Dynamic Heterogeneous Graph Embedding via Heterogeneous Hawkes Process

Yugang Ji[1], Tianrui Jia[1], Yuan Fang[2], and Chuan Shi[1]✉

[1] Beijing University of Posts and Telecommunications
[2] Singapore Management University
{jiyugang, jiatianrui}@bupt.edu.cn, yfang@smu.edu.sg,
shichuan@bupt.edu.cn

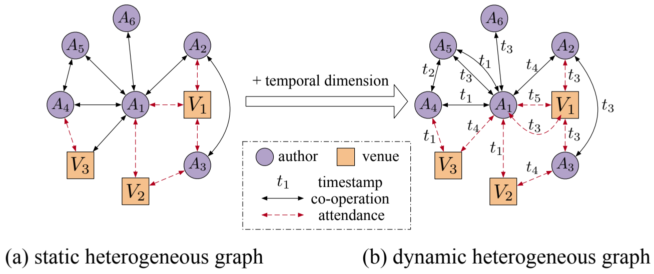
Abstract. Graph embedding, aiming to learn low-dimensional representations of nodes while preserving valuable structure information, has played a key role in graph analysis and inference. However, most existing methods deal with static homogeneous topologies, while graphs in real-world scenarios are gradually generated with different-typed temporal events, containing abundant semantics and dynamics. Limited work has been done for embedding dynamic heterogeneous graphs since it is very challenging to model the complete formation process of heterogeneous events. In this paper, we propose a novel **H**eterogeneous Hawkes **P**rocess based dynamic **G**raph **E**mbedding (**HPGE**) to handle this problem. HPGE effectively integrates the Hawkes process into graph embedding to capture the excitation of various historical events on the current type-wise events. Specifically, HPGE first designs a heterogeneous conditional intensity to model the base rate and temporal influence caused by heterogeneous historical events. Then the heterogeneous evolved attention mechanism is designed to determine the fine-grained excitation to different-typed current events. Besides, we deploy the temporal importance sampling strategy to sample representative events for efficient excitation propagation. Experimental results demonstrate that HPGE consistently outperforms the state-of-the-art alternatives.

Keywords: Dynamic heterogeneous graph · Graph embedding · Heterogeneous Hawkes process · Heterogeneous evolved attention mechanism.

1 Introduction

Graphs, such as social networks, e-commerce platforms and academic graphs, occur naturally in various real-world applications. Recently, graph embedding, whose goal is to encode high-dimensional non-Euclidean structures into low-dimensional vector space [2,10], has shown great popularity in tackling graph analytic problems such as node classification and link predictions.

Most existing graph embedding methods focus on modeling static homogeneous graphs, where both edges and nodes are of the same type and never change over time. However, in the real world, complex systems are commonly associated with multiple temporal interactions between different-typed nodes, forming the



(a) static heterogeneous graph (b) dynamic heterogeneous graph

Fig. 1: Toy examples of static and dynamic heterogeneous graphs.

so-called dynamic heterogeneous graphs. Taking Fig. 1(b) as an example, there are two types of interactions ("co-operation" and "attendance") between two types of nodes (authors and venues) and each interaction is marked with a continuous timestamp to describe when it happened, compared to the static one in Fig. 1(a). Dynamic heterogeneous graphs indeed describe richer semantics and dynamics besides structural information, indicating the multiple evolutions of node representations, compared to static homogeneous graphs.

Paying attention to the abundant semantics, there have been several heterogeneous graph embedding methods [5,12,27,34], taking into account both types of nodes and edges when learning representations. While earlier approaches [5,6] employ shallow skip-gram models on heterogeneous sequences generated by metapaths [24], recent studies [7,12,27,34] apply deeper graph neural networks (GNNs) which usually gather information from heterogeneous neighborhoods to enhance node representations. On the other line, to capture the temporal evolution of dynamic graphs, it is general to split the whole graph into several snapshots and generate representations by inputting all snapshot-based embeddings into sequential models like Long-Short Term Memory (LSTM) and Gated Recurrent Units (GRU) [22,8,19]. Recently, aware of the fact that historical events (i.e., temporal edges) consistently influence and excite the generation of current interactions, recent researchers [36,17,29] attempt to introduce temporal point process, especially Hawkes process, into graph embedding to model the formation process of dynamic graphs.

However, limited work has been done for embedding dynamic heterogeneous graphs. The semantics and dynamics introduce two essential challenges:

First, how to model the continuous dynamics of heterogeneous interactions? Although several works attempt to describe the formation process as sequential heterogeneous snapshots [1,18,32], the heterogeneous dynamics can only be reflected via the number of snapshots, while different-typed edges are indeed continuously generated over time. For instance, as shown in Fig. 1(a), heterogeneous events like "co-operation" and "attendance" are continuously generated over time and historical connections can excite current events. A naïve idea is to integrate Hawkes process into graph embedding, inspired by [36,17,29]. However, these methods deal with homogeneous events and cannot directly introduce into heterogeneous graphs.

Second, how to model the complex influence of different semantics? While different semantics indicate different views of information, they usually impact current various interactions in different patterns. While existing methods only model the difference of semantics [27,32], they neglect that the influence to different-typed current or future events could be very different. For example, in Fig. 1(b), the co-operation between A_1 and A_5 at T_3 could be excited more from historical co-operation events of A_4 and A_5, rather than the attendance between A_4 and V_3. Meanwhile, the attendance between A_1 and V_3 at time t_4 would be affected more from the historical attendance events of A_4. In a word, different-typed historical events would excite different-typed current events in different patterns.

Motivated by these challenges, we propose the **H**eterogeneous Hawkes **P**rocess for Dynamic Heterogeneous **G**raph **E**mbedding (**HPGE**). To handle the continuous dynamics, we treat heterogeneous interactions as multiple temporal events, which gradually occur over time, and introduce Hawkes process into heterogeneous graph embedding by designing a *heterogeneous conditional intensity* to model the excitation of historical heterogeneous events to current events. To handle the complex influence of semantics, we further design the *heterogeneous evolved attention mechanism* which considers both the intra-typed temporal importance of historical events but also the inter-typed temporal impacts from multiple historical events to current type-wise events. Moreover, as current events are influenced more by past important interactions, we adopt the temporal importance sampling strategy to select representative events from historical candidates, balancing their importance and recency. The contributions of this work are summarized as follows.

