Motif-Preserving Temporal Network Embedding

Hong Huang[1,2,3] , **Zixuan Fang**[1,2,3] , **Xiao Wang**[4*] , **Youshan Miao**[5] and **Hai Jin**[1,2,3]
[1]National Engineering Research Center for Big Data Technology and System
[2]Service Computing Technology and Systems Laboratory
[3]School of Computer Science and Technology, Huazhong University of Science and Technology, China
[4]Beijing University of Posts and Telecommunications, China
[5]Microsoft Research Asia
{honghuang, zixuan, hjin}@hust.edu.cn, xiaowang@bupt.edu.cn, yomia@microsoft.com

Abstract

Network embedding, mapping nodes in a network to a low-dimensional space, achieves powerful performance. An increasing number of works focus on static network embedding, however, seldom attention has been paid to temporal network embedding, especially without considering the effect of mesoscopic dynamics when the network evolves. In light of this, we concentrate on a particular motif — triad — and its temporal dynamics, to study the temporal network embedding. Specifically, we propose *MTNE*, a novel embedding model for temporal networks. MTNE not only integrates the Hawkes process to stimulate the triad evolution process that preserves motif-aware high-order proximities, but also combines attention mechanism to distinguish the importance of different types of triads better. Experiments on various real-world temporal networks demonstrate that, compared with several state-of-the-art methods, our model achieves the best performance in both static and dynamic tasks, including node classification, link prediction, and link recommendation.

1 Introduction

Recently academic and industry have both witnessed rapid development of network embedding, which maps the nodes into a low-dimensional space while preserving certain proximities among nodes, features, or structures. Due to its powerful performance, network representation learning, namely network embedding, has been widely applied to various network-related tasks, such as link prediction, node clustering, and node classification.

Inspired by word embedding [Mikolov *et al.*, 2013] in language processing, Deepwalk [Perozzi *et al.*, 2014] is proposed to embed nodes in the networks. Furthermore, LINE [Tang *et al.*, 2015] preserves first- and second-proximities, GraRep [Cao *et al.*, 2015] preserves k-order proximities, Node2vec [Grover and Leskovec, 2016] preserves neighborhood proximity, and MNMF [Wang *et al.*, 2017] preserves communities. All these methods achieve good performance in various network-related tasks.

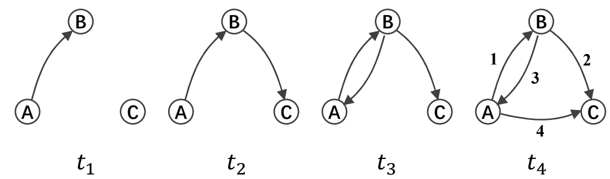


Figure 1: A toy example of network evolution

However, the methods mentioned above are all proposed to deal with static networks, while most networks exhibit complex temporal properties, which means nodes and edges always evolve over the time. For example, in an information diffusion network, information interactions between users shape the typology of temporal network and influence the network evolution. As shown in Figure 1, there are a group of three users in an information diffusion network. At time t_1, when B receives A's information, an edge from A to B establishes. An edge forms once there is an action associated with information diffusion happens. If we consider a static network at time t_4, we only know that there is a closed triad in which A and B send messages mutually, and C receives from B and A, respectively. However, if we capture its temporal properties, more information will be preserved that B first receives from A, which indicates that A has a higher social power according to the social status theory [Waters, 2015]. Therefore, evolution dynamics and properties need to be well examined and preserved for temporal networks.

In literature, there are only a few works focus on temporal network embedding. For instance, HTNE [Zuo *et al.*, 2018] considers the temporal network embedding via neighborhood formation process. M^{2}DNE [Lu *et al.*, 2019] studies micro- and macro-dynamics by taking edge dynamics and network evolution patterns into account. The former method only considers the influence of one node's neighbor formation sequence on the current neighbor. Nevertheless the establishment of a relationship between two nodes is an interacted process, and the neighbor formation sequence cannot reflect the law of network evolution very well. Besides from edge level, the latter takes the whole network's evolution into account. However, except the consideration of micro- and macro-dynamics, the influence from a mesoscopic view for temporal networks has been ignored.

Meso-dynamics has been widely used in mining complex

*Contact Author

networks, as well as social networks. It is crucial to understand the network structure and function of social systems [Huang *et al.*, 2018]. Unlike micro-dynamics focusing on node and edge level, meso-dynamics considers the coupling effects of a small group of nodes and edges. Besides, it exhibits group properties with only a few nodes and edges, thus preserving group-aware high-order proximity. Usually, meso-dynamics is in the form of subgraph evolution, where subgraph patterns are also called network motifs. Since network motifs have been well studied, meso-dynamics can make full use of their correlations.

