The overall temporal contrastive loss across all snapshots is given by, $\mathcal{L}t = \frac{1}{|N-1|}\sum_{i=1}^{|N-1|}\mathcal{L}_t^i$.

4.2 Structural Contrastive Enhanced User Encoding

4.2.1 User Structure Embedding

Existing studies (Yuan et al., 2020; Sun et al., 2022) have demonstrated that the diffusion process is affected by multiple user preferences. Thus, we further incorporate additional types of user relations to extract users' personal preferences and social biases for the diffusion process to extend users' dynamic preferences and enhance prediction accuracy.

Specifically, we integrate social graph $\mathcal{G}_s$ and bipartite graph $\mathcal{G}_b$ with diffusion graph $\mathcal{G}_d$, forming a unified heterogeneous graph $\mathcal{G}_h = \{\mathcal{V}_h, \mathcal{E}_h\}$. Here, $\mathcal{V}_h = \mathcal{V}_b$ is the set of nodes and $\mathcal{E}_h = \mathcal{E}_s \cup \mathcal{E}_d \cup \mathcal{E}_b$ is the edge set. As heterogeneous graph $\mathcal{G}_h$ reflects users' personal preferences and social biases from different perspectives, we apply two layers of GNN on the heterogeneous graph to learn users' preferences,

$$\mathbf{X}^{s(2)} = \mathbf{\Psi}^s\left(\mathcal{G}_h, \mathbf{X}^{s(0)}\right), \tag{3}$$

where $\mathbf{X}^{s(0)}$ is a randomly initialized input embedding matrix. $\mathbf{\Psi}^s(\cdot)$ is a two layer GNN. Since the embeddings are derived by attending to multiple relational structures, we refer to them user structure embeddings. Additionally, as cascade node embeddings hardly contribute to prediction, we retain only the user embedding from $\mathbf{X}^{s(2)}$, forming the final structural embedding matrix embedding matrix $\mathbf{X}^s$.

4.2.2 Structural Contrastive Enhancement

To extract critical user relations from such a complex structure, we introduce a structural contrastive learning module. As illustrated in the left part of Fig. 2, the key objective is to align the embeddings of the same user across different views while separating them from other users' embeddings to identify invariant critical user relations in the graph.

Specifically, , we first generate an augmented view for user structure embedding $\mathbf{X}_s$. Similar to temporal contrastive learning, we apply random shuffling to the input matrix $\mathbf{X}^{s(0)}$ and feed it to GNN $\mathbf{\Psi}^s(\cdot)$, to obtain augmented user embedding matrix $\tilde{\mathbf{X}}^s$. We then define the positive and negative samples. For a given user u_j, the embedding $\mathbf{x}^s_{u_j}$ and $\tilde{\mathbf{x}}^s_{u_j}$ from two views are naturally considered as positive sample pairs with each other. Their negative samples consist of $2K$ node embeddings, denoted as $\tilde{\mathbf{x}}^n_{u_k} \in \tilde{\mathbf{X}}^n$, which are selected randomly and evenly from both views. To optimize the task, we adopt the InfoNCE (Zhu et al., 2020) objective function, which is formulated as,

$$\mathcal{L}_s^i = \frac{1}{2K}\sum_{k=1}^{2K}\log\frac{exp\left(\mathbf{x}^s_{u_j}\cdot\tilde{\mathbf{x}}^s_{u_j}/\tau\right)}{\sum_{k=1}^{2K} exp\left(\mathbf{x}^s_{u_j}\cdot\tilde{\mathbf{x}}^n_{u_k}/\tau\right)} \tag{4}$$

where τ is a temperature parameter. We apply this operation on all nodes in the heterogeneous graph and aggregate the loss from each node. The final objective is defined as, $\mathcal{L}_s = 1/(|\mathcal{U}| + |\mathcal{C}|)\sum_{i=1}^{|\mathcal{U}|+|\mathcal{C}|}\mathcal{L}_s^i$, where $|\mathcal{U}|$ and $|\mathcal{C}|$ represent the size of user nodes and cascade nodes, respectively.

4.3 Cascade Representation and Prediction

To predict the current cascade c_o, we first construct its contextual representations by integrating users' dynamic embeddings and structural embeddings and then attend to these contextual representations to make predictions.