– We introduce Hawkes process into dynamic heterogeneous graph embedding, which can preserve both semantics and dynamics by learning the formation process of all heterogeneous temporal events. Although few works [17,36] attempt to model the formation process of graphs, they pay no attention to types of either historical or current events.
– Our proposed approach HPGE not only integrates complex evolved excitation of events but also enables efficient extraction of representative past events. To these ends, we respectively design the heterogeneous evolved attention mechanism and the temporal importance sampling strategy.
– We study the effectiveness and efficiency of HPGE empirically on three public datasets and the experimental results of node classification and temporal link prediction demonstrate that HPGE consistently outperforms the state-of-the-art alternatives.

2 Related Work

We discuss the related work on two lines, namely, static graph embedding and dynamic graph embedding, taking both homogeneous and heterogeneous methods into consideration.

Static graph embedding. This line of methods are to embed non-Euclidean structures into low-dimensional vector space. Earlier methods [23,9]input random walk-based contextual sequences into skip-gram framework to preserve relevance of connected nodes. Recently, graph neural networks (GNNs) [16,11,25] have attached much attention for their ability to integrate neighborhood influence via message passing. However, they neglect the types of either edges or nodes, and thus fail to model the abundant semantics in real-world graphs. Focus on dealing with heterogeneity, previous Metapath2Vec [5] and HIN2Vec [6] associate nodes by their local proximity through heterogeneous sequences, while current works focus on heterogeneous GNNs [7,35] to better exploit structures and semantics over the whole graph. In these methods, various heterogeneous attention mechanisms are designed to enhance traditional information aggregation [27,3,7,12,34]. More detailed discussions are summarized in [2,26]. However, all the above methods cannot deal with dynamic heterogeneous graphs because of overlooking evolution within interactions.

Dynamic graph embedding. On another line, there is significant research interest in dynamic graph embedding (also called temporal network embedding) during the past decade. CTDNE [21] considers dynamics as temporal bias and deploy temporal random walks to learn nodes. TGAT [30] designs a temporal encoder to project continuous timestamps as temporal vectors. Aware of the dynamic evolution of graphs, recent works prefer to split a graph into several snapshots and integrate deep auto-encoders [8] or recurrent neural networks [20,22] to learn the evolving embeddings. Focusing on handle both dynamics and semantics, dynamic heterogeneous graph embedding has also been explored to some extent [12,13,32,31]. Nevertheless, the performance of these methods is often limited as the timestamps of interactions in a snapshot are removed, whereas the formation process of graphs remains unknown. Recently, temporal point processes, most notably the Hawkes process, have become popular for their ability to simulate the formation history [36,17]. However, they are designed for homogeneous graphs while the heterogeneity introduces essential challenges to learn and inference.

3 Preliminaries

In this section, we introduce the definition of dynamic heterogeneous graphs, the problem of dynamic heterogeneous graph embedding as well as the general Hawkes process framework.

Definition 1 ***Dynamic Heterogeneous Graph.*** *A dynamic heterogeneous graph is* $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T}, \mathcal{O}, \mathcal{R})$ *where* $\mathcal{V}$ *denotes the set of nodes,* $\mathcal{E}$ *denotes the* *temporal edges (i.e., events), $\mathcal{T}$ denotes the set of timestamps, $\mathcal{O}$ and $\mathcal{R}$ respectively denote node and edge types. In addition, there are two corresponding type mapping functions including $\phi : \mathcal{V} \to \mathcal{O}$ and $\psi : \mathcal{E} \to \mathcal{R}$. Notice that, each event is a quad $e = (v_i, v_j, t, r)$ where v_i and v_j are source and target nodes, $t \in \mathcal{T}$ is the continuous timestamp and $r \in \mathcal{R}$ is the event type.*

For instance, the academic graph in Fig. 1(b) consists of two types of nodes (i.e., authors and venues), two types of events (i.e., "co-operation" and "attendance") as well as the continuous timestamps t_1, t_2, t_3, t_4 and t_5 of these heterogeneous events, naturally forming a dynamic heterogeneous graphs. Obviously, heterogeneous events gradually happen and excite future interactions over time, expressing abundant semantics and dynamics, compared to static graphs.

Definition 2 ***Dynamic Heterogeneous Graph Embedding.*** *Given a dynamic heterogeneous graph $\mathcal{G}$, the goal of dynamic heterogeneous graph embedding is to learn a representation function $\mathcal{H}$ to project such a high-dimensional non-Euclidean structures into low-dimensional vector space, namely, $\mathcal{H}(\mathcal{G}) \to \boldsymbol{H}$, $\boldsymbol{H} \in \mathcal{R}^{|\mathcal{V}| \times d}$ where $|\mathcal{V}|$ and d are the size and dimension of nodes, $d \ll |\mathcal{V}|$. Meanwhile, both the dynamics and semantics besides structural information should be preserved as well.*

Definition 3 ***Hawkes process.*** *Hawkes process is a typical temporal point process with the assumption that historical events can influence the occurrence of the current event. Given historical events $\{e_h | t_h < t\}$ before current time t, a conditional intensity function is defined to characterizes the arrival rate of current event e, namely,*

$$\lambda(e) = \mu(e) + \sum_{e_h : t_h < t} \kappa(t - t_h), \tag{1}$$

where $\mu(e)$ is the base intensity (i.e., spontaneous arrival rate) of current event e, $\kappa(\cdot)$ is a time decay effect of historical events on the current e.

Obviously, the temporal excitation is well modeled and there are several works [17,36] attempt to embed dynamic graphs with Hawkes process. Nevertheless, these methods cannot handle the heterogeneity. In this paper, we focus on introducing Hawkes process into dynamic heterogeneous graph embedding, to learn the complete temporal formation process of heterogeneous events, keeping both semantics and dynamics.

