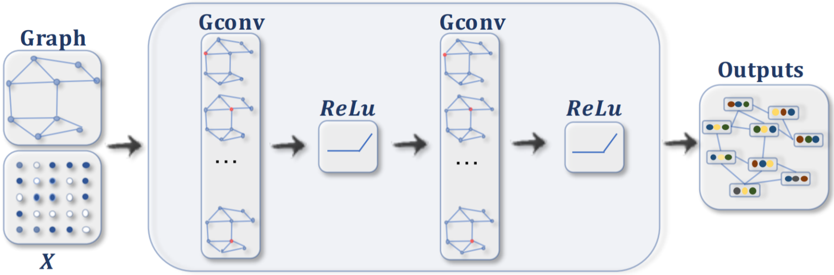
Inductive Representation Learning on Temporal Graphs

Медведев Алексей Владимирович

МГУ имени М. В. Ломоносова, факультет ВМК, кафедра ММП

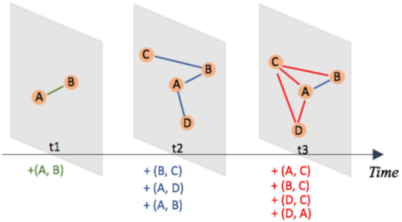
GNNs



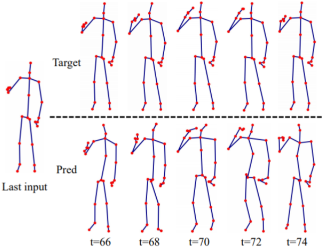
Architecture

- **Spectral**: based on graph Laplacian. Solves only **transductive** tasks.
- **Spatial**: based on neighborhood aggregation via spatial operator. Solves both **inductive** and **transductive tasks**.

Task types

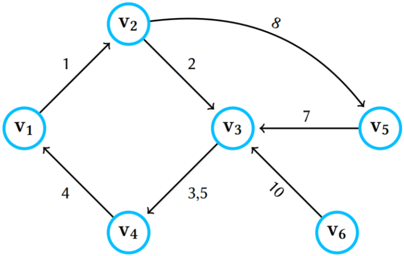


Inductive



Transductive

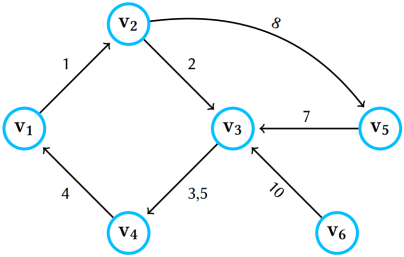
Continuous-Time Dynamic Network Embeddings (Nguyen et al., 2018)



Continuous-Time Dynamic Network

$G = (V, E_T, \mathcal{T})$, V is a set of vertices, and $E_T \subseteq V \times V \times \mathbb{R}^+$ is the set of temporal edges, and $T : E \to \mathbb{R}^+$ is a function that maps each edge to a corresponding timestamp.

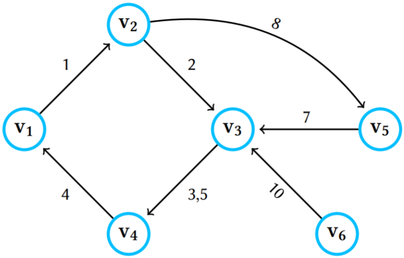
Continuous-Time Dynamic Network Embeddings



Temporal Walk

$\langle v_1, v_2, \ldots, v_k \rangle$ such that $\langle v_i, v_{i+1} \rangle \in E_T$, $1 \leq i < k$, and $\mathcal{T}(v_i, v_{i+1}) \leq \mathcal{T}(v_{i+1}, v_{i+2})$, $1 \leq i < (k-1)$.

Continuous-Time Dynamic Network Embeddings



Temporal Neighborhood

$$\Gamma_t(v) = \{(w, t') \mid e = (v, w, t') \in E_T \wedge \mathcal{T}(e) > t\}$$

Goal

Given $G = (V, E_T, \mathcal{T})$ goal is to learn $f : V \to \mathbb{R}^D$, that maps nodes to representations suitable for a down-stream machine learning task such as temporal link prediction.