4.3.1 Cascade Representation

Although users' dynamic embeddings capture diverse user preferences, they still contain a massive amount of irrelevant information for predicting the current cascade. To filter useful information from these embeddings, we design a hierarchical adaptation module to represent the user context in the current cascade with user dynamic embedding, consisting of a multi-head attention layer and a user-wise attention layer. In detail, given cascade c_o and its user sequence c_o^u, we first construct user embedding sequences at each diffusion period by looking up the corresponding user embedding matrix. For the i^{th} period, the embedding sequence is represented as $\mathbf{C}_o^{d,i} = \left[\mathbf{x}^{d,i}_{u_1}, \mathbf{x}^{d,i}_{u_2}, \ldots, \mathbf{x}^{d,i}_{u_L}\right]$.

Previous works (Wang and Li, 2019) have shown that user correlations in the current cascade are reflected in their dependency on previous users. We employ a multi-head self-attention module to capture dependency contexts in the current cascade and extract helpful user preferences within each diffusion period. In practice, we consider each diffusion period as an attention head for parallelization and apply a mask matrix to avoid future information leakage. The process can be formulated as,

$$\begin{aligned}
&\mathbf{Att}(\mathbf{Q},\mathbf{K},\mathbf{V}) = \text{softmax}\left(\frac{\mathbf{Q}\mathbf{K}^T}{\sqrt{d_h}} + \mathbf{M}\right)\mathbf{V},\\
&\mathbf{C}_o^{h,i} = \mathbf{Att}\left(\mathbf{C}_o^{d,i}\mathbf{W}_i^Q, \mathbf{C}_o^{d,i}\mathbf{W}_i^K, \mathbf{C}_o^{d,i}\mathbf{W}_i^V\right),\\
&\mathbf{C}_o^{r,d} = [\mathbf{C}_o^{h,1}; \ldots; \mathbf{C}_o^{h,N}]\mathbf{W}^A,
\end{aligned} \tag{5}$$

where $\mathbf{W}_i^Q, \mathbf{W}_i^K, \mathbf{W}_i^V, \mathbf{W}_i^A$ are learnable matrices. d_h is the scaling factor. $\mathbf{M}$ is a matrix to mask out future users to avoid label leakage, which is $\mathbf{M}_{i,j} = -\infty$ if $i \geq j$ else $\mathbf{M}_{i,j} = 0$. We refer to $\mathbf{C}_o^{r,d}$ as an overall refined user representation, which encodes users' relative preferences within the cascade at each period.

Then, we perform user-wise attention to fuse refined user representations across different diffusion periods. Given user u_j in the current cascade, we first obtain his refined representations $\left[\mathbf{x}^{r,1}_{u_j}, \ldots, \mathbf{x}^{r,N}_{u_j}\right]$ from the overall refined representation $\mathbf{C}_o^{r,d}$ and stack them as matrix $\mathbf{X}^r_{u_j}$. We then apply user-wise attention to integrate users' relative preferences from different periods. The process is,

$$\alpha^i = \frac{\exp\left(\boldsymbol{a}^\top\mathbf{W}_a\mathbf{x}^{r,i}_{u_j}\right)}{\sum_i^N \exp\left(\boldsymbol{a}^{top}\mathbf{W}_a\mathbf{x}^{t,k}_{u_j}\right)}, \quad \mathbf{x}^d_{u_j} = \sum_i^N \alpha^i\mathbf{X}^r_{u_j}, \tag{6}$$

where $\boldsymbol{a}$ and $\mathbf{W}_a$ are trainable parameter. Similarly, we also perform single head attention for structural user embedding sequence $\mathbf{C}_o^s = \left[\mathbf{x}^s_{u_1}, \mathbf{x}^s_{u_2}, \ldots, \mathbf{x}^s_{u_L}\right]$ and concatenate its output user representation with the above user-wise representations to generate users' contextual representation, which is,

$$\begin{aligned}
&\mathbf{C}_o^{r,s} = \mathbf{Att}\left(\mathbf{C}_o^s\mathbf{W}_i^{s,Q}, \mathbf{C}_o^s\mathbf{W}_i^{s,K}, \mathbf{C}_o^s\mathbf{W}_i^{s,V}\right),\\
&\mathbf{z}_o^{u_i} = [\mathbf{x}_o^{r,s,u_i}; \mathbf{x}_o^{r,d,u_i}]
\end{aligned} \tag{7}$$

where $\mathbf{W}_i^{s,Q}, \mathbf{W}_i^{s,K}, \mathbf{W}_i^{s,V}$ are learnable matrices. $[;]$ means concatenation operation. $\mathbf{x}_o^{r,s,u_i} \in \mathbf{C}_o^{r,s}$. Thus, the final contextual representation of current cascade c_o is given by $\mathbf{Z}_o = [\mathbf{z}_o^{u_1}, \mathbf{z}_o^{u_2}, \ldots, \mathbf{z}_o^{u_L}]$.