In this paper, we propose a novel temporal network embedding method with the consideration of meso-dynamics, named MTNE. In view of meso-dynamics, we focus on the simplest and fundamental motif — triad — in the networks. Specially, we first incorporate the triad dynamics into the Hawkes process. So that, our model can well capture the effects of past events of users inside a triad. Moreover, we adopt attention mechanism to specify the importance of different types of triads. Note that our method can be easily applied to model other motifs due to the generality of the Hawkes process. We test our model on several real-world datasets. The experimental results show that our proposed MTNE outperforms state-of-the-art baselines in both static and dynamic tasks.

The major contributions of our work are as follows:

- To our knowledge, we are the first to concentrate on temporal network embedding with meso-dynamics that preserves motif-aware high-order proximity.
- We propose a novel model (MTNE), which leverages Hawkes process to model motif evolution and design an attention mechanism to model the importance of motif structures.
- Experiment results on five real-world datasets show that our MTNE has better performance than several state-of-the-art baselines in both static and dynamic tasks.

2 Preliminaries

In a temporal network, each edge indicates an event that happens between two nodes at a time point. Formally, given a temporal network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$, where $\mathcal{V}$ and $\mathcal{E}$ indicate nodes and edges of the temporal network and $\mathcal{T}$ is the set of timestamps. A temporal edge $e_{ij}^t \in \mathcal{E}$ represents an event that is initiated by node v_i together with node v_j at time t.

Note that since node v_i and node v_j may arise multiple events at different time t, temporal edge e_{ij}^t represents distinct events while edge e_{ij} is only a static edge between node v_i and node v_j, and can be either directed or undirected. For example, in one mobile communication network, we can construct a static network based on the relationship among users. On the other hand, we can also build a temporal network to describe the interactions among users. In the temporal network, each temporal edge means a phone call between two users at a certain time. Once user v_i calls user v_j at time t, then a temporal edge e_{ij}^t establishes.

In this paper, we start with the simplest and fundamental motif – triad, and introduce the definition of temporal triad.

Definition 1: ***Temporal Triad.*** Given three nodes $\Delta = (v_i, v_j, v_k)$, if there are at least two temporal edges that involve the three nodes, —e.g., $\exists e_{ij}^{t_1}, e_{jk}^{t_2} \in \mathcal{E}$ — then we call Δ a *temporal triad*. Specially, if there exists at least one temporal edge for any two nodes in a temporal triad, we call Δ a *temporal closed triad*. If there exist two nodes in a temporal triad without any temporal edges, then we call Δ a *temporal open triad*.

Network evolution is, in some sense, driven by the triadic closure process [Huang *et al.*, 2015]. From a microscopic view, the triadic closure process describes how an open triad forms a closed triad. Given an open triad (v_i, v_j, v_k) at time t_1, where v_i and v_j do not know each other but they have a common friend v_k. Now, if v_k decides to introduce v_i and v_j and lets them know each other, there will be a connection between v_i and v_j and the open triad (v_i, v_j, v_k) will become closed at time t_2. Nevertheless, there is a problem that if there are multiple open triads which consist of v_i and v_j before time t_1, we cannot accurately infer from the dataset which open triad determines the connection between v_i and v_j. We call all these temporal open triads as *candidate temporal triads*.

In this paper, we aim to represent the nodes in the temporal networks incorporating the influence of all candidate temporal triads. We formally define our problem as below.

Problem. ***Motif-Preserving Temporal Network Embedding.*** Given a temporal network $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{T})$, during network evolution, we generate all candidate temporal triads for every node pair in $\mathcal{V}$. Then, motif-preserving temporal network embedding aims to learn a mapping function $f : \mathcal{V} \mapsto \mathbb{R}^d$, where d is a positive integer indicating the number of embedding dimensions and $d \ll |V|$. The objective of the function f is to not only preserve the feature of the network structure but also capture the influence of motif-based network evolution.

3 The Proposed Model

3.1 Model Overview

In this section, we propose *MTNE*, a novel model that is capable of learning desirable representations for nodes in temporal networks and simultaneously preserve the characteristics of motifs in the network. As illustrated in Figure 2, node pairs (i, j) and (p, q) belong to different open triads, respectively. Considered their different past temporal information, even if they have the same property, the generated embeddings for them in the new dimension would be different. Our proposed MTNE would capture such precise and diverse structural information – triad motif evolution property. Specifically, MTNE leverages Hawkes process to model motif evolution process. Then we can learn it using optimization techniques like *stochastic gradient descent* (SGD).