4 The Proposed HPGE Model

In this section, we propose our model called HPGE. We begin with an overview, before zooming into the details.

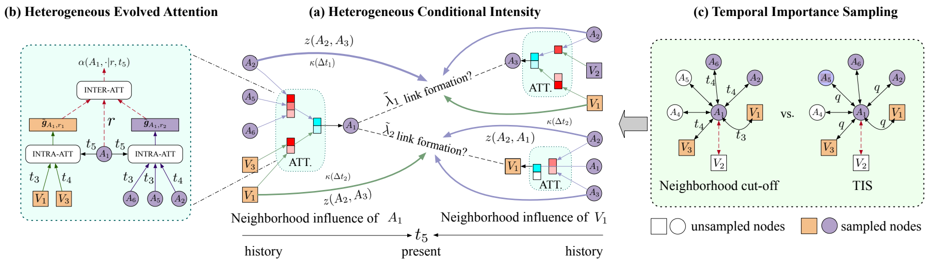


Fig. 2: The overall architecture of HPGE. (a) Heterogeneous conditional intensity function to model the heterogeneous temporal influence of A_1, A_3 or V_1, (b) Heterogeneous evolved attention to measure the relevance and evolution from historical neighbors to current type-wise event, consisting of intra- and inter-typed temporal attention, (c) Temporal importance sampling of heterogeneous events where q denotes the sampling probability and the nodes in white are unsampled, in comparison to a naïve cut-off strategy.

4.1 Overview

There are three main components of HPGE, namely, the heterogeneous conditional intensity function to learn the semantics and dynamics within the formation process of heterogeneous temporal events, the heterogeneous evolved attention mechanism to measure the importance and evolution from historic neighborhoods to current type-wise event, and the temporal importance sampling to handle the efficient extraction of representative events. First, as shown in Fig. 2(a), given the respective temporal heterogeneous neighbors of A_1, A_3, and V_1, HPGE evaluates the affinity between each node and its neighbors with a type-wise influence measure. Subsequently, hinged on a heterogeneous conditional intensity function, it accumulates the influence from historical heterogeneous neighbors, which characterizes the arrival rate at present. Second, an attentive manner is designed in 2(b) to capture both the temporal importance of same-typed neighborhoods (intra-att) and the evolution from historical types to the current type (inter-att). Third, as the graph evolves, in Fig. 2(c), the number of events gradually grows. For effective and efficient HPGE, we adopt a Temporal Importance Sampling (TIS) strategy to extract representative neighbors in both temporal and structural dimensions, instead of using the full neighborhood which is inefficient, or the traditional cut-off strategy based on recency only.

4.2 Heterogeneous conditional intensity modeling

On a dynamic heterogeneous graph, various kinds of interactions are constantly being established over time, which can be regarded as a series of observed heterogeneous events. Intuitively, the current events are influenced by past events, and the heterogeneity of events implies different strengths of influence. For instance, attendance in a conference at present is influenced by different historical views, including the past attendance view and author collaboration view. Therefore, given current event $e = (v_i, v_j, t, r)$, we introduce the general heterogeneous conditional intensity function as follows:

$$\begin{aligned}\tilde{\lambda}(e) = \; & \underbrace{\mu_r(v_i, v_j)}_{\text{base rate}} \\ & + \underbrace{\gamma_1 \textstyle\sum_{r' \in \mathcal{R}} \sum_{p \in \mathcal{N}_{i,r',<t}} \alpha(p, e) z(v_p, v_j) \kappa_i(t - t_p)}_{\text{neighborhood influence on source } v_i}, \\ & + \underbrace{\gamma_2 \textstyle\sum_{r'' \in \mathcal{R}} \sum_{q \in \mathcal{N}_{j,r'',<t}} \alpha(q, e) z(v_q, v_i) \kappa_j(t - t_q)}_{\text{neighborhood influence on target } v_j}\end{aligned} \tag{2}$$

where γ_1 and γ_2 are the balance parameters. This conditional intensity function consists of three major parts, including the type-wise base rate, the heterogeneous neighborhood on source node v_i and on target node v_j. At first, given v_i and v_j as well as event type r, the base rate $\mu_r(v_i, v_j)$ is defined as:

$$\mu_r(v_i, v_j) = -\sigma(f(\boldsymbol{h}_i \boldsymbol{W}_{\phi(v_i)} - \boldsymbol{h}_j \boldsymbol{W}_{\phi(v_j)}) \boldsymbol{W}_r + b_r), \tag{3}$$

where $\boldsymbol{h}_i \in \mathbb{R}^d$ and $\boldsymbol{h}_j \in \mathbb{R}^d$ are the embedding of v_i and v_j, d is the dimension of node embedding, $\boldsymbol{W}_{\phi(\cdot)} \in \mathbb{R}^{d \times d}$ denotes the type-$\phi(\cdot)$ projection matrix, $f(\cdot)$ denotes the element-level non-negative operation to measure the symmetrical similarity of v_i and v_j, and we adopt self Hadamard product in this paper, namely $f(\boldsymbol{X}) = \boldsymbol{X} \odot \boldsymbol{X}$, $\boldsymbol{W}_r$ and b_r are the projection and bias of type-r events, $\sigma(\cdot)$ is the non-linear activate function. In the base rate evaluation, both the types of nodes and edges are taken into consideration.