4.3.2 Cascade Prediction

Given the user representation matrix $\mathbf{Z}_o$ for the current cascade c_o, we then employ the multi-head self-attention mechanism (as defined in Eq. 5) to attend to contextual correlations within the cascade, facilitating more effective prediction. The process is formulated as follows,

$$\mathbf{Z}_o^{h,i} = \mathbf{Att}\left(\mathbf{Z}_o\mathbf{W}_i^{o,Q}, \mathbf{Z}_o\mathbf{W}_i^{o,K}, \mathbf{Z}_o\mathbf{W}_i^{o,V}\right), \quad \mathbf{Z}_o^h = [\mathbf{Z}_o^{h,1}; \dots; \mathbf{Z}_o^{h,H}]\mathbf{W}^{o,A}, \tag{8}$$

where $\mathbf{W}_i^{o,Q}, \mathbf{W}_i^{o,K}, \mathbf{W}_i^{o,V}, \mathbf{W}_i^{o,A}$ are learnable matrices. Then, we apply two layers of fully connected neural networks to further refine user representations, given by,

$$\mathbf{Z}_p^o = \sigma(\mathbf{Z}_h^o\mathbf{W}_1 + \mathbf{b}_1)\mathbf{W}_2 + \mathbf{b}_2, \tag{9}$$

where $\mathbf{W}_1$, $\mathbf{W}_2, \mathbf{b}_1, \mathbf{b}_2$ are learnable parameters.

Finally, we compute the infection probability $\hat{\boldsymbol{y}} \in R^{L\times|\mathcal{U}|}$, which represents the likelihood of each user participating in the cascade at each timestamp, formulated as,

$$\hat{\boldsymbol{y}} = \mathbf{softmax}(\mathbf{W}_p\mathbf{Z}_p^o + \mathbf{Mask}), \tag{10}$$

where $\mathbf{W}_p$ is a learnable parameter to calculate the infect probability. $\mathbf{Mask}$ matrix is to mask users who have already participated in the current cascade.

For model optimization, we apply cross-entropy loss for the task of information diffusion prediction, which is defined as,

$$\mathcal{L}_p = -\sum_{i=2}^{|c_o|}\sum_{j=1}^{|\mathcal{U}|} \boldsymbol{y}_{ij}\log(\hat{\boldsymbol{y}}_{ij}), \tag{11}$$

where $\boldsymbol{y}_{ij} = 1$ denotes that the predicted user u_j is infected at position i; otherwise, $\boldsymbol{y}_{ij} = 0$. The overall objective function is as follows,

$$\mathcal{L}(\theta) = \mathcal{L}_p + \gamma_t\mathcal{L}_t + \gamma_s\mathcal{L}_s, \tag{12}$$

where γ_t and γ_s are hyper-parameters balancing the relative importance of two auxiliary tasks. θ denotes the set of all learnable parameters.

5 Experiment

5.1 Experimental Setups

5.1.1 Datasets

To evaluate the performance of our model, we employ four publicly available real-world datasets from previous works (Sun et al., 2022; Yuan et al., 2020), which are, 1. **Twitter** (Hodas and Lerman, 2014) consists of tweets containing URLs posted on Twitter during October 2010, along with their diffusion paths and a predefined social network. 2. **Douban** (Sun et al., 2022) records book-sharing behaviors on the Douban website. The co-occurrence connection of users is interpreted as their social relation. 3. **Android** (Sankar et al., 2020) consists of users' interactions with the topic "Android" from a community Q&A website and social relations among users. 4. **Meme** (Leskovec et al., 2009) tracks the migration of frequent quotes and phrases (*i.e.*, memes) across different websites. The detailed statistics of these datasets are presented in Table 3.

5.1.2 Baselines

To evaluate the performance of our model, we compare it with the following seven state-of-the-art information diffusion prediction models, 1. **NDM** (Yang et al., 2021) utilizes the self-attention mechanism and CNN modules to capture long-term user correlation in cascade sequences. 2. **FOREST** (Yang et al., 2019) is a recurrent model with a GRU to learn sequential features and a GCN to extract social structure information. 3. **DyHGCN** (Yuan et al., 2020) applies GCN to learn users' dynamic preferences by discretizing the diffusion process into heterogeneous subgraphs. 4. **MS-HGAT** (Sun et al., 2022) constructs a series of hyper-graphs to model user interactions and integrate them with social relations to depict user interaction dependencies. 5. **MINDS** (Jiao et al., 2024) models diffusion process as hyper-graph snapshots and incorporates acroscopic size prediction to assist prediction. 6. **DisenIDP** (Cheng et al., 2023) leverages two hyper GCN to learn users' intents and designs a self-supervised disentanglement task to assist the procedure. 7. **RotDiff** (Qiao et al., 2023) maps the users into the hyperbolic representation space based on the social relations and diffusion paths, which achieves state-of-the-art performances.