3.2 Triad Motif Evolution Modeling

As we know, the triad motif evolution process is influenced by the historical triad evolution events. Therefore, the Hawkes process [Hawkes, 1971] can be used to model triad motif evolution. We define the conditional intensity function as

$$\tilde{\lambda}_{x,y}(t) = \mu_{x,y} + \sum_{t_h < t} \sigma_{\Delta_h}(t), \tag{1}$$

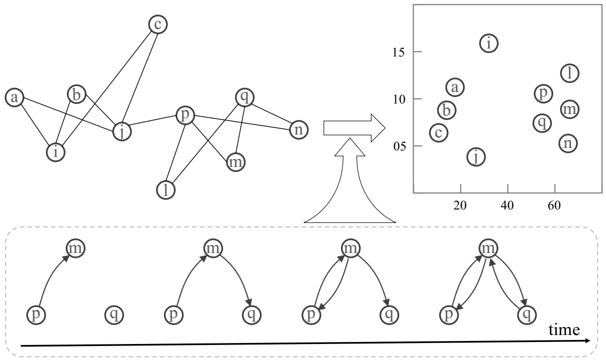


Figure 2: Illustration of MTNE

where $\tilde{\lambda}_{x,y}(t)$ is the intensity of the new emerged temporal edge e_{xy} at time t, which also means the formation of certain triad that contains v_x and v_y at time t; $\mu_{x,y}$ denotes the base rate of establishing a connection between v_x and v_y, while Δ_h is the candidate open triads that contains node v_x and v_y prior to time t. $\sigma_{\Delta_h}(t)$ is the kernel function that represents the influence of history, which will decay over time. According to Eq. (1), we model the triad evolution process by modeling the base rate and past influence, respectively.

Base rate. Suppose the base rate between nodes will be high if they are similar to each other. So we can model the base rate as a function that measures the similarity between nodes. For brevity, we calculate the similarity using negative squared Euclidean distance. Specifically, suppose the node embeddings of node v_x and v_y are u_x and u_y respectively, then the base rate can be defined as

$$\mu_{x,y} = similarity(v_x, v_y) = -||u_x - u_y||^2. \qquad (2)$$

Influence of open triads. Likely, similarity function can also help us to model the influence of historical open triad motifs, say $\sum_{t_h<t} \sigma_{\Delta_h}(t)$.

Specifically, we define the influence of one open triad (v_x, v_m, v_y) as

$$\sigma_{\Delta_h}(t) = \eta_{x,m} + \eta_{y,m}, \qquad (3)$$

where v_m is the middle node of the open triad, $\eta_{x,m}$ and $\eta_{y,m}$ represent the strength of the internal relationship of the two connected node pairs in the open triad respectively. Note that the relationship between nodes can be established repeatedly in some temporary networks (e.g., mobile communication network, co-author network, etc.), $\eta_{x,m}$ is calculated as the sum of a series of connection effects between v_x and v_m, i.e.,

$$\eta_{x,m} = f(u_x, u_m) \times \sum_{t_{x,m}<t} \kappa(t - t_{x,m}), \qquad (4)$$

where $f(u_x, u_m)$ is the similarity function between node v_x and v_m, i.e., $f(u_x, u_m) = -||u_x - u_m||^2$. $\kappa(\cdot)$ is the memory kernel function, which denotes the time decay effect. Here we have $\kappa(t - t_{x,m}) = e^{(-\delta_x(t - t_{x,m}))}$, where δ is a discount rate that controls time decay effects. Note that δ is a node dependent parameter, which means that for each node, the history open triad can influence the current closure with different intensity. It can be learned during training.

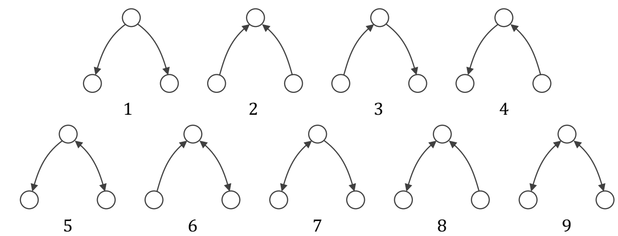


Figure 3: Open triad motif

Then we can get the influence of one open triad as

$$\begin{aligned} \sigma_{\Delta_h}(t) = & f(u_x, u_m) \times \sum_{t_{x,m}<t} \kappa(t - t_{x,m}) \\ & + f(u_y, u_m) \times \sum_{t_{y,m}<t} \kappa(t - t_{y,m}). \end{aligned} \qquad (5)$$

The value of the influence obtained in the above calculations are all negative values since the similarity is measured by negative Euclidean distance. However, the conditional intensity function should return a positive value. So we transfer $\tilde{\lambda}_{x,y}(t)$ to a positive real number by an exponential function, *i.e*, $\lambda_{x,y}(t) = exp(\tilde{\lambda}_{x,y}(t))$, and its value range is between 0 and 1, which accords with the range of the motif formation probability.