Besides, historical neighbors can continuously excite the occurrence of the current event. Taking the neighborhood influence on source node as an example, given its historical neighborhoods $\{\mathcal{N}_{i,r',<t} | r' \in \mathcal{R}\}$ the excitation is indeed associated with three aspects, (1) the time span to the current time, (2) the relevant historical neighbors to target node v_j and (3) the importance of historical neighbors to source node v_i. As the time decay to different nodes are different, we design $\kappa_i(\mathit{\Delta} t)$as $\exp(-\delta_i(\mathit{\Delta} t))$, where $delta_i$ is the learnable personalized parameter and the influence become exponentially weak over time. The relevance between historical neighbors and target nodes are related to their types as well, namely,

$$z(v_p, v_j) = -\|\boldsymbol{h}_p \boldsymbol{W}_{\phi(p)} - \boldsymbol{h}_j \boldsymbol{W}_{\phi(j)}\|_2^2, \tag{4}$$

where $\| \cdot \|_2^2$ denotes the Euclidean distance measure, and the negative symbol indicates that closer nodes could affect greater. To measure the importance to source node, attention mechanisms [12,27,7] have shown powerful performance on static heterogeneous graphs. However, when dealing with the heterogeneous formation process, the complex temporal influence between different semantics remains an essential challenge. To handle the second challenge, we design the heterogeneous evolved attention mechanism in Section 4.3.

4.3 Heterogeneous evolved attention mechanism

As mentioned in Section 1, the excitation of historical interactions not only associate with types of historical events but also depend on types of current events. Thus, the importance to current event $\alpha(p, e)$ is defined as

$$\alpha(p, e) = \xi(v_p, t_p|r', v_i, t)\beta(r|r', v_i, t), \tag{5}$$

where r' and r respectively denote the type of historical and current event, t_p and t are the corresponding timestamps, $\xi(v_p, t_p|r', v_i, t)$ is the intra-type heterogeneous temporal attention, calculated by

$$\xi(v_p, t_p|r', v_i, t) = \mathrm{softmax}(\sigma(\kappa_i(t - t_p)[\boldsymbol{h}_i \boldsymbol{W}_{\phi(v_i)} \oplus \boldsymbol{h}_j \boldsymbol{W}_{\phi(v_j)}]\boldsymbol{W}_\xi)), \tag{6}$$

where $\boldsymbol{W}_\xi \in \mathbb{R}^{2d\times 1}$ denotes the attention projection matrix need to learn, $\oplus$ denotes the concatenation operation, $\mathrm{softmax}(x)$ is in the form of $\exp(x)/\sum_{x'} \exp(x')$. Both the heterogeneity and time decay are taken into consideration. Furthermore, we design the inter-typed $\beta(r|r', v_i, t)$ to model the relevance from historical types to current types, namely

$$\beta(r|r', v_i, t) = \mathrm{softmax}(\tanh(\tilde{\boldsymbol{g}}_i \boldsymbol{W}_r)\boldsymbol{w}_r)^{\mathrm{T}}, \tag{7}$$

where $\boldsymbol{W}_r \in \mathbb{R}^{d|\mathcal{R}|\times d_m}$ and $\boldsymbol{w}_r \in \mathbb{R}^{d_m\times 1}$ are the projection matrices need to learn, d_m is the length of latent dimension and we set $d_m = 0.5d$ here. $\tilde{\boldsymbol{g}}_i$ is the concatenation of historical excitation, namely $\tilde{\boldsymbol{g}}_i = [\tilde{\boldsymbol{g}}_{i,1} \oplus \tilde{\boldsymbol{g}}_{i,2} \oplus \cdots \oplus \tilde{\boldsymbol{g}}_{i,|\mathcal{R}|}]$, and the sub-excitation from type-r' neighbors is calculated by

$$\tilde{\boldsymbol{g}}_{i,r'} = \sigma\left(\left[\sum_p \xi(v_p, t_p|r', v_i, t)\boldsymbol{h}_p \boldsymbol{W}_{\phi(v_p)}\kappa_i(t - t_p)\right] \boldsymbol{W}_{\beta,r'} + b_{\beta,r'}\right), \tag{8}$$

where $\boldsymbol{W}_{\beta,r'} \in \mathbb{R}^{d\times d}$ and $b_{\beta,r'}$ are the projection matrix and bias need to learn. It is naturally a intra-typed attention based temporal excitation aggregation.

4.4 Temporal importance sampling

As more events are accumulated over time, it becomes expensive to materialize the heterogeneous conditional intensity function. For efficiency, existing Hawkes process on homogeneous graphs cut off events happened far away in the past, and only focus on the most recent events. However, the cut-off point is often arbitrary and difficult to set. Furthermore, the recency-only strategy risks in omitting structurally important neighbors that have frequent interactions over time. As illustrated in Fig. 2(b), A_5 would be cut off based on recency only, but it is desirable to retain A_5 for modeling due to its frequent interaction with A_1.

To efficiently extract representative candidates with both recency and structural importance, inspired by importance sampling [4,14], we propose the strategy of Temporal Importance Sampling (TIS). TIS considers both temporal and structural information to extract representation neighbors. Weighed by the excitation rate and the time decay function, we design the sampler of TIS as follows,

$$q(v_p|v_i, r', t) = \frac{\kappa_i(t - t_p)N_i(v_p)}{\sum_{v_{p'}\in\mathcal{N}_{i,r',<t}} \kappa_i(t - t'_p)N_i(v'_p)}, \tag{9}$$

where $q(v_p|v_i, r', t)$ denotes the sampling probability, depending on the importance of node v_h relating to event type r', times of historical occurrence $N_i(v_p)$ as well as time t. Thus, the estimator of the sampled neighbor influence is given by

$$z(\hat{v}_p, v_j) = \frac{1}{n} \cdot \frac{z(\hat{v}_p, v_j)}{q(\hat{v}_p|v_i, r', t)}, \quad \hat{v}_p \sim q(v_p|v_i, r', t) \tag{10}$$

where n is the sample size, $\hat{v}_p$ denotes a sampled historical neighbor. Thus, both temporal and structural importance of the neighbors can be retained for influence modeling. In particular, the estimator ensures the expectation of weighted sampled excitation is equal to propagate all historical influences.