5.1.3 Parameter Settings

We implement TSCLA with PyTorch and optimize it by Adam. The learning rate is set to 0.001. The batch size for training is 16, and the dimensionality of all embeddings and representations is fixed at $d = 64$. To construct the dynamic graph $\mathcal{G}_D$, we split the diffusion graph into $N = 14$ periods.

Table 1: Experimental results on HITS score over four datasets (%).

Models	Twitter			Douban			Android			Meme		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
NDM	19.34	29.41	35.73	10.13	21.23	31.25	3.39	9.53	15.72	20.83	36.63	45.83
FOREST	25.52	38.50	46.07	18.68	30.84	38.57	7.00	15.14	22.37	29.63	47.80	57.86
DyHGCN	29.01	46.88	57.19	19.87	32.89	39.42	8.42	19.15	26.79	29.52	48.64	58.48
MS-HGAT	29.96	46.54	57.35	20.65	35.04	41.36	10.49	19.87	27.81	28.43	49.66	60.47
MINDS	32.56	48.60	58.60	18.83	31.35	37.97	9.49	18.68	25.93	27.42	47.34	58.68
DisenIDP	34.89	50.74	59.67	20.59	35.45	42.84	9.30	19.09	26.61	31.14	51.75	62.67
RotDiff	35.90	52.46	61.21	22.16	38.23	46.37	11.44	23.04	**31.30**	30.66	51.70	62.06
TSCLA	**36.57**	**54.97**	**65.21**	**31.48**	**46.84**	**54.05**	**12.27**	**23.29**	31.21	**37.17**	**57.82**	**66.96**
Improvement	1.87	4.78	6.53	42.07	22.53	16.57	7.26	1.09	-	16.58	10.29	6.74

Table 2: Experimental results on MAP score over four datasets (%).

Models	Twitter			Douban			Android			Meme		
	@10	@50	@100	@10	@50	@100	@10	@50	@100	@10	@50	@100
NDM	12.96	13.39	13.48	8.36	8.79	9.36	2.19	2.44	2.52	10.59	11.31	11.44
FOREST	17.33	17.90	18.01	10.86	11.46	11.83	3.81	4.16	4.26	15.53	16.37	17.51
DyHGCN	17.51	18.32	18.47	10.48	11.14	11.48	4.58	5.03	5.14	16.11	16.23	17.25
MS-HGAT	18.80	19.51	19.65	11.22	11.87	11.98	6.33	6.75	6.85	15.42	16.41	16.57
MINDS	18.62	19.29	19.42	10.19	10.76	10.85	6.13	6.53	6.62	15.35	16.23	16.38
DisenIDP	22.46	23.58	23.71	10.41	11.10	11.21	5.69	6.11	6.21	16.66	17.62	17.77
RotDiff	24.06	24.82	24.95	11.70	12.54	12.66	6.96	7.45	7.56	16.53	16.91	17.66
TSCLA	**24.24**	**25.11**	**25.25**	**17.13**	**17.87**	**17.97**	**7.25**	**7.74**	**7.86**	**20.15**	**21.13**	**21.26**
Improvement	7.93	1.17	1.20	46.43	42.54	41.97	4.22	3.92	3.97	20.92	19.92	19.65

Table 3: Statistics of the datasets.

Datasets	**Twitter**	**Douban**	**Android**	**Meme**
# Users	12,627	12,232	9,958	4,709
# Links	309,631	348,280	48,573	-
# Cascades	3,442	3,475	679	12,661
Avg. Length	32.60	21.76	33.3	16.24

All GNN layer $\Psi(\cdot)$ used for generating user embeddings are implemented as GCNs. our model is flexible and can accommodate other GNN architectures. For the temperature τ in our structural contrastive learning module, we select it from $\{0.1, 0.3, 0.5, 0.7, 1\}$ for each dataset. The number of negative samples K is set to 150. The number of heads H in Eq. 8 is chosen from $\{8, 10, 16\}$ and is empirically set to 10 after comparison. The hyper-parameters γ_g and γ_c are set to 0.1.