Influence of triad motifs. The influence of historical events is defined as the sum of historical open triad motifs' effects. Intuitively, different types of open triad would have different closing probabilities. Then, as shown in Figure 3, there exist nine types of open triad motifs considered its edge direction. According to [Huang *et al.*, 2015], among all types of open triads, the open triad motif No.9 has the highest probability to become closed, of which both edges are bi-directional. Besides, the closure probability of No.2 is the lowest due to the phenomenon of assembled fans in social networks. Therefore, it is necessary to consider the motif evolution dynamics, especially the closure process of the open triad motifs when modeling σ_{Δ_h}, which makes up conditional intensity function.

Inspired by attention mechanism that used in neural machine translation [Bahdanau *et al.*, 2015], we define the weight of the n-th open triad motif using a softmax function as follows:

$$w_{\Delta_n} = \frac{e^{s_n}}{\sum_n e^{s_n}}, \qquad (6)$$

where s_n indicates the influence of n-th type of triad motif in Figure 3. It will be also learned and updated during training. Therefore, the conditional intensity function can be re-formulated as,

$$\tilde{\lambda}_{x,y}(t) = \mu_{x,y} + \sum_{t_h<t} w_{\Delta_h} \sigma_{\Delta_h}(t), \qquad (7)$$

where w_{Δ_h} calculates the influence of the type of triad motif that Δ_h belongs to.

Loss function. By modeling triad evolution with Hawkes process, we can infer the edge evolution with the consideration of candidate temporal triads. That is, given a node pair (v_x, v_y) before time t, considering candidate temporal triads $H_\Delta(t)$, the probability of forming connection between v_x and

datasets	nodes	static edges	temporal edges	time steps	labels
school	178	9,846	18,648	331	3
digg	1,832	16,538	45,727	79	0
mobile	11,569	211,759	406,802	67	0
weibo	76,408	335,970	726,670	23	4
dblp	28,085	162,451	236,894	27	10

Table 1: Dataset description

v_y can be inferred by the conditional intensity as,

$$p(v_x, v_y|H_\Delta(t)) = \frac{\lambda_{x,y}(t)}{\sum_{y'} \lambda_{x,y'}(t)}, \tag{8}$$

where y' represents all nodes except v_x in the network.

After taking a log function, the likelihood of edge formation for all node pairs in the network can be written as,

$$logL = \sum_{(v_x,v_y)\in E} logp(v_x, v_y|H_\Delta(t)). \tag{9}$$

Since we have transferred the conditional intensity to a positive number by an exponential function, $p(v_x, v_y|H_\Delta(t))$ is actually a softmax unit. Besides, in order to avoid to summarize the entire set of nodes when calculating Eq. (9), we use negative sampling techniques as in [Zuo *et al.*, 2018]. Then the loss function of the connection between a node pair (v_x, v_y) at time t can be computed as follows,

$$-log\sigma(\tilde{\lambda}_{x,y}(t)) - \sum_{k=1}^{K} E_{v^k \sim P_n(v)}[log\sigma(-\tilde{\lambda}_{x,k}(t))], \tag{10}$$

where K is the number of negative nodes sampled according to the degree distribution $P_n(v) \propto d_v^{3/4}$, where d_v is the degree of node v, and $\sigma(x) = 1/(1 + e^{-x})$ is the sigmoid function.

In experiments, the number of candidate temporal triads will affect computational cost. Thus we fix the maximum number of candidate temporal triads in this paper and discuss its influence in section 4.3.

Finally, we use *Stochastic Gradient Descent* (SGD) to optimize the loss function. After converging, we can get the learned node embeddings.

4 Experiments and Discussions

4.1 Experimental Setup

Datasets. We test MTNE on five different real-world datasets, say School [Fournet and Barrat, 2014], DBLP [Ley, 2009], Digg [Hogg and Lerman, 2012], Mobile [Huang *et al.*, 2018], and Weibo [Zhang *et al.*, 2013]. Their statistics is listed in Table 1.

Baselines. We compare the performance of MTNE against the following seven network embedding methods, including four static network embedding methods and four temporal network embedding methods, shown in Table 2.

Parameter settings. For all methods, the embedding dimension d is set as 64. For our proposed MTNE, the batch size, the learning rate of the SGD, the number of candidate temporal triads, and the number of negative samples are set to be 1000, 0.003, 5, 5, respectively, while for other baselines, we use the default parameters settings. For each experiment, we repeat 10 times and report the average value as the final results.

Methods	Temporal	Remark
Deepwalk [Perozzi *et al.*, 2014]	×	-
LINE [Tang *et al.*, 2015]	×	2-order
GraRep [Cao *et al.*, 2015]	×	k-order
Noe2Vec [Grover and Leskovec, 2016]	×	-
TNE [Zhu *et al.*, 2016]	✓	Snapshot
DynamicTriad [Zhou *et al.*, 2018]	✓	Snapshot
HTNE [Zuo *et al.*, 2018]	✓	Evolution
M^{2}DNE [Lu *et al.*, 2019]	✓	Evolution

Table 2: Baselines

4.2 Experiment Performance

We validate our proposed model from two aspects: tasks for static networks and temporal networks. For the former, we first learn node embeddings and then treat them as features. We test two traditional tasks here: node classification and link prediction, and use precision, recall, and F1 as the measures. For the latter, we perform a temporal recommendation task and use precision and recall as measures.