4.5 Optimization objective

By modeling the temporal heterogeneous event formation with heterogeneous Hawkes process, the current neighbor formation events can be inferred from the heterogeneous conditional intensity. Given all the historical neighborhoods $\mathcal{N}_{i,<t}$ of v_i and $\mathcal{N}_{j,<t}$ of v_j before time t, the probability of forming type-r connection between v_i and v_j at time t can be inferred as

$$p(e_{i,j,r}|\mathcal{N}_{i,t}, \mathcal{N}_{j,t}) = \frac{\lambda(e_{i,j,r})}{\sum_{r' \in \mathcal{R}} \left(\sum_{j' \in \mathcal{N}_{i,t}^{r'}} \lambda(e_{i,j',r'}) + \sum_{i' \in \mathcal{N}_{j,t}^{r'}} \lambda(e_{i',j,r'}) \right)}, \tag{11}$$

where $\lambda(e_{i,j,r}) = \exp(\lambda(\widetilde{e_{i,j,r}}))$ denotes the positive intensity. As directed likelihood optimization would suffer from the heavily computational complexity of $p(e_{i,j,r}|\mathcal{N}_{i,t}, \mathcal{N}_{j,t})$, we consider Eq. (11) as the softmax normalization of $\tilde{\lambda}(e_{i,j,r})$, and adopt negative sampling to accelerate learning, thus, the loss of the current event e is defined as follows,

$$\mathcal{L}_{hp}(e) = -\sum_{e \in \mathcal{E}} \log \sigma(\tilde{\lambda}(e)) - \sum_{k} \mathbb{E}_{j'} \log \sigma(-\tilde{\lambda}(e_{j'})) - \sum_{k} \mathbb{E}_{i'} \log \sigma(-\tilde{\lambda}(e_{i'})), \tag{12}$$

where $e_{i'}$ and $e_{j'}$ are the abbreviations of $e_{i',j,r,t}$ and $e_{i,j',r,t}$, k is the size of negative samples, and $\mathcal{L}_{hp} = \frac{1}{|\mathcal{E}|} \sum_{e \in \mathcal{E}} \mathcal{L}_{hp}(e)$.

Besides, focusing on the downstream tasks like node classification and temporal link prediction, we design the unified loss function as follows:

$$\mathcal{L} = \mathcal{L}_{hp} + \omega_1 \mathcal{L}_{task} + \omega_2 \Omega(\boldsymbol{\Theta}), \tag{13}$$

where $\Omega(\boldsymbol{\Theta})$ is the l2-norm regularization of learnt parameters, $\mathcal{L}_{task}$ is the loss of specific tasks. For node classification and temporal link prediction, we input node embedding or the concatenation of embedding pair into a Multi-Layer Perception to extract the distribution of classifications or the probability of connections, and then evaluate the cross-entropy loss values, i.e., $\mathcal{L}_{task}$. ω_1 and ω_2 are the weights. We adopt Adam optimizer [15] to minimize the loss function for each mini-batch.

Table 1: Statistics of the three public datasets.

Datasets	Node Types	#Nodes	Event Types	#Events	Time Span
Aminer	Author (A)	23,037	A-A	71,121	16 years
	Conference (C)	22	A-C	52,399	
DBLP	Author (A)	34,766	A-A	133,684	10 years
	Venue (V)	20	A-V	98,262	
Yelp	User (U)	494,524	BrU	1,145,070	60 quarters
	Business (B)	13,507	BtU	226,728	

5 Experiments

In this section, we conduct extensive experiments on three public real-world dynamic heterogeneous graphs to demonstrate the effectiveness of HPGE.

5.1 Experimental Settings

Datasets. The three real-world datasets are the academic Aminer and DBLP graphs and the Yelp business graph. The details are introduced as follows and the statistics are listed in Table 1. (1) **Aminer** [3]. This is a benchmark bibliographic graph, which consists of two types of nodes, namely, authors (A) and conferences (C), as well as two types of temporal events, namely "co-operation" (A-A) and "attendance" (A-C). Notice that each author is labeled by one of the five research domains including data mining, database, medical informatics, theory, and visualization. (2) **DBLP** [4]. This is another bibliographic graph, which also consists of two types of temporal events between authors (A) and venues (V), namely, A-A and A-V. We follow previous work [27] to extract 20 venues in four areas, namely, database, data mining, machine learning, information retrieval. The authors are labeled by the research area they focus on. (3) **Yelp** [5]. This is a business review dataset, containing timestamped user reviews and tips on businesses. There are two types of nodes, users (U) and businesses (B), and four types of temporal events including "reviewed" (UrB), "tipped" (UtB), "reviewed by" (BrU) and "tipped by" (BtU). We extract interactions of three categories of businesses, including "Fast Food", "Sushi" and "American (New) Food", to construct the dynamic graph. Each business is labeled with its most related category.

Baselines. We compare the proposed HPGE with three groups of graph embedding models, namely, heterogeneous graph embedding (Metapath2vec [5], HEP [35], HAN [27] and HGT [12]), dynamic graph embedding (CTDNE [21], EvolveGCN [22], and M^{2}DNE [17]), and dynamic heterogeneous graph embedding (DHNE [33], DyHNE [28], and DyHATR [31]).

[3] Available at Aminer website

[4] Available at DBLP website

[5] Available at Yelp website

- **Metapath2vec** [5] and **HEP** [35]: They are two heterogeneous graph embedding models, where the former learns node embedding with sequences generated by a meta-path, and the latter propagates embedding information among different-typed interactions.
- **HAN** [27] and **HGT**: They are two attentive heterogeneous GNNs, where the former designs a hierarchical attention considering both node- and semantic-levels while the latter takes into account both the types of nodes and edges to design a heterogeneous mutual attention.
- **CTDNE** [21], **EvolveGCN** [22] and **M^{2}DNE** [17]: They are three typical dynamic homogeneous graph embedding approaches. CTDNE is a skip-gram model based on temporal random walks; EvolveGCN learns the evolution among snapshots by integrating with RNNs to sequentially update convolutional parameters; and M^{2}DNE introduces Hawkes process into modeling the formation process of dynamic graphs where neighbor influence of both source and target nodes are simultaneously extracted.
- **DHNE** [33], **DyHNE** [28] and **DyHATR** [31]: These are three representative temporal heterogeneous graph embedding models. DHNE performs metapath-based random walk between historical snapshots and the current snapshot and design a dynamic heterogeneous skip-gram model to capture representations of nodes; DyHNE splits graphs into several snapshots and employs eigenvalue perturbation to derive the updated embeddings between different snapshots; DyHATR uses hierarchical attention to learn heterogeneous information and incorporates RNNs with temporal attention to capture evolutionary patterns between different snapshots.