Following the previous works (Yuan et al., 2020; Qiao et al., 2023), we set the maximum cascade length to 200. For all datasets, 80% of cascades are used for training; the remaining 20% are split evenly for validation and testing. Following prior works (Yuan et al., 2020; Sun et al., 2022), we frame the task as an information retrieval task and evaluate the performance with two ranking metrics, *i.e.*, Mean Average Precision on top K (MAP@K) and Hits scores on top K (HITS@K), with $K = [10, 50, 100]$. We abbreviate them as M@K and H@K, respectively.

5.2 Experimental Results

5.2.1 Performance Comparison

The experimental results of the information diffusion prediction task are shown in Table 1 and Table 2. Numbers in bold denote the best results among all models, and the underlined ones denote the second-best results. With the result, we have the following observations: **(O1)** TSCLA achieves significant improvements over state-of-the-art baselines, with over 40% relative improvement on the Douban dataset in MAP scores. The result shows that identifying users' critical connections within noisy structures enhances the retrieval of relevant users. Moreover, integrating contextual information into each diffusion period enables more precise localization of relevant user preferences. **(O2)** We attribute the variation in TSCLA's performance across different datasets to the inherent characteristics of each dataset. In the Douban and Meme datasets, our model achieves better results by identifying critical user interactions from users' weak social connections. However, Twitter and Android datasets are derived from real-world social networks. They contain social relationships that are

Table 4: Ablation study results on three datasets. (%)

Models	Twitter				Douban				Android			
	H@50	H@100	M@50	M@100	H@50	H@100	M@50	M@100	H@50	H@100	M@50	M@100
TSCLA	**54.97**	**65.21**	**25.11**	**25.25**	**46.84**	**54.05**	**17.87**	**17.97**	23.04	**31.21**	**7.74**	**7.86**
TSCLA$_{\text{-DY}}$	53.54	64.45	24.74	24.87	41.55	48.98	15.12	15.23	20.65	28.57	6.80	6.91
TSCLA$_{\text{-ST}}$	48.21	60.99	18.59	18.74	28.46	34.71	10.74	10.83	19.15	27.42	6.64	6.76
TSCLA$_{\text{-SC}}$	54.20	63.70	19.56	19.71	36.73	44.28	13.37	13.47	**23.21**	30.51	7.73	7.84
TSCLA$_{\text{-TC}}$	53.90	64.78	23.70	23.85	46.70	53.60	17.10	17.20	21.51	29.19	6.95	7.06
TSCLA$_{\text{-NC}}$	53.53	62.78	19.33	19.49	36.28	44.43	13.18	13.29	21.28	28.75	6.87	6.98
TSCLA$_{\text{-HA}}$	54.59	63.54	24.44	24.57	45.55	52.66	17.05	17.15	20.11	28.18	6.69	6.80

unrelated to the diffusion process. Without additional information, our model faces challenges in distinguishing between meaningful interactions and the underlying noise in these networks.

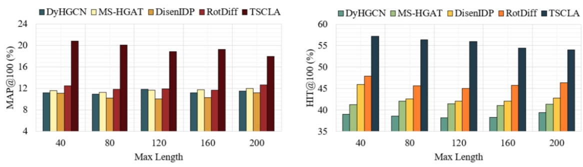
5.2.2 Ablation Study

To validate the contribution of each component in TSCLA, we designed six variants for our model and conducted an ablation study on them. They are: 1. TSCLA$_{\text{-DY}}$ removes temporal contrastive enhancement user encoding module. 2. TSCLA$_{\text{-ST}}$ removes the structural contrastive enhancement user encoding module. 3. TSCLA$_{\text{-SC}}$ removes the structural contrastive enhancement module . 4. TSCLA$_{\text{-TC}}$ removes the temporal contrastive enhancement module. 5. TSCLA$_{\text{-AC}}$ removes both temporal and structural contrastive enhancements. 6. TSCLA$_{\text{-HA}}$ replaces the hierarchical user adaptation module with a simple lookup module in (Sun et al., 2022).

