Learning Dynamic Graph Representations

Rakshit S. Trivedi
College of Computing
Georgia Institute of Technology
rstrivedi@gatech.edu

Mehrdad Farajtabar
DeepMind
farajtabar@google.com

Prasenjeet Biswal
College of Computing
Georgia Institute of Technology
bprasenjeet1108@gmail.com

Hongyuan Zha
College of Computing
Georgia Institute of Technology
zha@cc.gatech.edu

Abstract

We address two fundamental questions that arise in learning over dynamic graphs: (i) How to elegantly model dynamical processes over graphs? (ii) How to leverage such a model to effectively encode evolving graph information into low-dimensional representations? We present *DyRep* - a novel modeling framework for dynamic graphs that posits representation learning as a *latent mediation process* bridging two observed processes – dynamic *of* the network (topological evolution) and dynamic *on* the network (activities of the nodes). To this end, we propose an inductive framework comprising of two-time scale deep temporal point process model parameterized by a temporal-attentive representation network and trained end-to-end using an efficient unsupervised procedure. We demonstrate that DyRep significantly outperforms state-of-art baselines for dynamic link prediction and event time prediction and provide qualitative analysis of our framework.

1 Introduction

Representation learning over graph structured data has ubiquitous applicability in variety of domains such as social networks, bioinformatics, natural language processing, and relational knowledge bases. Learning node representations to effectively encode high-dimensional and non-Euclidean graph information is a challenging problem but recent advances in deep learning has helped important progress towards addressing it [1, 2, 3, 4, 5, 6, 7], with majority of the approaches focusing on advancing the state-of-art in static graph setting. However, several domains (e.g. social network communications, financial transaction graphs or longitudinal citation data) now present highly dynamic data that exhibit complex temporal properties in addition to earlier cited challenges. These recent developments have created a conspicuous need for principled approaches to advance graph embedding techniques for dynamic graphs [8]. We focus on two pertinent questions fundamental to representation learning over dynamic graphs: (i) *What can serve as an elegant model for dynamic processes over graphs?* (ii) *How can one leverage such a model to learn dynamic node representations that are effectively able to capture evolving graph information over time?*

As noted in [9], an important requirement to effectively learn over such dynamical systems is the ability to express the dynamical processes at different scales. We propose that any dynamic graph must be minimally expressed as a result of two fundamental processes evolving at different time scales: **Association Process** (dynamics *of* the network), that brings change in the graph structure and leads to long lasting information exchange between nodes; and **Communication Process** (dynamics *on* the network), that relates to node's self evolution and activities between (not necessarily connected)

Workshop on Modeling and Decision-Making in the Spatiotemporal Domain, 32nd Conference on Neural Information Processing Systems (NeurIPS 2018), Montréal, Canada.

nodes which leads to temporary information flow between them [10, 11]. We, then, posit our goal of learning node representations as modeling a *latent mediation process* that bridges the above two observed processes such that learned embeddings drive the complex temporal dynamics of both processes and these processes subsequently lead to the nonlinear evolution of node representations. Further, the information propagated across the graph is governed by the temporal dynamics of communication and association histories of nodes with its neighborhood (Figure 3 Appendix **A**).

2 Preliminaries

Dynamic Graph Setting — Let $\mathcal{G}_t = (\mathcal{V}_t, \mathcal{E}_t)$ denote graph $\mathcal{G}$ at time t, where $\mathcal{V}_t$ is the set of nodes and $\mathcal{E}_t$ is the set of edges in $\mathcal{G}_t$. An event at time t is represented as $e = (u, v, t, k)$, where u, v are the two nodes involved in an event. $k \in \{0, 1\}$ and we use $k = 0$ to signify events from the topological evolution process (association) and $k = 1$ to signify events from node interaction process (communication). We then represent complete set of P observed events ordered by time in window $[0, T]$ as $\mathcal{O} = \{(u, v, t, k)_p\}_{p=1}^{P}$. Here, $t_p \in \mathbb{R}^+$, $0 \leq t_p \leq T$. Let $\mathbf{z}^v(t) \in \mathbb{R}^d$ represent d-dimensional representation of node v at time t. We use $\mathbf{z}^v(\bar{t})$ for most recently updated embedding of node v just before t.

3 Proposed Method

The rationale behind our framework is that the observed set of events are the realizations of the nonlinear dynamic processes governing the changes in topological structure of graph and interactions between the nodes in the graph. Now, when an event is observed between two nodes, information flows from the neighborhood of one node to the other and affects the representations of the nodes accordingly. While a communication event (interaction) only propagates local information across two nodes, an association event changes the topology and thereby has more global effect. The goal is to learn node representations that encode information evolving due to such local and global effects.