Continuous-Time Dynamic Network Embeddings

Initial Temporal Edge Selection

Unbiased:

$$\Pr(e) = 1/|E_T|$$

Exponential:

$$\Pr(e) = \frac{\exp\left[\mathcal{T}(e) - t_{\min}\right]}{\sum\limits_{e' \in E_T} \exp\left[\mathcal{T}(e') - t_{\min}\right]}$$

Linear:

$$\Pr(e) = \frac{\text{rank-asc}(e)}{\sum\limits_{e' \in E_T} \text{rank-asc}(e')}$$

Continuous-Time Dynamic Network Embeddings

Temporal Random Walk

Unbiased:

$$\Pr(w) = 1/|\Gamma_t(v)|$$

Exponential:

$$\Pr(w) = \frac{\exp\left[\tau(w) - \tau(v)\right]}{\sum_{w' \in \Gamma_t(v)} \exp\left[\tau(w') - \tau(v)\right]}$$

Linear:

$$\Pr(w) = \frac{\text{rank-desc}(w)}{\sum_{w' \in \Gamma_t(v)} \text{rank-desc}(w')}$$

Continuous-Time Dynamic Network Embeddings

Optimization problem

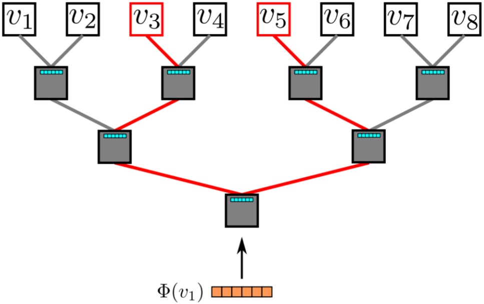
$$\max_f \log \Pr\left(W_T = \{v_{i-w}, \ldots, v_{i+w}\} \setminus v_i \mid f(v_i)\right),$$

$$\mathcal{T}(v_{i-\omega}, v_{i-\omega+1}) < \cdots < \mathcal{T}(v_{i+\omega-1}, v_{i+\omega})$$

$$\Pr\left(W_T \mid f(v_i)\right) = \prod_{v_{i+k} \in W_T} \Pr\left(v_{i+k} \mid f(v_i)\right) =$$

$$\Pr\left(n \mid f(u)\right) = \frac{\exp(f(n)f(u))}{\sum_{v \in V} \exp(f(v)f(u))},\ O(V)$$

Continuous-Time Dynamic Network Embeddings



Hierarchical Softmax

$$\Pr(n \mid f(u)) = \prod_{l=1}^{\log|V|} \Pr(b_l \mid f(u)), \; O(\log|V|)$$

Continuous-Time Dynamic Network Embeddings

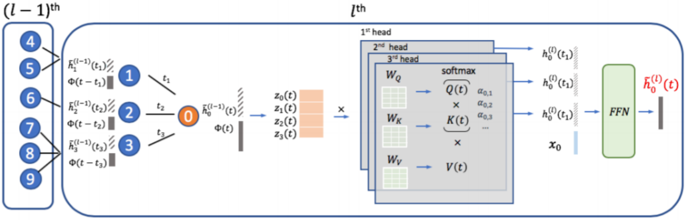
Algorithm 1 Continuous-Time Dynamic Network Embeddings

Input:

a (un)weighted and (un)directed dynamic network $G = (V, E_T, \mathcal{T})$,
temporal context window count β, context window size ω,
embedding dimensions D,

Set maximum walk length $L = 80$
Initialize set of *temporal walks* $\mathcal{S}_T$ to $\emptyset$
Initialize number of context windows $C = 0$
Precompute sampling distribution $\mathbb{F}_s$ using G
    $\mathbb{F}_s \in \{\text{Uniform, Exponential, Linear}\}$
$G' = (V, E_T, \mathcal{T}, \mathbb{F}_s)$
while $\beta - C > 0$ **do**
    Sample an edge $e_* = (v, u)$ via distribution $\mathbb{F}_s$
    $t = \mathcal{T}(e_*)$
    $S_t = \textsc{TemporalWalk}(G', e_* = (v, u), t, L, \omega + \beta - C - 1)$
    if $|S_t| > \omega$ **then**
        Add the *temporal walk* S_t to $\mathcal{S}_T$
        $C = C + (|S_t| - \omega + 1)$
end while
$\mathbf{Z} = \textsc{StochasticGradientDescent}(\omega, D, \mathcal{S}_T)$
return the *dynamic* node embedding matrix $\mathbf{Z}$

Inductive Representation Learning on Temporal Graphs (Xu et al., 2020)



Motivation

Nguyen et al. (2018) approach only generates embeddings for the **final** state of temporal graph and **transductive**.