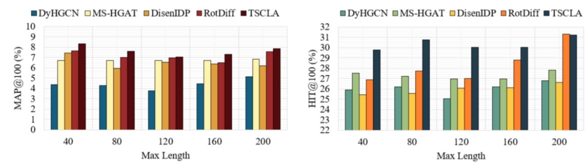
The results of these variants are shown in Table 4. By analyzing the results, we have the following observations: **(O1)** The performance of TSCLA$_{\text{-ST}}$ and TSCLA$_{\text{-DY}}$ demonstrates that removing any user encoding mechanism would result in performance degradation. Notably, TSCLA$_{\text{-ST}}$ experiences a more significant drop, underscoring the importance of identifying critical user connections for accurate diffusion modeling. **(O2)** When we omit any contrastive learning enhancement modules, the performance drops across all datasets, demonstrating their importance. **(O3)** When we replace the hierarchical user adaptation module with a memory lookup module (TSCLA$_{\text{-HA}}$), it leads to a performance decline. It implies that without the guidance of the user context of the current cascade, extracting helpful information from users' diversified preferences becomes challenging.

5.2.3 Parameter Analysis

We conduct comparative experiments to analyze the effect of maximum cascade length on model performance. The results in Fig. 3 show that TSCLA can outperform other models under any cascade lengths, illustrating its stability and effectiveness. We contribute its remarkable performance to our structural contrastive learning user encoding module, which could extend user preference from a global perspective regardless of cascade length and duration.



(a) Douban



(b) Android

Figure 3: Impact of maximum cascade length.

6 Conclusion

In this paper, we propose temporal and structural contrastive learning augmented graph neural networks (**TSCLA**). Specifically, we extract users' dynamic preferences from discretized structure snapshots and design a temporal contrastive learning module that extracts users' diversified preferences in the diffusion process. Furthermore, we propose a hierarchical adaptation module that dynamically fuses relevant preferences based on the user context of the current cascade. Additionally, we develop a structural contrastive learning module to identify critical user relations within noisy structural information to extend users' dynamic preferences. Extensive experiments on four real-world datasets validate the robustness and effectiveness of TSCLA.

7 Limitations

We consider our model to have the following three limitations:

1) Inefficient model updating & training. Since our model extracts users' periodic preferences by splitting the historical diffusion process into diffusion periods, it requires continuously incorporating new diffusion subgraphs to learn the users' latest preferences for more timely predictions. Furthermore, our model utilizes the multi-head attention mechanism to strengthen user correlation relationships at each time period. With the increasing sub-graphs, this module's computational complexity will also grow, which will result in inefficient model training in the future.

2) Transductive graph learning. As our model only explores structural factors that affect the online information diffusion process, it is limited to handling graph structures with a fixed node set and cannot effectively address dynamic node changes. Although the issue of node addition can be alleviated by incorporating user features (*e.g.*, gender, age, re-posting frequency) into the graph encoders, it remains challenging for our model to handle node deletion, a limitation that has been effectively addressed in some continuous-time dynamic graph networks.

3) Ignoring other factors of the diffusion process. The online information diffusion process is a complex process. The dissemination of specific information by users is influenced not only by the social structure or historical diffusion structures but also by many other factors, including the content's semantics, social background, and even the publishing time. Moreover, with the continuous advancement of recommendation systems, the diffusion pattern of online information in social networks has shifted from traditional word-of-mouth propagation to a model increasingly reliant on recommendation systems for amplification. Thus, our model TSCLA, which is based solely on the diffusion structure, lacks the incorporation of more intricate factors, making it challenging to accurately and efficiently identify potential disseminators who may participate in the current information propagation.

References

Simon Bourigault, Sylvain Lamprier, and Patrick Gallinari. 2016. Representation learning for information diffusion through social networks: an embedded cascade model. In *Proceedings of the Ninth ACM International Conference on Web Search and Data Mining, San Francisco, CA, USA, February 22-25, 2016*, pages 573–582. ACM.

Zhangtao Cheng, Wenxue Ye, Leyuan Liu, Wenxin Tai, and Fan Zhou. 2023. Enhancing information diffusion prediction with self-supervised disentangled user and cascade representations. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, pages 3808–3812. ACM.

Xavier Glorot and Yoshua Bengio. 2010. Understanding the difficulty of training deep feedforward neural networks. In *Proceedings of the Thirteenth International Conference on Artificial Intelligence and Statistics, AISTATS 2010, Chia Laguna Resort, Sardinia, Italy, May 13-15, 2010*, volume 9 of *JMLR Proceedings*, pages 249–256. JMLR.org.

Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural Comput.*, 9(8):1735–1780.

Nathan Hodas and Kristina Lerman. 2014. The simple rules of social contagion. *Scientific reports*, 4.

Mohammad Raihanul Islam, Sathappan Muthiah, Bijaya Adhikari, B. Aditya Prakash, and Naren Ramakrishnan. 2018. Deepdiffuse: Predicting the 'who' and 'when' in cascades. In *IEEE International Conference on Data Mining, ICDM 2018, Singapore, November 17-20, 2018*, pages 1055–1060. IEEE Computer Society.