Modeling Two-Time Scale Observed Graph Dynamics. Given an observed event $p = (u, v, t, k)$, we define a continuous-time deep model of temporal point process using the intensity function $\lambda_k^{u,v}(t)$ that models the occurrence of event p between nodes u and v at time t: $\lambda_k^{u,v}(t) = f_k(g_k(\bar{t}))$. The inner function $g_k(\bar{t})$ computes the compatibility of the most recently updated representations of two nodes, $\mathbf{z}^u(\bar{t})$ and $\mathbf{z}^v(\bar{t})$ as follows: $g_k(\bar{t}) = \boldsymbol{\omega}_k^T \cdot [\mathbf{z}^u(\bar{t}); \mathbf{z}^v(\bar{t})]$. Here, [;] signifies concatenation and $\boldsymbol{\omega}_k \in \mathbb{R}^{2d}$ serves as the model parameter that learns time-scale specific compatibility. The choice of outer function f_k needs to: 1.) keep intensity positive. 2.) account for difference in time-scale of dynamics corresponding to communication and association processes. To this end, we use a modified version of softplus function parameterized by a dynamics parameter ψ_k to capture *timescale* dependence: $f_k(x) = \psi_k \log(1 + \exp(x/\psi_k))$ where, ψ_k is scalar time-scale parameter learned as part of training. ψ_k corresponds to the rate of events arising from a corresponding process.

Learning latent Mediation Process via Temporally Attentive Representation Network. After an event has occurred, the representation of both the participating nodes need to be updated to capture the effect of the observed event based on the principles of: *Self-Propagation.*, *Exogenous Drive.* and *Localized Embedding Propagation.* Two nodes involved in an event form a temporary (communication) or a permanent (association) pathway for the information to propagate from the neighborhood of one node to the other node. For any event at time t, we update the embeddings for both nodes involved in the event using a recurrent architecture. Specifically, for p-th event of node v, we evolve $\mathbf{z}^v$ as:

$$\mathbf{z}^v(t_p) = \sigma(\underbrace{\mathbf{W}^{struct}\mathbf{h}^u_{struct}(\bar{t_p})}_{\text{Localized Embedding Propagation}} + \underbrace{\mathbf{W}^{rec}\mathbf{z}^v(\bar{t_p^v})}_{\text{Self-Propagation}} + \underbrace{\mathbf{W}^t(t_p - \bar{t_p^v})}_{\text{Exogenous Drive}}), \tag{1}$$

where, $\mathbf{h}^u_{struct} \in \mathbb{R}^d$ is the output representation vectors obtained from aggregator function on node u's neighborhood and $\mathbf{z}^v(\bar{t_p^v})$ is the recurrent state. t_p is time point of current event, $\bar{t_p}$ signifies the timepoint just before current event and $\bar{t_p^v}$ represent time point of previous event for node v. $\mathbf{W}^{struct}, \mathbf{W}^{rec} \in \mathbb{R}^{d \times d}$ and $\mathbf{W}^t \in \mathbb{R}^d$ are model parameters that govern the aggregate effect of all the three processes respectively. The above formulation is flexible in supporting various features. For example, in case of dynamic heterogenous graphs, Eq. 1 can be extended to include edge type with corresponding parameters. Similarly, one can support high-dimensional attributes (including spatial) by adding them to Eq. 1 as feature inputs to the model.

Temporally Attentive Aggregation. We propose a novel *Temporal Attention Mechanism* to compute $\mathbf{h_{struct}}$ by attending to the neighbors based on node's communication and association history.

Let $\mathbf{A}(t) \in \mathbb{R}^{n \times n}$ be the adjacency matrix for graph $\mathcal{G}_t$ at time t. Let $\mathcal{S}(t) \in \mathbb{R}^{n \times n}$ be a stochastic matrix capturing the strength between pair of vertices at time t. One can consider $\mathcal{S}$ as a selection matrix that induces a ***natural selection*** process for a node – it would tend to communicate more with other nodes that it wants to associate with or has recently associated with. And it would want to attend less to non-interesting nodes. Formally, we perform *localized attention* for a given node u and compute the coefficients pertaining to the 1-hop neighbors i of node u as: $q_{ui}(t) = \frac{\exp(\mathcal{S}_{ui}(\bar{t}))}{\sum_{i' \in \mathcal{N}_u(t)} \exp(\mathcal{S}_{ui'}(\bar{t}))}$, where q_{ui} signifies the attention weight for the neighbor i at time t and hence it is a temporally evolving quantity. These attention coefficients are used to compute the aggregate information $\mathbf{h}^u_{struct}(\bar{t})$ by employing an attended aggregation mechanism across neighbors as follows: $\mathbf{h}^u_{struct}(\bar{t}) = \max\left(\left\{\sigma\left(q_{ui}(t) \cdot \mathbf{h}^i(\bar{t})\right), \forall i \in \mathcal{N}_u(\bar{t})\right\}\right)$, where, $\mathbf{h}^i(\bar{t}) = \mathbf{W}^h \mathbf{z}^i(\bar{t}) + \mathbf{b}^h$ and $\mathbf{W}^h \in \mathbb{R}^{d \times d}$ and $\mathbf{b}^h \in \mathbb{R}^d$ are parameters governing the information propagated by each neighbor of u. $\mathbf{z}^i(\bar{t}) \in \mathbb{R}^d$ is the most recent embedding for node i.