Parameter settings. For all methods, we set the embedding dimension $d = 128$, batch size as 1024, learning rate as 0.001, regularization weight $\omega_2 = 0.01$ (if any), and negative sampling size as $k = 5$ (if any). These values give robust performance and are consistent with guidelines from the literature. For HAN, HGT, M^{2}DNE, DyHATR and our HPGE, we respectively limit the size of neighboring candidates to 5, 5 and 10 on the three datasets, using TIS for our method, recency cut-off for M^{2}DNE and random sampling for others. For dynamic homogeneous baselines, we treat events as homogeneous. For Metapath2Vec and DHNE, we sample sequences via A-A, A-A and B-U-B on the three datasets, respectively. The other parameters of all baselines follow their original papers. For our HPGE, we set $\gamma_1 = 0.5$ and $\gamma_2 = 0.5$, $\omega_1 = 1$. In addition, the max iteration is set as 500, 500 and 50 on the three datasets.

5.2 Effectiveness analysis

Node classification. This task is to predict the research area of authors on Aminer and DBLP and the category of businesses on Yelp. The train/test ratio is set to 80%/20%. We run all methods five times and evaluate the average Micro-F1 and Macro-F1 scores.

Table 2: Performance evaluation (with standard deviation) on node classification. The best performance is bolded and the second best is underlined.

Dataset	Aminer		DBLP		Yelp	
Metric	Micro-F1	Macro-F1	Micro-F1	Macro-F1	Micro-F1	Macro-F1
M2V	0.824(0.029)	0.853(0.032)	0.874(0.024)	0.885(0.029)	0.537(0.023)	0.642(0.017)
HEP	0.949(0.016)	0.952(0.013)	0.903(0.022)	0.913(0.018)	0.622(0.012)	0.694(0.009)
HAN	0.967(0.008)	0.970(0.009)	0.912(0.014)	0.914(0.007)	0.621(0.019)	0.691(0.025)
HGT	0.963(0.007)	0.971(0.011)	0.920(0.002)	0.927(0.001)	0.633(0.026)	0.705(0.022)
CTDNE	0.897(0.038)	0.895(0.025)	0.872(0.001)	0.892(0.005)	0.512(0.011)	0.639(0.011)
E.GCN	0.952(0.020)	0.955(0.018)	0.887(0.009)	0.881(0.010)	0.611(0.009)	0.687(0.008)
M2DNE	0.969(0.015)	0.972(0.018)	0.891(0.022)	0.909(0.027)	0.619(0.003)	0.693(0.005)
DHNE	0.901(0.010)	0.913(0.009)	0.888(0.007)	0.909(0.008)	0.578(0.001)	0.665(0.001)
DyHNE	0.970(0.008)	0.978(0.007)	0.922(0.003)	0.922(0.004)	0.622(0.011)	0.721(0.015)
DyHATR	0.973(0.002)	0.969(0.003)	0.933(0.011)	0.935(0.010)	0.627(0.008)	0.717(0.007)
HPGE	**0.988(0.002)**	**0.984(0.003)**	**0.951(0.005)**	**0.952(0.004)**	**0.649(0.010)**	**0.731(0.012)**

As shown in Table 2, our proposed HPGE consistently outperforms all baselines on the three datasets. We make the following observations. (1) Compared with heterogeneous graph embedding approaches (Metapath2vec, HEP, HAN and HGT), HPGE is able to model the temporal dynamics of heterogeneous events. Similarly, compared to dynamic graph embedding approaches (CTDNE, EvolveGCN and M^{2}DNE), HPGE benefits from integrating the abundant semantic information within heterogeneous events. Not surprisingly, the performance gains of HPGE are larger relative to these baselines. (2) Compared with the best competitor DyHATR, which considers both the temporal and heterogeneous information, our HPGE can still achieve substantial improvements. The stable improvements demonstrate that modeling the formation process of DHGs can embed evolving nodes better than just paying attention to the evolution between snapshots. (3) Compared with Aminer and DBLP, our model improves more on Yelp. The potential reason is that Yelp is a larger dataset, such that our temporal importance sampling strategy can benefit more.

Temporal link prediction. This task is to predict the type-r interaction at time t. Given all temporal heterogeneous events before time t and two nodes v_i and v_j. We treat all events at time t as the positive link, and randomly sample 2 negative instances for both v_i and v_j as the negative links. Subsequently, we test all baselines and our HPGE five times and report the average performance of Accuracy, F1 score, and ROC-AUC in Table 3. Obviously, HPGE still achieves the best performance on all datasets. Besides the observations on node classification, HPGE evaluates node proximity based on event types and continuously propagates the influence of types via the temporal point process, while traditional type-wise projections can only model the heterogeneity rather than the interactivity. In addition, HAN, HEP, HGT, DyHNE, DyHATR and our HPGE always performs better than CTDNE, EvolveGCN and M^{2}DNE. This phenomenon indicates that integrating semantics into link formation can benefit temporal link prediction more, compared with simply preserving evolving structures.

Table 3: Performance evaluation on temporal link prediction. The best performance is bolded and the second best is underlined.