Paras Jain, Ajay Jain, Tianjun Zhang, Pieter Abbeel, Joseph Gonzalez, and Ion Stoica. 2021. Contrastive code representation learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 5954–5971. Association for Computational Linguistics.

Pengfei Jiao, Hongqian Chen, Qing Bao, Wang Zhang, and Huaming Wu. 2024. Enhancing multi-scale diffusion prediction via sequential hypergraphs and adversarial learning. In *Thirty-Eighth AAAI Conference on Artificial Intelligence, AAAI 2024, Thirty-Sixth Conference on Innovative Applications of Artificial Intelligence, IAAI 2024, Fourteenth Symposium on Educational Advances in Artificial Intelligence, EAAI 2014, February 20-27, 2024, Vancouver, Canada*, pages 8571–8581. AAAI Press.

David Kempe, Jon M. Kleinberg, and Éva Tardos. 2003. Maximizing the spread of influence through a social network. In *Proceedings of the Ninth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Washington, DC, USA, August 24 - 27, 2003*, pages 137–146. ACM.

Jure Leskovec, Lars Backstrom, and Jon M. Kleinberg. 2009. Meme-tracking and the dynamics of the news cycle. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Paris, France, June 28 - July 1, 2009*, pages 497–506. ACM.

Yixin Liu, Ming Jin, Shirui Pan, Chuan Zhou, Yu Zheng, Feng Xia, and Philip S. Yu. 2023a. Graph self-supervised learning: A survey. *IEEE Trans. Knowl. Data Eng.*, 35(6):5879–5900.

Yong Liu, Susen Yang, Yinan Zhang, Chunyan Miao, Zaiqing Nie, and Juyong Zhang. 2023b. Learning hierarchical review graph representations for recommendation. *IEEE Trans. Knowl. Data Eng.*, 35(1):658–671.

Zhen Peng, Wenbing Huang, Minnan Luo, Qinghua Zheng, Yu Rong, Tingyang Xu, and Junzhou Huang. 2020. Graph representation learning via graphical mutual information maximization. In *WWW '20: The Web Conference 2020, Taipei, Taiwan, April 20-24, 2020*, pages 259–270. ACM / IW3C2.

Hongliang Qiao, Shanshan Feng, Xutao Li, Huiwei Lin, Han Hu, Wei Wei, and Yunming Ye. 2023. Rotdiff: A hyperbolic rotation representation model for information diffusion prediction. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, pages 2065–2074. ACM.

Aravind Sankar, Xinyang Zhang, Adit Krishnan, and Jiawei Han. 2020. Inf-vae: A variational autoencoder framework to integrate homophily and influence in diffusion prediction. In *WSDM '20: The Thirteenth ACM International Conference on Web Search and Data Mining, Houston, TX, USA, February 3-7, 2020*, pages 510–518. ACM.

Ling Sun, Yuan Rao, Xiangbo Zhang, Yuqian Lan, and Shuanghe Yu. 2022. MS-HGAT: memory-enhanced sequential hypergraph attention network for information diffusion prediction. In *Thirty-Sixth AAAI Conference on Artificial Intelligence, AAAI 2022, Thirty-Fourth Conference on Innovative Applications of Artificial Intelligence, IAAI 2022, The Twelveth Symposium on Educational Advances in Artificial Intelligence, EAAI 2022 Virtual Event, February 22 - March 1, 2022*, pages 4156–4164. AAAI Press.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008.

Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks. In *6th International Conference on Learning Representations, ICLR 2018, Vancouver, BC, Canada, April 30 - May 3, 2018, Conference Track Proceedings*. OpenReview.net.

Ding Wang, Lingwei Wei, Chunyuan Yuan, Yinan Bao, Wei Zhou, Xian Zhu, and Songlin Hu. 2022. Cascade-enhanced graph convolutional network for information diffusion prediction. In *Database Systems for Advanced Applications - 27th International Conference, DASFAA 2022, Virtual Event, April 11-14, 2022, Proceedings, Part I*, volume 13245 of *Lecture Notes in Computer Science*, pages 615–631. Springer.

Haiwen Wang, Ruijie Wang, Chuan Wen, Shuhao Li, Yuting Jia, Weinan Zhang, and Xinbing Wang. 2020. Author name disambiguation on heterogeneous information network with adversarial representation learning. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 238–245. AAAI Press.