Spatial Properties. Our work considers spatial properties implicitly. Concretely, the distance between the nodes of graph in embedded space effects the pattern of connections and communications between the nodes. Simultaneously, the dynamics "of" and "on" the network affect the node distances in embedded space. If one wants to introduce explicit spatial features (e.g. location in transportation or mobile communication network), our model will support it by adding such features as part of Eq. 1. Another approach could be to combine the temporal attentive operator with a local spatial operator (e.g. geometric deep learning) to build a sophisticated spatiotemporal operator.

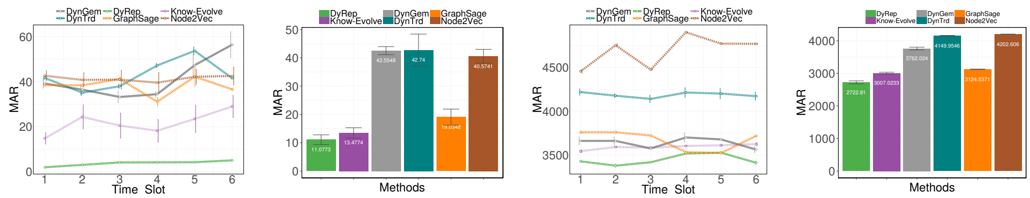
Efficient Learning Procedure. The complete parameter space for the current model is $\mathbf{\Omega} = \{\mathbf{W}^{struct}, \mathbf{W}^{rec}, \mathbf{W}^t, \mathbf{W}^h, \mathbf{b}^h, \{\boldsymbol{\omega}_k\}_{k=0,1}, \{\psi_k\}_{k=0,1}\}$. For a set $\mathcal{O}$ of P observed events, we learn these parameters by minimizing the negative log likelihood: $\mathcal{L} = -\sum_{p=1}^{P} \log\left(\lambda_p(t)\right) + \int_0^T \Lambda(\tau) d\tau$, where $\lambda_p(t) = \lambda_{k_p}^{u_p, v_p}(t)$ represent the intensity of event at time t and $\Lambda(\tau) = \sum_{u=1}^{n} \sum_{v=1}^{n} \sum_{k \in \{0,1\}} \lambda_k^{u,v}(\tau)$ represent total survival probability for events that do not happen. While it is intractable (will require $\mathcal{O}(n^2 k)$ time) to compute the integral in log-likelihood equation for all possible non-events in a stochastic setting, we locally optimize $\mathcal{L}$ using mini-batch stochastic gradient descent where we estimate the integral using novel sampling technique.

4 Experiments

We evaluate DyRep and baselines on two real world datasets: **Social Evolution Dataset** released by MIT Human Dynamics Lab — #nodes: 100, #Initial Associations: 407, #Final Associations: 809, #Communications: 2020554 and Clustering Coefficient: 0.548. **Github Dataset** available at Github Archive — #nodes: 12328, #Initial Associations: 70640, #Final Associations: 166565, #Communications: 604954 and Clustering Coefficient: 0.087. We study the effectiveness of DyRep by evaluating our model on tasks of: (i) Dynamic link prediction – for a given test record (u, v, t, k), replace v with other entities in the graph and compute the conditional density $f_k^{u,v}(t) = \lambda_k^{u,v}(t) \cdot \exp\left(\int_{\bar{t}}^{t} \lambda(s) ds\right)$, where $\bar{t}$ is the time of the most recent event on either dimension u or v. We then rank all the entities in descending order of the density and report Mean Average Rank and HITS(@10) metric for dynamic link prediction. (ii) Event Time Prediction – Given a pair of nodes (u, v) and event type k at time t, we use the above density formulation to compute conditional density at time t. The next time point $\hat{t}$ for the event can then be computed as: $\hat{t} = \int_t^\infty t f_k^{u,v}(t) dt$. For a given test record, we predict the next time an event may occur and report MAE against the ground truth. For Dynamic Link Prediction task, we compare the performance of our model against Know-Evolve [12], DynGem [13], DynTrd [14], GraphSage [15] and Node2Vec [2]. For Event Time Prediction, we compare our model against Know-Evolve which has the ability to predict time and Multi-dimensional Hawkes Process (MHP) [16] model where all events in graph are considered as dyadic. During evaluation, we divide our test sets into $n(=6)$ slots based on time and report the performance for each time slot, thus providing comprehensive temporal evaluation of different methods.