Inductive Representation Learning on Temporal Graphs

Self Attention

$$Z_e = [z_{e_1} + p_1, \ldots, z_{e_l} + p_l]^T \in \mathbb{R}^{l \times d},$$

$$Z_e = [z_{e_1} || p_1, \ldots, z_{e_l} || p_l]^T \in \mathbb{R}^{l \times d + d_l}$$

Where z_{e_i} – input embeddings and p_i – positional embeddings.

$$\mathrm{Attn}(Q, K, V) = \mathrm{softmax}\left(\frac{QK^T}{\sqrt{d}} V\right)$$

$$Q = Z_e W_Q,\ K = Z_e W_k, V = Z_e W_V$$

Kernel Trick

$$F : T \to R^{d_T},\ K(t_1, t_2) = \langle F(t_1), F(t_2) \rangle = \psi(t1 - t2)$$

Bochner's Theorem

A continuous, translation-invariant kernel $K(t_1, t_2) = \psi(t1 - t2)$ is positive definite if and only if there exists a non-negative measure on $\mathbb{R}$ such that ψ is the Fourier transform of the measure.

$$\psi(t1 - t2) = \int_{\mathbb{R}} \exp\left(iw(t_1 - t_2)\right) p(w) dw = \mathbb{E}_w \left[\xi_w(t_1)\xi_w(t_2)^*\right] =$$

$$= \mathbb{E}_w \left[\cos(w(t_1 - t_2))\right] = \mathbb{E}_w \left[\cos(wt_1)\cos(wt_2) + \sin(wt_1)\sin(wt_2)\right] \approx$$

$$\approx \frac{1}{d} \sum_{i=1}^{d} \cos(w_i t_1)\cos(w_i t_2) + \sin(w_i t_1)\sin(w_i t_2);\ w_1, \ldots, w_d \sim p(w)$$

$$F_d(t) = \sqrt{\frac{1}{d}} \left[\cos(w_1 t), \sin(w_1 t), \ldots, \cos(w_d t), \sin(w_d t)\right],$$

Claim 1

Let $p(w)$ be the corresponding probability measure stated in Bochner's Theorem for kernel function K. Suppose the feature map F is constructed using samples $\{w_i\}_{i=1}^{d}$, then we only need $d = \Omega\left(\frac{1}{\epsilon^2} \log \frac{\sigma_p^2 t_{\max}}{\epsilon}\right)$ samples to have

$$\sup_{t_1, t_2 \in T} \left| F_d(t_1)^T F_d(t_2) - K(t_1, t_2) \right| < \epsilon$$

with any probability $\forall \epsilon > 0$ where σ_p^2 is the second momentum with respect to $p(w)$

Inductive Representation Learning on Temporal Graphs

Notation

- v_i is a vertex
- x_i is corresponding feature vector
- $\hat{h}_i^l(t)$ output for node i at time t from the l'th layer
- $N(v_0; t) = \{v_1, \ldots, v_N\}$ – neighborhood for node v_0 at time t.
  $v_0, v_i \in N(v_0; t) \Leftrightarrow (v_0, v_i, t_i) \in G, t_i < t$

temporal graph attention layer (TGAT layer)

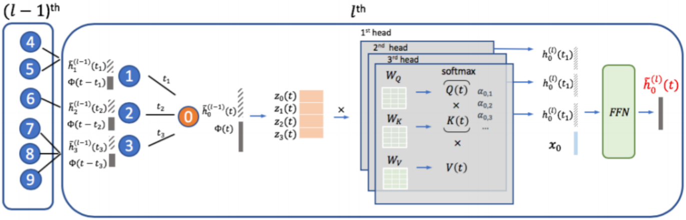
$$Z(t) = \left[\hat{h}_0^{l-1}(t) || F_{d_T}(0), \ldots, \hat{h}_N^{l-1}(t_N) || F_{d_T}(t - t_N)\right]$$

$$q(t) = [Z(t)]_0 \, W_Q, K(t) = [Z(t)]_{1:N} \, W_K, V(t) = [Z(t)]_{1:N} \, W_V$$