Node classification. We train a logistic regression classifier using the learned node embeddings as features to predict node labels. We use School, Weibo, and DBLP datasets, and report the results in Table 3.

From the results, we can see that our method MTNE performs better than all baselines. Specifically, compared with methods for static network embedding (i.e., DeepWalk, LINE, GraRep, and Node2vec), the better performance of MTNE suggests that the evolutionary dynamics are of great importance for node classification. On the other hand, compared with methods for temporal network embedding (i.e., TNE, DynamicTriad, HTNE, and M^{2}DNE), our method captures motif-aware high-order proximity by modeling motif evolution process, which increases the semantic meaning of the embeddings. Specifically, MTNE encodes motif features in the latent space and the nodes in the same motif are more likely to be in the same class, which further improves the performance of classification.

Link prediction. For the task of link prediction, we aim to predict whether there is an edge between the given node pair (v_x, v_y). We utilize $|u_x - u_y|$ as the feature to train a Logistic Regression classifier, where u_x and u_y are embeddings of v_x and v_y respectively. In each dataset, we randomly choose $10,000$ edges as positive samples and $10,000$ unconnected node pairs as negative samples. The experiment results are shown in Table 4.

From the table, we can see that our method performs the best on all datasets, which again proves the effectiveness of our method. We believe the significant improvement is because that our method captures network structure features more accurately by modeling the motif evolution process, which is an essential feature for real network evolution, as discussed in [Huang *et al.*, 2015].

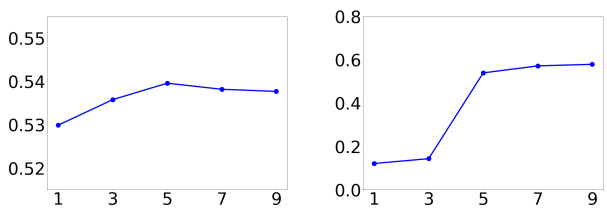
Temporal recommendation. We study the effectiveness of MTNE for capturing temporal information in networks with the task of temporal recommendation. Specifically, given a testing timestamp t, we obtain the node embeddings in the network before time t and recommend possible new connections for the testing node after time t. We conduct experi-

Datasets	Metrics	DeepWalk	LINE	GraRep	Node2vec	TNE	DynamicTriad	HTNE	M^{2}DNE	MTNE
school	precision	92.40%	47.17%	86.58%	88.25%	89.95%	93.29%	95.59%	95.11%	**96.53%**
	recall	89.44%	40.37%	77.32%	79.50%	87.23%	92.35%	93.78%	93.66%	**94.59%**
	f1	91.30%	42.10%	79.47%	80.99%	89.45%	92.61%	94.43%	94.10%	**95.36%**
weibo	precision	44.39%	40.77%	43.19%	44.80%	42.14%	44.93%	44.65%	44.97%	**45.20%**
	recall	47.73%	47.24%	47.46%	47.37%	47.28%	47.29%	47.23%	47.38%	**48.17%**
	f1	38.82%	34.68%	37.38%	37.95%	35.32%	37.26%	33.22%	35.27%	**39.38%**
dblp	precision	53.80%	35.00%	54.41%	55.23%	45.57%	54.88%	55.67%	52.85%	**57.65%**
	recall	54.12%	33.74%	53.15%	53.49%	44.63%	53.92%	54.15%	50.09%	**55.83%**
	f1	53.34%	30.19%	52.52%	52.01%	44.28%	53.50%	52.48%	48.41%	**53.96%**