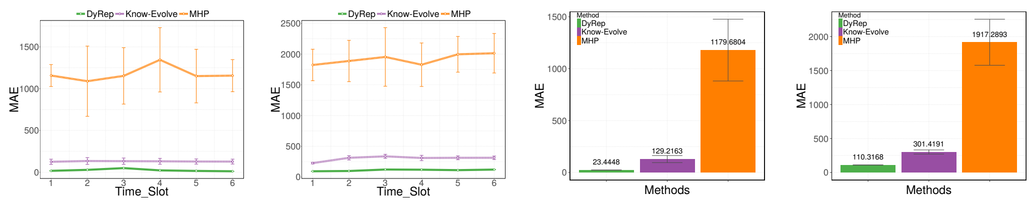
4.1 Experimental Results

Communication Event Prediction Performance (Figure 1 **(a-b)**). For Social Evolution dataset, our method significantly and consistently outperforms all the baselines on both metrics. While the



(a) Social (Communication) (b) Social (Association) (c) Github (Communication) (c) Github (Association)

Figure 1: Dynamic Link Prediction Performance for **(a-b)** Social Evolution Dataset **(c-d)** Github Dataset. We report HITS@10 results and zoomed versions in **Appendix D**. Best viewed in pdf.



(a) Social Evolution (b) Github (c) Social Evolution (d) Github

Figure 2: Time Prediction Performance (unit is hrs). Figure best viewed in pdf or colored print.

performance of our method drops a little over time, it is expected due to the temporal recency affect on node's evolution. For Github dataset, we demonstrate comparable performance with both Know-Evolve and GraphSage on Rank metric. It is notable that that overall performance for all methods on rank metric is low. Github dataset is very sparse with very low clustering coefficient and challenging dataset to learn. It is expected that for a large number of nodes with no communication history, most of the methods will show comparable performance but our method outperforms all others when there is some history available. This is demonstrated by our significantly better performance for HITS@10 metric where we are able to do highly accurate prediction for nodes where we learn better history.

Association Event Prediction Performance. Association events are not available for all time slots so Figure 1 **(c-d)** report the aggregate number for this task. For both the datasets, our model significantly outperforms the baselines for this task. Specifically, our model's strong performance on HITS@10 metric across both datasets demonstrates its robustness in accurate learning from various properties of data. On Social evolution dataset, the number of association events are very small (only 485) and hence our strong performance shows that the model is able to capture the influence of communication events on the association events through the learned representations (mediation). On the Github dataset, the network grows through new nodes and our model's strong performance across both metric demonstrates its inductive ability to generalize across new nodes across time.

Time Prediction Performance. Figure 2 demonstrates consistently better performance than state-of-art baseline for event time prediction on both datasets. While Know-Evolve models both processes as two different relations between entities, it does not explicitly capture the variance in the time scales of two processes. Further, Know-Evolve does not consider influence of neighborhood which may lead to capturing weaker temporal-structural dynamics across the graph. MHP uses specific parametric intensity function which fails to account for intricate dependencies across graph. We refer the readers to **Appendix B** for sample qualitative analysis of our framework.

5 Conclusion

We introduced a novel modeling framework for dynamic graphs that effectively and efficiently learns node representations by posing representation learning as latent mediation process bridging dynamic processes of topological evolution and node interactions. We proposed a deep temporal point process model parameterized by temporally attentive representation network that models these complex and nonlinearly evolving dynamic processes and learns to encode structural-temporal information over graph into low dimensional representations. Our superior evaluation performance demonstrates the effectiveness of our approach compared to state-of-arts. Our framework can support wide range of dynamic graph characteristics which can potentially have many exciting adaptations. As a part of our framework, we also propose a novel temporal point process based attention mechanism that can attend over neighborhood based on the history of communications and associations in the graph.

References

[1] Shaosheng Cao, Wei Lu, and Qiongkai Xu. Grarep: Learning graph representations with global structural information. In *CIKM*, 2015.

[2] Aditya Grover and Jure Leskovec. node2vec: Scalable feature learning for networks. In *KDD*, 2016.