$$\alpha_i = \exp\left(q^T K_i\right) / \left(\sum_q \exp\left(q^T K_q\right)\right)$$

$$h(t) = \mathrm{Attn}\left(q(t), K(t), V(t)\right) \in \mathbb{R}^{d_h}$$

Inductive Representation Learning on Temporal Graphs



temporal graph attention layer (TGAT layer)

$$\hat{h}_0^l(t) = \text{FFN}\left(h(t)||x_0\right) = \text{ReLU}\left(\left[h(t)||x_0\right] W_0^l + b_0^l\right) W_1^l + b_1^l,$$

$$W_0^l \in \mathbb{R}^{(d_h+d_0)\times d_f},\ W_1^l \in \mathbb{R}^{d_f\times d},\ b_0^l \in \mathbb{R}^{d_f},\ b_1^l \in \mathbb{R}^{d}$$

Inductive Representation Learning on Temporal Graphs

Dataset	Reddit		Wikipedia		Industrial	
Metric	Accuracy	AP	Accuracy	AP	Accuracy	AP
GAT	89.86 (0.2)	95.37 (0.3)	82.36 (0.3)	91.27 (0.4)	68.28 (0.2)	79.93 (0.3)
GAT+T	90.44 (0.3)	96.31 (0.3)	84.82 (0.3)	93.57 (0.3)	69.51 (0.3)	81.68 (0.3)
GraphSAGE	89.43 (0.1)	96.27 (0.2)	82.43 (0.3)	91.09 (0.3)	67.49 (0.2)	80.54 (0.3)
GraphSAGE+T	90.07 (0.2)	95.83 (0.2)	84.03 (0.4)	92.37 (0.5)	69.66 (0.3)	82.74 (0.3)
Const-TGAT	88.28 (0.3)	94.12 (0.2)	83.60 (0.4)	91.93 (0.3)	65.87 (0.3)	77.03 (0.4)
TGAT	**90.73** (0.2)	**96.62** (0.3)	**85.35** (0.2)	**93.99** (0.3)	**72.08** (0.3)	**84.99** (0.2)

Dataset	Reddit	Wikipedia	Industrial
GAE	58.39 (0.5)	74.85 (0.6)	76.59 (0.3)
VGAE	57.98 (0.6)	73.67 (0.8)	75.38 (0.4)
CTDNE	59.43 (0.6)	75.89 (0.5)	78.36 (0.5)
GAT	64.52 (0.5)	82.34 (0.8)	87.43 (0.4)
GAT+T	64.76 (0.6)	82.95 (0.7)	88.24 (0.5)
GraphSAGE	61.24 (0.6)	82.42 (0.7)	88.28 (0.3)
GraphSAGE+T	62.31 (0.7)	82.87 (0.6)	89.81 (0.3)
Const-TGAT	60.97 (0.5)	75.18 (0.7)	82.59 (0.6)
TGAT	**65.56** (0.7)	**83.69** (0.7)	**92.31** (0.3)

References

- Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, Kannan Achan.
  Inductive Representation Learning on Temporal Graphs
  International Conference on Learning Representations (ICLR), 2020.

- Giang Hoang Nguyen, John Boaz Lee, Ryan A Rossi, Nesreen K Ahmed, Eunyee Koh, and Sungchul Kim.
  Continuous-time dynamic network embeddings.
  In Companion Proceedings of the The Web Conference 2018, pp. 969–976. International World Wide Web Conferences Steering Committee, 2018.

- Bryan Perozzi, Rami Al-Rfou, and Steven Skiena.
  Deepwalk: Online learning of social representations.
  pp. 701–710. ACM, 2014.

References

- Aditya Grover and Jure Leskovec.
  node2vec: Scalable feature learning for networks.
  pp. 855–864. ACM, 2016.
- Petar Velickovic, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio.
  Graph attention networks.
  arXiv preprint arXiv:1710.10903, 2017.
- Will Hamilton, Zhitao Ying, and Jure Leskovec.
  Inductive representation learning on large graphs.
  Advances in Neural Information Processing Systems, pp. 1024–1034, 2017.

References

- Da Xu, Chuanwei Ruan, Evren Korpeoglu, Sushant Kumar, and Kannan Achan.
  Self-attention with functional time representation learning.
  Advances in Neural Information Processing Systems, pp. 15889–15899, 2019