Table 3: Performance on node classification

Methods	School			Digg			Mobile			weibo		
	precision	recall	f1	precision	recall	f1	precision	recall	f1	precision	recall	f1
Deepwalk	81.60%	81.18%	81.21%	69.01%	68.75%	68.65%	72.35%	72.17%	72.03%	71.66%	71.40%	71.31%
LINE	70.29%	69.72%	69.64%	72.49%	72.29%	72.23%	67.83%	67.78%	67.76%	80.96%	80.91%	80.90%
GraRep	71.82%	69.77%	69.93%	72.28%	72.15%	72.04%	69.19%	68.93%	68.44%	80.31%	80.42%	80.11%
Node2vec	80.38%	80.03%	80.06%	70.37%	70.34%	70.32%	71.00%	70.96%	70.95%	76.21%	76.09%	76.08%
TNE	78.23%	77.96%	78.01%	70.15%	69.24%	69.18%	69.47%	69.10%	68.89%	81.38%	81.02%	80.97%
DynamicTriad	79.23%	79.15%	79.17%	70.55%	70.37%	70.31%	70.52%	69.71%	69.59%	83.82%	83.57%	83.45%
HTNE	80.35%	80.21%	80.25%	65.58%	65.05%	65.00%	67.18%	66.85%	66.69%	84.64%	84.39%	84.36%
M^{2}DNE	81.64%	81.20%	81.25%	72.92%	72.78%	72.75%	72.45%	72.25%	72.18%	83.88%	83.40%	83.34%
MTNE	**82.37%**	**82.12%**	**82.16%**	**75.36%**	**74.73%**	**75.00%**	**74.56%**	**74.26%**	**74.18%**	**85.75%**	**85.66%**	**85.42%**

Table 4: Performance on link prediction



(a) (b)

Figure 4: Impacts of candidate temporal triads and negative samples on *dblp*. The X-axis represents the number of of candidate temporal triads (a) and negative samples (b), and the Y-axis represents the Macro-F1 of node classification.

ments on School, Digg, and Mobile, and use data from the first 80% of the period as a training set the rest as a testing set.

For each test node v_i in the network, we predict the top-10 possible new connections for v_i after t. We use the negative squared Euclidean distance between node embeddings as the ranking score, and obtain the top-10 nodes with the highest score as our predicted nodes. The experimental results are reported in Table 5 with respect to recall and precision.

From the results, we can see that MTNE outperforms all the baselines. The improvement confirms that the triad motif evolution process proposed in MTNE captures the evolution patterns of the temporal network to some extent. Also, the dynamic methods (i.e., TNE, DynamicTriad, HTNE, M^{2}DNE, and MTNE) perform better than the static methods (i.e., DeepWalk, LINE, GraRep, and Node2vec), which demonstrates the importance of temporal information in the network. In the dynamic methods, TNE and DynamicTriad perform relatively worse, which might because they only model the temporal information with network snapshots and do not consider the entire evolution process. Although HTNE and M^{2}DNE consider the evolution process, they ignore the motif evolution process in the temporal network, which is one of the critical characteristics of network evolution.

4.3 Parameter Analysis

We then study the influence of some important parameters, that is, the number of candidate temporal triads, the number of negative samples, and the number of recommended nodes.

The number of candidate temporal triads. The number of candidate temporal triads is designed to preserve the time effects of triads and reduce computational cost. From Figure 4 (a), we can see that the Macro-F1 first increases along with the number of candidate temporal triads h when $h < 5$. Then, the Macro-F1 begins to drop. Therefore, for a balance between effectiveness and efficiency, we set the number to 5 in our default experiment settings.

The number of negative sample. We then test how the number of negative samples influences the performance. The results are shown in Figure 4 (b). From the figure, we can see that Macro-F1 is very low when the number of negative sample is less than 5, and then increases slowly after 5.

The number of recommended nodes. We now test the influence of the number of recommended nodes in the task of temporal recommendation. Take Mobile data as an example, and we test the results with top-5, 10, 15, 20 nodes, respectively. The results are shown in Table 6.

5 Related Work

We discuss related work from two parts: static network embedding and temporal network embedding.

Methods	School		Digg		Mobile		dblp	
	precision	recall	precision	recall	precision	recall	precision	recall
Deepwalk	8.35%	10.3%	4.54%	3.92%	15.88%	13.38%	7.07%	15.36%
LINE	7.24%	8.94%	4.78%	3.67%	6.07%	5.17%	3.74%	8.11%
GraRep	8.29%	10.21%	4.74%	3.57%	10.77%	8.92%	5.13%	11.28%
Node2vec	10.16%	12.54%	3.66%	3.16%	13.22%	11.26%	6.25%	13.56%
TNE	13.19%	15.84%	4.32%	3.6%	14.57%	12.41%	6.84%	14.86%
DynamicTriad	12.58%	15.37%	4.31%	3.54%	14.31%	12.25%	6.49%	14.16%
HTNE	14.48%	18.57%	3.93%	3.39%	13.36%	11.38%	7.49%	16.25%
M^2DNE	15.27%	18.85%	4.35%	3.76%	15.34%	13.06%	6.67%	14.47%
MTNE	**16.06%**	**19.82%**	**5.66%**	**4.37%**	**16.23%**	**13.83%**	**8.12%**	**17.17%**