[3] Bryan Perozzi, Rami Al-Rfou, and Steven Skiena. Deepwalk: Online learning of social representations. In *KDD*, 2014.

[4] Jian Tang, Meng Qu, Mingzhe Wang, Ming Zhang, Jun Yan, and Qiaozhu Mei. Line: Large-scale information network embedding. In *WWW*, 2015.

[5] Daixin Wang, Peng Cui, and Wenwu Zhu. Structural deep network embedding. In *KDD*, 2016.

[6] Xiao Wang, Peng Cui, Jing Wang, Jian Pei, Wenwu Zhu, and Shiquiang Yang. Community preserving network embedding. In *AAAI*, 2017.

[7] Linchuan Xu, Xiaokai Wei, Jiannong Cao, and Philip Y. Yu. Embedding identity and interest for social networks. In *WWW*, 2017.

[8] William L. Hamilton, Rex Ying, and Jure Leskovec. Representation learning on graphs: Methods and applications. *arXiv:1709.05584*, 2017.

[9] Bernard Chazelle. Natural algorithms and influence systems. *Communications of the ACM*, 2012.

[10] Damien Farine. The dynamics of transmission and the dynamics of networks. *Journal of Animal Ecology*, 86(3):415–418, 2017.

[11] Oriol Artime, Jose J. Ramasco, and Maxi San Miguel. Dynamics on networks: competition of temporal and topological correlations. *arXiv:1604.04155*, 2017.

[12] Rakshit Trivedi, Hanjun Dai, Yichen Wang, and Le Song. Know-evolve: Deep temporal reasoning for dynamic knowledge graphs. In *ICML*, 2017.

[13] Palash Goyal, Nitin Kamra, Xinran He, and Yan Liu. Dyngem: Deep embedding method for dynamic graphs. *IJCAI International Workshop on Representation Learning for Graphs*, 2017.

[14] Lekui Zhou, Yang Yang, Xiang Ren, Fei Wu, and Yueting Zhuang. Dynamic network embedding by modeling triadic closure process. In *AAAI*, 2018.

[15] William L Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *NIPS*, 2017.

[16] Nan Du, Yichen Wang, Niao He, and Le Song. Time sensitive recommendation from recurrent user activities. In *NIPS*, 2015.

[17] Giang Hoang Ngyuyen, John Boaz Lee, Ryan A. Rossi, Nesreen K. Ahmed, Eunyee Koh, and Sungchul Kim. Continuous-time dynamic network embeddings. In *WWW*, 2018.

[18] Wenchao Yu, Wei Cheng, Charu C. Aggarwal, Kai Zhang, Haifeng Chen, and Wei Wang. Netwalk: A flexible deep embedding approach for anamoly detection in dynamic networks. In *KDD*, 2018.

[19] Linhong Zhu, dong Guo, Junming Yin, Greg Ver Steeg, and Aram Galstyan. Scalable temporal latent space inference for link prediction in dynamic social networks. *TKDE*, 2016.

[20] Mehrdad Farajtabar, Yichen Wang, Manuel Gomez-Rodriguez, Shuang Li, Hongyuan Zha, and Le Song. Coevolve: A joint point process model for information diffusion and network co-evolution. In *NIPS*, 2015.

[21] Franco Scarselli, Marco Gori, Ah Chung Tsoi, Markus Hagenbuchner, and Gabriele Monfardini. The graph neural network model. *Neural Networks, IEEE Transactions on*, 20(1):61–80, 2009.

[22] Yujia Li, Daniel Tarlow, Marc Brockschmidt, and Richard Zemel. Gated graph sequence neural networks. In *ICLR*, 2016.

[23] Hanjun Dai, Bo Dai, and Le Song. Discriminative embeddings of latent variable models for structured data. In *ICML*, 2016.

[24] Thomas N Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *ICLR*, 2017.

[25] Thomas N Kipf and Max Welling. Variational graph auto-encoders. *arXiv:1611.07308*, 2016.

[26] Joan Bruna, Wojciech Zaremba, Arthur Szlam, and Yann LeCun. Spectral networks and locally connected networks on graphs. In *ICLR*, 2014.

[27] Youngjoo Seo, Michael Defferrard, Pierre Vandergheynst, and Xavier Bresson. Structured sequence modeling with graph convolutional recurrent networks. *arXiv:1612.07659*, 2016.

[28] Bomin Kim, Kevin Lee, Lingzhou Xue, and Xiaoyue Niu. A review of dynamic network models with latent variables. *arXiv:1711.10421*, 2017.

[29] Carl Yang, Mengxiong Liu, Zongyi Wang, Liuyan Liu, and Jiawei Han. Graph clustering with dynamic embedding. *arXiv:1712.08249*, 2017.