Dataset	Aminer			Yelp			DBLP		
Metric	ACC	F1	AUC	ACC	F1	AUC	ACC	F1	AUC
M2V	0.806	0.359	0.759	0.790	0.419	0.702	0.798	0.375	0.656
HEP	0.921	0.814	0.944	0.853	0.566	0.829	0.910	0.753	0.934
HAN	0.923	0.811	0.955	0.855	0.591	0.833	0.903	0.751	0.940
HGT	0.938	0.822	0.963	0.859	0.588	0.833	0.899	0.761	0.941
CTDNE	0.824	0.382	0.763	0.806	0.342	0.635	0.713	0.345	0.653
E.GCN	0.904	0.767	0.922	0.822	0.526	0.785	0.853	0.714	0.905
M2DNE	0.929	0.790	0.951	0.854	0.547	0.818	0.896	0.734	0.939
DHNE	0.875	0.634	0.827	0.831	0.504	0.717	0.821	0.668	0.808
DyHNE	0.928	**0.838**	0.959	0.861	0.592	0.831	0.909	0.767	0.940
DyHATR	0.941	0.832	0.966	0.870	0.598	0.843	0.914	0.773	0.936
HPGE	**0.953**	0.835	**0.976**	**0.873**	**0.603**	**0.850**	**0.938**	**0.793**	**0.957**

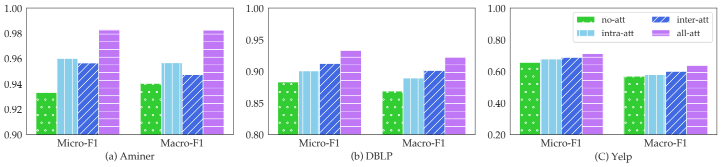
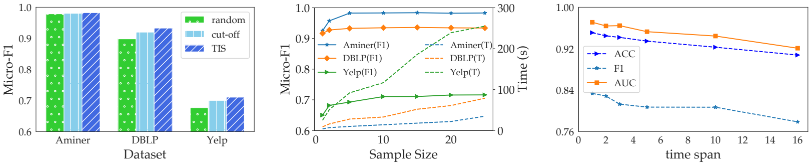


Fig. 3: Effect of hierarchical attention mechanism on node classification.

5.3 Model Analysis

Effect of heterogeneous evolved attention mechanism. We further discuss the effect of heterogeneous evolved attention mechanism by comparing with three model variants including no attention (no-att), intra-type temporal attention (intra-att) and inter-type temporal attention (inter-att), as well as HPGE (all-att). The results for the node classification task are shown in Fig. 3. We observe the following. (1) Simultaneously modeling both intra- and inter-type temporal attention achieves the most improvements, while the no-attention variant performs the worst on all datasets. (2) Compared with the intra-attention variant, HPGE has the ability to evaluate the importance of influence of different types of historical events to current type of interactions. Meanwhile, HPGE can filter the neighborhoods via intra-typed attention, compared with the inter-typed variant. These observations demonstrate the effectiveness of our heterogeneous evolved attention mechanism.

Efficacy of temporal importance sampling. The other key design is our temporal importance sampling (TIS), which considers both structural importance and time decay. We analyze the effectiveness of TIS by comparing with the often used random sampling and recency-based cut-off, as well as the efficiency of TIS under the effective sample size. (1) Comparison of sampling strategies. Fig. 4(a) reports the Micro-F1 scores of different sampling strategies for the node classification task. Notice that the sample size is set as 5, 5 and 10 for all strategies on the three datasets, respectively. Among the three sam-



(a) sampling strategies (b) effective sample size (c) varying the dynamics

Fig. 4: Efficacy of TIS and the ability of evolution modeling.

pling strategies, it is clear that our TIS strategy performs the best, especially on the larger datasets DBLP and Yelp. The results are intuitive since the cut-off strategy ignores structurally important neighbors, while the random sampling, which performs the worst, pays no attention to either structure or dynamics. (2) Effective sample size. Effective sample size plays an important role in sampling to achieve the balance between effectiveness and efficiency. As shown in Fig. 4(b), we increase the sample size from 5 to 25 and showcase both the Micro-F1 score (solid lines) and time cost (dotted lines). A larger sample size gradually increases Micro-F1, which converges quickly around 5 or 10. Here 5 or 10 is the effective samples size, which is much smaller than the full neighborhoods. In particular, when using a larger sample size (e.g., 25 or even the full size), the time cost becomes unbearable.

Ability of modeling evolution As the dynamics of graphs are in the form of timestamps, we "coarsen" the timestamps by considering time spans of varying size. In Fig. 4(c), on the Aminer dataset, we vary the size of time span from every 1 year (i.e., finest time units) to 16 year (i.e., the entire graph consists of a single time span of 16 years, which effectively become a static graph) , and showcase the performance on temporal link prediction. The performance of HPGE consistently degrades with the increasing size of time span, indicating that modeling evolving dynamics with finer granularity (i.e., smaller time span) lead to better performance. Notice that when the time span is 16, the graph becomes a static graph and our HPGE also degrades to a static model. Overall, the results further illustrate the effectiveness of HPGE in handling evolution.

6 Conclusion

In this paper, we propose the HPGE model which introduces Hawkes process to handle the challenging dynamic heterogeneous graph embedding problem. Focusing on modeling the formation process of temporal heterogeneous events, we respectively design the heterogeneous conditional intensity function to capture the excitation from historical multiple events, the heterogeneous evolved attention mechanism to learn fine-grained representations considering both intra- and inter-typed temporal influences. HPGE hinges on a novel temporal importance sampling strategy, to enable efficient extraction of representative events. Experimental results on three public datasets demonstrate that HPGE outperforms the alternatives on fundamental graph tasks.

Acknowledgements

This work is supported in part by the National Natural Science Foundation of China (No. U20B2045, 61772082, 62002029). This research is also supported by the Agency for Science, Technology and Research (A*STAR) under its AME Programmatic Funds (Grant No. A20H6b0151).