Table 5: Performance on temporal recommendation

Methods	top@5		top@10		top@15		top@20	
	precision	recall	precision	recall	precision	recall	precision	recall
Deepwalk	20.3%	8.5%	15.88%	13.38%	11.69%	15.49%	9.54%	16.66%
LINE	7.81%	3.32%	6.07%	5.17%	5.06%	6.47%	4.36%	7.43%
GraRep	11.26%	4.58%	10.77%	8.92%	8.10%	11.25%	5.93%	10.51%
Node2vec	19.36%	8.25%	13.22%	11.26%	10.35%	13.22%	8.56%	14.58%
TNE	20.2%	8.24%	14.57%	12.41%	10.65%	13.73%	8.39%	13.96%
DynamicTriad	20.45%	8.87%	14.31%	12.25%	10.46%	13.91%	8.47%	14.5%
HTNE	19.07%	8.12%	13.36%	11.38%	10.52%	13.44%	8.74%	14.88%
M^2DNE	21.97%	9.35%	15.34%	13.06%	12.05%	15.39%	10.02%	17.07%
MTNE	**23.23%**	**9.89%**	**16.23%**	**13.83%**	**12.67%**	**16.2%**	**10.52%**	**17.92%**

Table 6: The influence of number of recommended nodes on *Mobile*

Static network embedding. Since Deepwalk [Perozzi *et al.*, 2014] first represents the nodes in networks, a significant amount of progress have been made in network representation learning. In order to represent static networks, researchers focus on various network information, aiming to preserve the proximity of neighbor nodes or high-order proximities among nodes. For example, Node2vec [Grover and Leskovec, 2016] preserves neighbor nodes, Line [Tang *et al.*, 2015] preserves first- and second-order proximities, MNMF [Wang *et al.*, 2017] preserves communities, and GraRep [Cao *et al.*, 2015] preserves k-order proximities, RUM [Yu *et al.*, 2019] preserves motifs. More related work can refer to [Cui *et al.*, 2018].

Temporal network embedding. There are also several attempts towards dynamic network embedding, especially temporal network embedding. In dealing with temporal information, two solutions are usually used. The first is to divide the time into several snapshots [Du *et al.*, 2018]. For example, [Zhu *et al.*, 2016] proposes a matrix factorization based method for representing dynamic network snapshots. DANE [Li *et al.*, 2017] and DHPE [Zhu *et al.*, 2018] solve the dynamic network embedding problem with matrix perturbation theory. DynamicTriad [Zhou *et al.*, 2018] models the dynamics via triadic closure process and obtains node embeddings at each snapshot.

On the other hand, in order to consider the full dynamics over time, the second line of works try to consider the whole evolution process [Nguyen *et al.*, 2018]. For instance, HTNE [Zuo *et al.*, 2018] represents temporal networks with integrating the Hawkes process with consideration of neighborhood formation. Netwalk [Yu *et al.*, 2018] proposes a deep neural network and reservoir sampling-based network representation learning framework for real-time anomaly detection. M^2DNE [Lu *et al.*, 2019] captures both micro- and macro-dynamics of networks during evolution.

However, all the methods mentioned above either represent nodes on each snapshot or only consider limited dynamics and structures for temporal network embedding. None of them capture motif proximities in temporal network embedding from a mesoscopic view.

6 Conclusion

In this paper, we investigate the problem of representing nodes in temporal networks by incorporating the influence of meso-dynamics, in terms of the triad. In detail, we propose MTNE, a novel embedding model that leverages the Hawkes process to stimulate the triad formation process, and combines attention mechanism to distinguish the importance of different types of triads better. Experiments on different real-world networks demonstrate that MTNE achieves the best performance, compared with several state-of-the-art techniques in both static and dynamic tasks, including link prediction, node classification, and link recommendation. In the future, we will perform our model with other motifs, say four-node motif or above in real network applications.

Acknowledgments

The research was supported by National Natural Science Foundation of China (No. 61802140, U1936104, 61972442) and CCF-Tencent Open Fund.

References

[Bahdanau *et al.*, 2015] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. Neural machine translation by jointly learning to align and translate. In *Proceedings of ICLR '15*, 2015.

[Cao *et al.*, 2015] Shaosheng Cao, Wei Lu, and Qiongkai Xu. Grarep: Learning graph representations with global structural information. In *Proceedings of CIKM '15*, pages 891–900. ACM, 2015.

[Cui *et al.*, 2018] Peng Cui, Xiao Wang, Jian Pei, and Wenwu Zhu. A survey on network embedding. *IEEE TKDE*, 31(5):833–852, 2018.

[Du *et al.*, 2018] Lun Du, Yun Wang, Guojie Song, Zhicong Lu, and Junshan Wang. Dynamic network embedding: An extended approach for skip-gram based network embedding. In *Proceedings of IJCAI '18*, pages 2086–2092, 2018.

[Fournet and Barrat, 2014] Julie Fournet and Alain Barrat. Contact patterns among high school students. *PloS one*, 9(9), 2014.

[Grover and Leskovec, 2016] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *Proceedings of KDD '16*, pages 855–864. ACM, 2016.