[30] Purnamrita Sarkar, Sajid Siddiqi, and Geoffrey Gordon. A latent space approach to dynamic embedding of co-occurence data. In *AISTATS*, 2007.

[31] DJ Daley and D Vere-Jones. *An Introduction to the Theory of Point Processes: Volume I: Elementary Theory and Methods*. 2007.

[32] Mehrdad Farajtabar, Nan Du, Manuel Gomez Rodriguez, Isabel Valera, Hongyuan Zha, and Le Song. Shaping social activity by incentivizing users. In *NIPS*, 2014.

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

[34] Yichen Wang, Bo Xie, Nan Du, and Le Song. Isotonic hawkes processes. In *ICML*, 2016.

[35] Behzad Tabibian, Isabel Valera, Mehrdad Farajtabar, Le Song, Bernhard Schölkopf, and Manuel Gomez-Rodriguez. Distilling information reliability and source trustworthiness from digital traces. In *WWW*, 2017.

[36] V. Isham and M. Westcott. A self-correcting point process. *Advances in Applied Probability*, 37:629–646, 1979.

[37] Nan Du, Hanjun Dai, Rakshit Trivedi, Utkarsh Upadhyay, Manuel Gomez-Rodriguez, and Le Song. Recurrent marked temporal point processes: Embedding event history to vector. In *KDD*, 2016.

[38] Hongyuan Mei and Jason M Eisner. The neural hawkes process: A neurally self-modulating multivariate point process. In *NIPS*, 2017.

[39] Shuai Xiao, Mehrdad Farajtabar, Xiaojing Ye, Junchi Yan, Le Song, and Hongyuan Zha. Wasserstein learning of deep generative point process models. In *NIPS*, 2017.

Appendix

A Overview of DyRep Framework

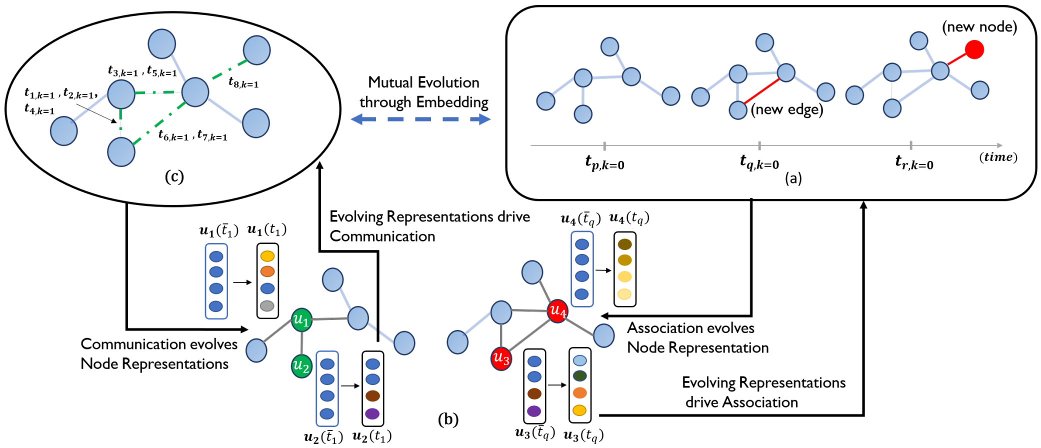
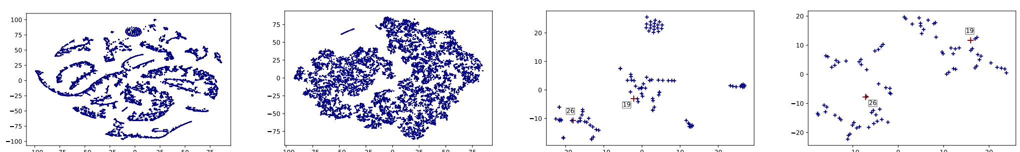


Figure 3: Evolution Through Mediation. **(a)** Association events (k=0) where the node or edge grows. **(c)** Communication Events (k=1) where nodes interact with each other. For both these processes, $t_{p,k=0} < (t_1, t_2, t_3, t_4, t_5)_{k=1} < t_{q,k=0} < (t_6, t_7)_{k=1} < t_{r,k=0}$. **(b)** Evolving Representations.

B Sample Qualitative Performance



(a) DyRep Embeddings (b) GraphSage Embeddings (c) DyRep Embeddings (d) GraphSage Embeddings

Figure 4: tSNE for learned embeddings after training. Figure best viewed in color.