References

1. Bian, R., Koh, Y.S., Dobbie, G., Divoli, A.: Network embedding and change modeling in dynamic heterogeneous networks. In: SIGIR. pp. 861–864 (2019)
2. Cai, H., Zheng, V.W., Chang, K.C.: A comprehensive survey of graph embedding: Problems, techniques, and applications. IEEE TKDE **30**(9), 1616–1637 (2018)
3. Cen, Y., Zou, X., Zhang, J., Yang, H., Zhou, J., Tang, J.: Representation learning for attributed multiplex heterogeneous network. In: ACM SIGKDD. pp. 1358–1368 (2019)
4. Chen, J., Ma, T., Xiao, C.: Fastgcn: Fast learning with graph convolutional networks via importance sampling. In: ICLR (2018)
5. Dong, Y., Chawla, N.V., Swami, A.: metapath2vec: Scalable representation learning for heterogeneous networks. In: ACM SIGKDD. pp. 135–144 (2017)
6. Fu, T., Lee, W., Lei, Z.: Hin2vec: Explore meta-paths in heterogeneous information networks for representation learning. In: CIKM. pp. 1797–1806 (2017)
7. Fu, X., Zhang, J., Meng, Z., King, I.: MAGNN: metapath aggregated graph neural network for heterogeneous graph embedding. In: WWW. pp. 2331–2341 (2020)
8. Goyal, P., Kamra, N., He, X., Liu, Y.: Dyngem: Deep embedding method for dynamic graphs. CoRR **abs/1805.11273** (2018)
9. Grover, A., Leskovec, J.: node2vec: Scalable feature learning for networks. In: SIGKDD. pp. 855–864 (2016)
10. Hamilton, W.L., Ying, R., Leskovec, J.: Representation learning on graphs: Methods and applications. IEEE Data Engineering Bulletin **40**(3), 52–74 (2017)
11. Hamilton, W.L., Ying, Z., Leskovec, J.: Inductive representation learning on large graphs. In: NeuIPS. pp. 1024–1034 (2017)
12. Hu, Z., Dong, Y., Wang, K., Sun, Y.: Heterogeneous graph transformer. In: WWW. pp. 2704–2710 (2020)
13. Ji, Y., Yin, M., Fang, Y., Yang, H., Wang, X., Jia, T., Shi, C.: Temporal heterogeneous interaction graph embedding for next-item recommendation. In: ECML-PKDD (2020)
14. Ji, Y., Yin, M., Yang, H., Zhou, J., Zheng, V.W., Shi, C., Fang, Y.: Accelerating large-scale heterogeneous interaction graph embedding learning via importance sampling. ACM TKDD **15**(1), 1–23 (2020)
15. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. In: ICLR (2015)
16. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. In: ICLR (2017)

17. Lu, Y., Wang, X., Shi, C., Yu, P.S., Ye, Y.: Temporal network embedding with micro- and macro-dynamics. In: CIKM. pp. 469–478 (2019)
18. Luo, W., Zhang, H., Yang, X., Bo, L., Yang, X., Li, Z., Qie, X., Ye, J.: Dynamic heterogeneous graph neural network for real-time event prediction. In: ACM SIGKDD. pp. 3213–3223 (2020)
19. Ma, Y., Guo, Z., Ren, Z., Tang, J., Yin, D.: Streaming graph neural networks. In: SIGIR. pp. 719–728 (2020)
20. Manessi, F., Rozza, A., Manzo, M.: Dynamic graph convolutional networks. Pattern Recognit. **97** (2020)
21. Nguyen, G.H., Lee, J.B., Rossi, R.A., Ahmed, N.K., Koh, E., Kim, S.: Continuous-time dynamic network embeddings. In: WWW. pp. 969–976 (2018)
22. Pareja, A., Domeniconi, G., Chen, J., Ma, T., Suzumura, T., Kanezashi, H., Kaler, T., Schardl, T., Leiserson, C.: Evolvegcn: Evolving graph convolutional networks for dynamic graphs. In: AAAI. vol. 34, pp. 5363–5370 (2020)
23. Perozzi, B., Al-Rfou, R., Skiena, S.: Deepwalk: online learning of social representations. In: ACM SIGKDD. pp. 701–710 (2014)
24. Sun, Y., Han, J., Yan, X., Yu, P.S., Wu, T.: Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. VLDB **4**(11), 992–1003 (2011)
25. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. In: ICLR (2018)
26. Wang, X., Bo, D., Shi, C., Fan, S., Ye, Y., Yu, P.S.: A survey on heterogeneous graph embedding: Methods, techniques, applications and sources. arXiv preprint arXiv:2011.14867 (2020)
27. Wang, X., Ji, H., Shi, C., Wang, B., Ye, Y., Cui, P., Yu, P.S.: Heterogeneous graph attention network. In: WWW. pp. 2022–2032 (2019)
28. Wang, X., Lu, Y., Shi, C., Wang, R., Cui, P., Mou, S.: Dynamic heterogeneous information network embedding with meta-path based proximity. TKDE (2020)
29. Wu, W., Liu, H., Zhang, X., Liu, Y., Zha, H.: Modeling event propagation via graph biased temporal point process. IEEE TNNLS (2020)
30. Xu, D., Ruan, C., Körpeoglu, E., Kumar, S., Achan, K.: Inductive representation learning on temporal graphs. In: ICLR (2020)
31. Xue, H., Yang, L., Jiang, W., Wei, Y., Hu, Y., Lin, Y.: Modeling dynamic heterogeneous network for link prediction using hierarchical attention with temporal rnn. arXiv preprint arXiv:2004.01024 (2020)
32. Yang, L., Xiao, Z., Jiang, W., Wei, Y., Hu, Y., Wang, H.: Dynamic heterogeneous graph embedding using hierarchical attentions. In: ECIR. Lecture Notes in Computer Science, vol. 12036, pp. 425–432 (2020)
33. Yin, Y., Ji, L.X., Zhang, J.P., Pei, Y.L.: Dhne: Network representation learning method for dynamic heterogeneous networks. IEEE Access **7**, 134782–134792 (2019)
34. Zhao, J., Wang, X., Shi, C., Hu, B., Song, G., Ye, Y.: Heterogeneous graph structure learning for graph neural networks. In: AAAI (2021)
35. Zheng, V.W., Sha, M., Li, Y., Yang, H., Fang, Y., Zhang, Z., Tan, K., Chang, K.C.: Heterogeneous embedding propagation for large-scale e-commerce user alignment. In: ICDM. pp. 1434–1439 (2018)
36. Zuo, Y., Liu, G., Lin, H., Guo, J., Hu, X., Wu, J.: Embedding temporal network via neighborhood formation. In: ACM SIGKDD. pp. 2857–2866 (2018)