Jia Wang, Vincent W. Zheng, Zemin Liu, and Kevin Chen-Chuan Chang. 2017. Topological recurrent neural network for diffusion prediction. In *2017 IEEE International Conference on Data Mining, ICDM 2017, New Orleans, LA, USA, November 18-21, 2017*, pages 475–484. IEEE Computer Society.

Ruijie Wang, Zijie Huang, Shengzhong Liu, Huajie Shao, Dongxin Liu, Jinyang Li, Tianshi Wang, Dachun Sun, Shuochao Yao, and Tarek F. Abdelzaher. 2021. Dydiff-vae: A dynamic variational framework for information diffusion prediction. In *SIGIR '21: The 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual Event, Canada, July 11-15, 2021*, pages 163–172. ACM.

Zhitao Wang, Chengyao Chen, and Wenjie Li. 2018a. Attention network for information diffusion prediction. In *Companion of the The Web Conference 2018 on The Web Conference 2018, WWW 2018, Lyon , France, April 23-27, 2018*, pages 65–66. ACM.

Zhitao Wang, Chengyao Chen, and Wenjie Li. 2018b. A sequential neural information diffusion model with structure attention. In *Proceedings of the 27th ACM International Conference on Information and Knowledge Management, CIKM 2018, Torino, Italy, October 22-26, 2018*, pages 1795–1798. ACM.

Zhitao Wang and Wenjie Li. 2019. Hierarchical diffusion attention network. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 3828–3834. ijcai.org.

Lingwei Wei, Dou Hu, Wei Zhou, Zhaojuan Yue, and Songlin Hu. 2021. Towards propagation uncertainty: Edge-enhanced bayesian graph convolutional networks for rumor detection. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP*

2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021, pages 3845–3854. Association for Computational Linguistics.

Lirong Wu, Haitao Lin, Cheng Tan, Zhangyang Gao, and Stan Z. Li. 2023. Self-supervised learning on graphs: Contrastive, generative, or predictive. *IEEE Trans. Knowl. Data Eng.*, 35(4):4216–4235.

Cheng Yang, Maosong Sun, Haoran Liu, Shiyi Han, Zhiyuan Liu, and Huanbo Luan. 2021. Neural diffusion model for microscopic cascade study. *IEEE Trans. Knowl. Data Eng.*, 33(3):1128–1139.

Cheng Yang, Jian Tang, Maosong Sun, Ganqu Cui, and Zhiyuan Liu. 2019. Multi-scale information diffusion prediction with reinforced recurrent networks. In *Proceedings of the Twenty-Eighth International Joint Conference on Artificial Intelligence, IJCAI 2019, Macao, China, August 10-16, 2019*, pages 4033–4039. ijcai.org.

Yang Yang, Jie Tang, Cane Wing-ki Leung, Yizhou Sun, Qicong Chen, Juanzi Li, and Qiang Yang. 2015. RAIN: social role-aware information diffusion. In *Proceedings of the Twenty-Ninth AAAI Conference on Artificial Intelligence, January 25-30, 2015, Austin, Texas, USA*, pages 367–373. AAAI Press.

Liang Yao, Chengsheng Mao, and Yuan Luo. 2019. Graph convolutional networks for text classification. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7370–7377. AAAI Press.

Chunyuan Yuan, Jiacheng Li, Wei Zhou, Yijun Lu, Xiaodan Zhang, and Songlin Hu. 2020. Dyhgcn: A dynamic heterogeneous graph convolutional network to learn users' dynamic preferences for information diffusion prediction. 12459:347–363.

Chunyuan Yuan, Qianwen Ma, Wei Zhou, Jizhong Han, and Songlin Hu. 2019. Jointly embedding the local and global relations of heterogeneous graph for rumor detection. In *2019 IEEE International Conference on Data Mining, ICDM 2019, Beijing, China, November 8-11, 2019*, pages 796–805. IEEE.

Yanning Zhou, Hang Xu, Wei Zhang, Bin Gao, and Pheng-Ann Heng. 2021. C^3-semiseg: Contrastive semi-supervised segmentation via cross-set learning and dynamic class-balancing. In *2021 IEEE/CVF International Conference on Computer Vision, ICCV 2021, Montreal, QC, Canada, October 10-17, 2021*, pages 7016–7025. IEEE.

Yanqiao Zhu, Yichen Xu, Feng Yu, Qiang Liu, Shu Wu, and Liang Wang. 2020. Deep graph contrastive representation learning. *CoRR*, abs/2006.04131.