Qualitative Performance. Figure 4 **(a-b)** shows the tSNE embeddings learned by Dyrep (left) and GraphSage (right) respectively. The visualization demonstrates that DyRep embeddings have more discriminative power as it can effectively capture the distinctive and evolving structural features over time as aligned with empirical evidence. Figure 4 **(c-d)** shows use case of two associated nodes (19 and 26) that has less communication at the two time points for above two methods. DyRep keeps the embeddings nearby although not in same cluster (cos. dist. - 0.649) which demonstrates its ability to learn the association and less communication dynamics between two nodes. For GraphSage the embeddings are on opposite ends of cluster with (cos. dist. - 1.964).

C Contributions and Related Work

Recent availability of dynamic graphs have created a conspicuous need for principled approaches to advance graph embedding techniques for dynamic graphs [8]. We focus on two pertinent questions fundamental to representation learning over dynamic graphs:

(i) *What can serve as an elegant model for dynamic processes over graphs?* — A key modeling choice in existing representation learning techniques for dynamic graphs [13, 14, 12, 17, 18] assume that graph dynamics evolve as a single time scale process. In contrast to these approaches, we observe that most real-world graphs exhibit at least two distinct dynamic processes that evolve at different time scales — *Topological Evolution*: where the number of nodes and edges are expected to grow (or shrink) over time leading to structural changes in the graph; and *Node Interactions*: which relates to activities between nodes that may or may not be structurally connected. Modeling interleaved dependencies between these non-linearly evolving dynamic processes is a crucial next step for advancing the formal models of dynamic graphs.

(ii) *How can one leverage such a model to learn dynamic node representations that are effectively able to capture evolving graph information over time?* — Existing techniques in this direction can be divided into two approaches: a.) Discrete-Time Approach, where the evolution of a dynamic graph is observed as collection of static graph snapshots over time [19, 13, 14]. These approaches tend to preserve (encode) very limited structural information and capture temporal information at a very coarse level which leads to loss of information between snapshots and lack of ability to capture fine-grained temporal dynamics. Another challenge in such approaches is the selection of appropriate aggregation granularity which is often misspecified. b.) Continuous-Time Approach, where evolution is modeled at finer time granularity in order to address the above challenges. While existing approaches have demonstrated to be very effective in specific settings, they either model simple structural and complex temporal properties in a decoupled fashion [12] or use simple temporal models (exponential family in [17]). But several domains exhibit highly nonlinear evolution of structural properties coupled with complex temporal dynamics and it remains an open problem to effectively model and learn informative representations capturing various dynamical properties of such complex systems.

