The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Joint Learning of Evolving Links for Dynamic Network Embedding

Aakas Zhiyuli,[1,2] Xun Liang,[1] YanFang Chen,[1] Peng Shu,[2] Xiaoping Zhou[1]
1. Renmin University of China, No.59 Zhongguancun Road, Beijing, China 100872
2. Sogou, Inc., AD-Tech, No.1 Zhongguancun East Road, Beijing, China 100084
Email: {zhiyulee, xliang, xpzhou, cyf}@ruc.edu.cn; {ps-adwr,shupeng203672}@sogou-inc.com;

Introduction and Key idea

This paper studies the problem of learning node embeddings (a.k.a. distributed representations) for dynamic networks. The embedding methods allocate each node in network with a d-dimensions vector, which can generalize across various tasks, such as item recommendation, node labeling, and link prediction (Goyal and Ferrara 2017). In practice, many real-world networks are evolving with nodes/links added or deleted. However, most of the proposed methods are focusing on static networks (Perozzi and et al. 2014; Tang and et al. 2015; Grover and Leskovec 2016). Although some previous researches (Jian and et al. 2017; Aakas and et al. 2017) have shown some promising results to handle the dynamic scenario, they just considered the added links and ignored the deleted ones.

In this work, we designed a joint learning of added and deleted links model, named RDEM, for dynamic network embedding. Our fundamental idea was inspired by a famous law, called "**Law of conservation of mass**". In RDEM, we have two types of graphs: Dynamic Memory Network (DMN) and Real Observed Network (RON). We supposed that links deleted from RON can transfer to newly added links to DMN. As an illustration, let E_T denotes the observed links with timestamp T in RON , and $\bar{E}_T$ denotes the collection of evolving links with timestamp 0 to T in DMN. For each timestamp, we have

$$E_T + \Delta E^{+}_{T\to T+1} - \Delta E^{-}_{T\to T+1} = E_{T+1} \tag{1}$$

$$\bar{E}_T + \Delta E^{-}_{T\to T+1} - \Delta E^{+}_{T\to T+1} = \bar{E}_{T+1} \tag{2}$$

here, $\Delta E^{+}_{T\to T+1}$ and $\Delta E^{-}_{T\to T+1}$ are the added and deleted collection of links from timestamp T to $T+1$. By considering formula (1) and (2), we can get

$$E_T + \bar{E}_T = E_{T+1} + \bar{E}_{T+1} \tag{3}$$

which can be further extended to

$$\mathrm{InitialState} = E_0 = E_N + \bar{E}_N, N = 1, 2, 3, \cdots \tag{4}$$

Equation (4) demonstrates that if we can preserve both of the added links (in RON) and deleted links (in DMN) at each timestamp, we can backtrack the deleted information of networks to an initial state. We called this "**Conservation of dynamic links**".

Algorithm 1 RDEM (Update $T+1$ from T)

Input: $G_T,\ G_{T+1}, \bar{G}_T, v_i^T, \bar{v}_j^T$
Output: $v_k^{T+1}, \bar{v}_l^{T+1}\ \left(k \in V_{T+1}, l \in \bar{V}_{T+1}\right)$
1: Duplicate $v_i^T \to v_k^{T+1};\ \bar{v}_j^T \to \bar{v}_l^{T+1}$
2: **for** edge(a,b) **in** $\Delta E^{+}_{T\to T+1}$ **do**
3: **LS** $\left(G_{T+1}\left(\text{edge}\left(a,b\right)\right)\right) \xrightarrow{\text{update}} v_k^{T+1}$
4: **for** edge(c,d) **in** $\Delta E^{-}_{T\to T+1}$ **do**
5: **LS** $\left(\bar{G}_{T+1}\left(\text{edge}\left(c,d\right)\right)\right) \xrightarrow{\text{update}} \bar{v}_l^{T+1}$
6: **for** node x **in** $\bar{V}_{T+1} \cap V_{T+1}$ **do**
7: **Mean** $\left(\bar{v}_x^{T+1}, v_x^{T+1}\right) \xrightarrow{\text{update}} \bar{v}_x^{T+1}, v_x^{T+1}$

Model

In this section, we introduce the framework of RDEM. Given an observed network with a series of timestamps $G_T = (V_T, E_T),\ T = 0, 1, 2, \cdots$. The goal of RDEM is mapping the nodes in V_T into a vector space with d-dimensions $V_T \to v_i^T \in \mathbb{R}^d\ \ (i \in V_T)$. As showed in Figure 1, RDEM has two main steps: the initial step and online learning step.

Initial Step. The input of initial step is network G_0, which can be regarded as a static network. Therefore, existing methods, such as DeepWalk (Perozzi and et al. 2014), LINE (Tang and et al. 2015), or our previous work (Aakas and et al. 2017), can be used for pre-training the node vectors.

Online Learning Step. In real-world, networks change very quickly over time through the addition of fresh nodes and links or the reduction of existing ones, which are referred to dynamic evolution of networks. Therefore, we designed an online learning step to meet the actual needs. As showed in Alg.1, we present the updating logic of node vectors in RON (G) and DMN ($\bar{G}$) from timestamp T to $T+1$. Line 2-5 shows the updating process of added information in RON and DMN respectively. Here, we adopt the Local Searching (LS) algorithm proposed in DNPS model (Aakas and et al. 2017), which is efficient for incremental learning. Finally, (line 6-7) we update the common node of RON (G_{T+1}) and DMN ($\bar{G}_{T+1}$) with the mean value of node vectors that they have learned in line 2-5, which means that for a node $x \in V_T \cap \bar{V}_T$, we have $v_x^T = \bar{v}_x^T$.