[Hawkes, 1971] Alan G. Hawkes. Spectra of some self-exciting and mutually exciting point processes. *Biometrika*, 58(1):83–90, 1971.

[Hogg and Lerman, 2012] Tad Hogg and Kristina Lerman. Social dynamics of digg. *EPJ Data Science*, 1(1):5, 2012.

[Huang *et al.*, 2015] Hong Huang, Jie Tang, Lu Liu, JarDer Luo, and Xiaoming Fu. Triadic closure pattern analysis and prediction in social networks. *IEEE TKDE*, 27(12):3374–3389, 2015.

[Huang *et al.*, 2018] Hong Huang, Yuxiao Dong, Jie Tang, Hongxia Yang, Nitesh V. Chawla, and Xiaoming Fu. Will triadic closure strengthen ties in social networks? *TKDD*, 12(3):30, 2018.

[Ley, 2009] Michael Ley. Dblp – some lessons learned. *Proceedings of the VLDB Endowment*, 2(2):1493–1500, 2009.

[Li *et al.*, 2017] Jundong Li, Harsh Dani, Xia Hu, Jiliang Tang, Yi Chang, and Huan Liu. Attributed network embedding for learning in a dynamic environment. In *Proceedings of CIKM '17*, page 387–396, 2017.

[Lu *et al.*, 2019] Yuanfu Lu, Xiao Wang, Chuan Shi, Philip S Yu, and Yanfang Ye. Temporal network embedding with micro-and macro-dynamics. In *Proceedings of CIKM '19*, pages 469–478. ACM, 2019.

[Mikolov *et al.*, 2013] Tomas Mikolov, Ilya Sutskever, Kai Chen, Greg S. Corrado, and Jeff Dean. Distributed representations of words and phrases and their compositionality. In *Proceedings of NIPS '13*, pages 3111–3119, 2013.

[Nguyen *et al.*, 2018] Giang Hoang Nguyen, John Boaz Lee, Ryan A. Rossi, Nesreen K. Ahmed, Eunyee Koh, and Sungchul Kim. Continuous-time dynamic network embeddings. In *Proceedings of the Web Conference 2018*, pages 969–976, 2018.

[Perozzi *et al.*, 2014] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *Proceedings of KDD '14*, pages 701–710. ACM, 2014.

[Tang *et al.*, 2015] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *Proceedings of WWW '15*, pages 1067–1077, 2015.

[Wang *et al.*, 2017] Xiao Wang, Peng Cui, Jing Wang, Jian Pei, Wenwu Zhu, and Shiqiang Yang. Community preserving network embedding. In *Proceedings of AAAI '17*, page 203–209, 2017.

[Waters, 2015] Tony Waters. *Weber's rationalism and modern society: New translations on politics, bureaucracy, and social stratification*. Springer, 2015.

[Yu *et al.*, 2018] Wenchao Yu, Wei Cheng, Charu C. Aggarwal, Kai Zhang, Haifeng Chen, and Wei Wang. Netwalk: A flexible deep embedding approach for anomaly detection in dynamic networks. In *Proceedings of KDD '18*, page 2672–2681, 2018.

[Yu *et al.*, 2019] Yanlei Yu, Zhiwu Lu, Jiajun Liu, Guoping Zhao, Ji-Rong Wen, and Kai Zheng. Rum: Network representation learning using motifs. In *Proceedings of ICDE'19*, pages 1382–1393, April 2019.

[Zhang *et al.*, 2013] Jing Zhang, Biao Liu, Jie Tang, Ting Chen, and Juanzi Li. Social influence locality for modeling retweeting behaviors. In *Proceedings of IJCAI '13*, page 2761–2767, 2013.

[Zhou *et al.*, 2018] Lekui Zhou, Yang Yang, Xiang Ren, Fei Wu, and Yueting Zhuang. Dynamic network embedding by modeling triadic closure process. In *Proceedings of AAAI '18*, 2018.

[Zhu *et al.*, 2016] Linhong Zhu, Dong Guo, Junming Yin, Greg Ver Steeg, and Aram Galstyan. Scalable temporal latent space inference for link prediction in dynamic social networks. *IEEE TKDE*, 28(10):2765–2777, 2016.

[Zhu *et al.*, 2018] Dingyuan Zhu, Peng Cui, Ziwei Zhang, Jian Pei, and Wenwu Zhu. High-order proximity preserved embedding for dynamic networks. *TKDE*, 30(11):2134–2144, Nov 2018.

[Zuo *et al.*, 2018] Yuan Zuo, Guannan Liu, Hao Lin, Jia Guo, Xiaoqian Hu, and Junjie Wu. Embedding temporal network via neighborhood formation. In *Proceedings of KDD '18*, pages 2857–2866. ACM, 2018.