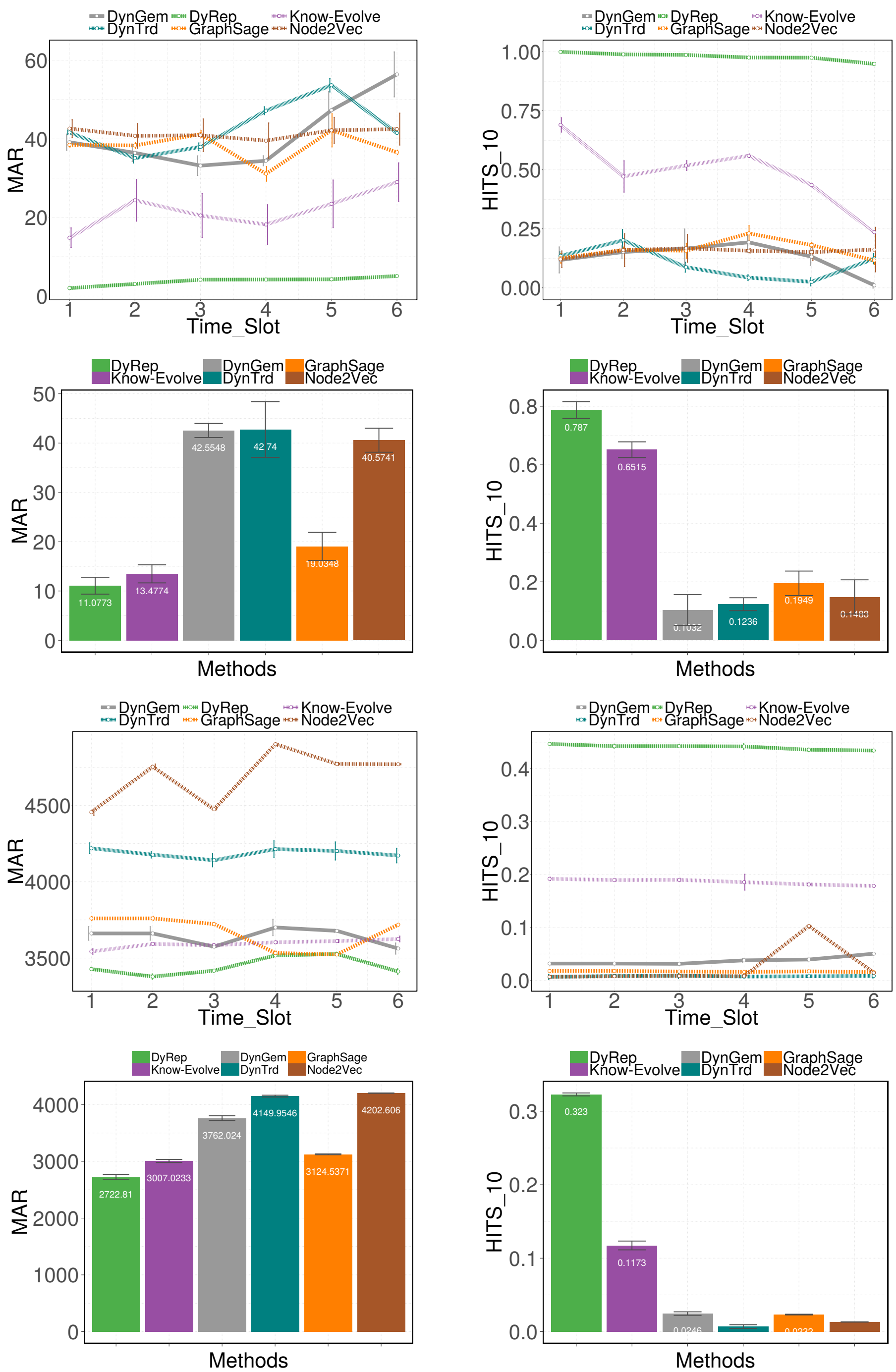
Contributions — First, our work expresses dynamic graphs at multiple scales as follows: a.) Dynamic "of" the Network: This corresponds to the topological changes in network – insertion or deletion of nodes and edges b.) Dynamic "on" the Network: This corresponds to various activities in the network – self evolution of node's interests/features, change in node's features due to exogenous drive (activities external to net-work), information propagation within network and within-network interactions between nodes which may or may not have direct edge between them. This dichotomy of dynamic network processes is well-known and has been subject of several studies [9, 10, 11, 20] in segregated manner. But none of the existing machine learning approaches has jointly modeled them for representation learning over dynamic graphs (our key objective) to the best of our knowledge.

Next, we propose a novel representation learning framework for dynamic graphs, to model interleaved evolution of two observed processes through latent mediation process expressed above and effectively learn richer node representations over time. Our framework ingests dynamic graph information in the form of association and communication events over time and updates the node representations as they appear in these events. We build a two-time scale deep temporal point process approach to capture the continuous-time fine-grained temporal dynamics of the two observed processes. We further parameterize the conditional intensity function of the temporal point process with a deep inductive representation network that learns functions to compute node representations. Finally, we couple the structural and temporal components of our framework by designing a novel *Temporal Attention Mechanism*, which induces temporal attentiveness over neighborhood nodes using the learned conditional intensity function. This allows to capture highly interleaved and nonlinear dynamics governing node representations over time. We design an efficient unsupervised training procedure for end-to-end training of our framework.

Related Work — Representation Learning approaches for static graphs either perform node embedding [1, 2, 3, 4, 5, 6, 7] or sub-graph embedding [21, 22, 23] which can also utilize convolutional neural networks [24, 25, 26, 15]. Dynamic network embedding is pursued through various techniques such as matrix factorization [19], structural properties [14], CNN-based approaches [27], deep recurrent models [12], and random walks [17]. Literature on temporal modeling of dynamic networks [28], that focus on link prediction tasks is orthogonal to our work as they do not focus on representation learning. Authors in [29, 30] proposed models of learning dynamic embeddings but none of them consider time at finer level and do not capture both topological evolution and interactions simultaneously. Temporal Point Processes [31] have gained lot of attention recently for modeling dynamic systems [32, 33, 34, 35, 36]. In parallel, research on deep point process models include parametric approaches to learn intensity [37, 38] using RNN and GAN based approaches to learn intensity functions [39].

D More Experimental Results

Figure 5 provides HITS@10 results in addition to the MAR results reported for Link Prediction in Section 5 (Experiments) of the main paper.



(a) MAR (b) HITS@10

Figure 5: Dynamic Link Prediction Performance: **Top 2 rows** show performance for Social Evolution Dataset. **Bottom 2 rows** show performance for Github Dataset. 1st and 3rd row show performance for Communication Events while 2nd and 4th row show performance for Association Events.