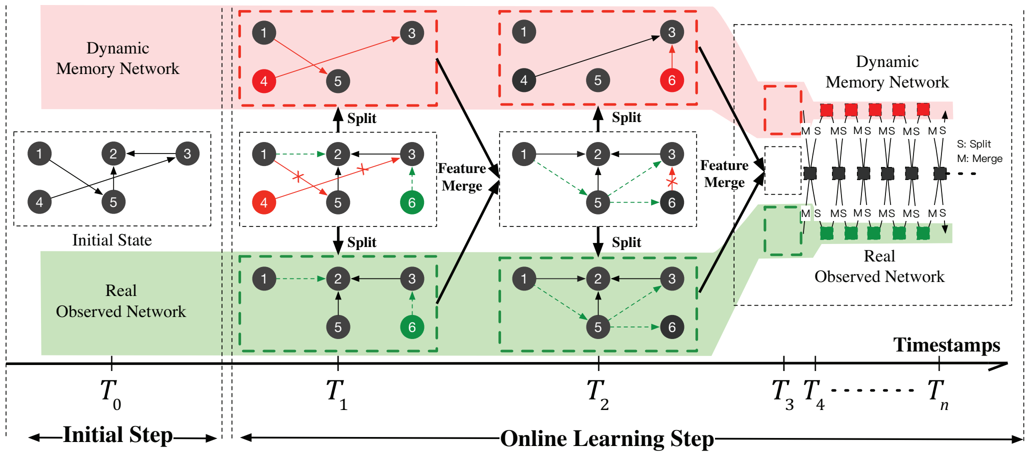

Figure 1: An instance of RDEM model. The red and green line indicates the deleted and added links at timestamp T and $T+1$. The node vectors of DMN and RON are updated in parallel during the training process (described in Alg.1). Note that, after each "feature merges" process, the common node in DMN and RON will keep the same vectors.

Table 1: Results of future link prediction (AUC%).

T.	SCnet			Gplus		
	RDEM	DW	LINE	RDEM	DW	LINE
1	**76.35**	74.53	72.75	**82.35**	79.30	79.98
2	**78.22**	75.32	73.20	**83.96**	81.82	81.60
3	**81.03**	77.14	74.86	**85.23**	82.47	82.64

Experiments

To investigate the effectiveness of our model, we conducted the experiment on two large-scale networks. We chose dynamic link prediction task as the baseline task. The first network is a click based graph (SCnet) which is provided by Sogou-Inc (AD-Tech research group). The undirected links with timestamp indicate the UserA-UserB (same clicked item) relationship formed. The second network is Google+ network (Gong and et al. 2012), in which the directed link corresponds to the social connections between users.

The preliminary results of the comparison between the proposed approach and two baseline methods are shown in Table 1. We can see that RDEM outperforms DeepWalk and Line with different networks in the meantime. This results demonstrate that we can not ignore the deleted links while learning embeddings for dynamic networks. In addition, we can observe the difference of AUC between RDEM and baseline methods becoming larger from timestamp 1 to 3. This observation aligns with our intuition that with more missing links learned, the model will become more robust.

Conclusions and Future Work

In this paper, we proposed RDEM, to learn the structural features of nodes in dynamic networks. We designed a dynamic memory network (DMN) to reserve the feature of deleted links while network evolving. Our experimental results showed that the RDEM approach works well in dynamic (with link added and deleted) scenario compared with baselines. With well learned node vectors of dynamic network, problems such as dynamic link prediction, dynamic community discovery, or even dynamic opinion leader prediction, can be solved faster and effective, especially the network is very large and unstable.

Acknowledgments

This work is supported by the National Natural Science Foundation of China Grant No. 71531012 and 71271211, and the Outstanding Innovative Talents Cultivation Funded Programs 2017 of Renmin University of China. Xun Liang is the corresponding author of this paper.

References

Aakas, Z., and et al. 2017. Learning distributed representations for large-scale dynamic social networks. In *Proc. of the 36th Annual IEEE Inter. Conf. on Com. Comm.*, 1–9.

Gong, N. Z., and et al. 2012. Evolution of social-attribute networks: measurements, modeling, and implications using google+. In *Proc. of the 2012 ACM Conf. on Internet Meas. Conf.*, 131–144.

Goyal, P., and Ferrara, E. 2017. Graph embedding techniques, applications, and performance: a survey. *arXiv preprint arXiv:1705.02801*.

Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proc. of the 22nd ACM SIGKDD Inter. Conf. on Know. Disc. & Data Min.*, 855–864.

Jian, L., and et al. 2017. Toward online node classification on streaming networks. *Data Min. and Know. Disc.* 1–27.

Perozzi, B. a., and et al. 2014. Deepwalk: Online learning of social representations. In *Proc. of the 20th ACM SIGKDD Inter. Conf. on Know. Disc. & Data Min.*, 701–710.

Tang, J., and et al. 2015. Line: Large-scale information network embedding. In *Proc. of the 24th Inter. Conf. on WWW*, 1067–1077.