Improving the Robustness of Wasserstein Embedding by Adversarial PAC-Bayesian Learning

Daizong Ding,[1] **Mi Zhang,**[*1] **Xudong Pan,**[1] **Min Yang,**[1] **Xiangnan He**[2]
[1]School of Computer Science, Fudan University
[2]School of Information Science and Technology, University of Science and Technology of China
{17110240010, mi_zhang, 18110240010, m_yang}@fudan.edu.cn, xiangnanhe@gmail.com

Abstract

Node embedding is a crucial task in graph analysis. Recently, several methods are proposed to embed a node as a distribution rather than a vector to capture more information. Although these methods achieved noticeable improvements, their extra complexity brings new challenges. For example, the learned representations of nodes could be sensitive to external noises on the graph and vulnerable to adversarial behaviors. In this paper, we first derive an upper bound on generalization error for Wasserstein embedding via the PAC-Bayesian theory. Based on this, we propose an algorithm called Adversarial PAC-Bayesian Learning (APBL) in order to minimize the generalization error bound. Furthermore, we provide a model called Regularized Adversarial Wasserstein Embedding Network (RAWEN) as an implementation of APBL. Besides our comprehensive analysis of the robustness of RAWEN, our work for the first time explores more kinds of embedded distributions. For evaluations, we conduct extensive experiments to demonstrate the effectiveness and robustness of our proposed embedding model compared with the state-of-the-art methods.

Introduction

Node embedding plays an increasingly significant role in modern graph analysis. The effectiveness of the embeddings largely influence the results of downstream machine learning tasks, e.g. link prediction and node classification. Traditionally, node embedding is modeled as low-dimensional vector representations of nodes in a given graph (Goyal and Ferrara 2018), where the similarity between nodes is represented as the distance between embedded vectors, e.g. Euclidean distance (Tang et al. 2015; Wang, Cui, and Zhu 2016).

Recently, some methods have been proposed to embed each node with a distribution e.g. multivariate Gaussian distribution (He et al. 2015), rather than a real vector. In distribution-based embedding models, the similarity between nodes is represented as the distance between embedded distributions such as Wasserstein distance (Zhu et al. 2018). Compared with real vectors, distributions contain more information of the graph. For instance, the mean of the distribution represents the position of the node and the variance reflects the degree of the node (Bojchevski and Günnemann 2018). Although empirical results prove that these methods are more effective than the vector-based embedding model, there exist two open problems.

The first problem is the certification of *robustness*. According to previous researches (Zügner, Akbarnejad, and Günnemann 2018), small perturbations can largely influence the embedding model. For instance, the distance between two nodes may change sharply when few random edges are added to the graph. In order to restore robustness against noises, some previous efforts have been made on the vector-based embedding model. For example, Dai et al. (2018a) propose to regularize the node embeddings with prior information. However, due to the complexity of how to impose prior knowledge on distributions, the research to date has not yet provided a robust model when embeddings of nodes are distributions.

The second problem is, most existing distribution-based embedding models simply used the Gaussian distribution and little is known for the effectiveness of other distributions. Although there were several studies to extend the vector-based embedding model to other vector space such as hyperbolic space (Chamberlain, Clough, and Deisenroth 2017), there were rarely any attempts to examine the effectiveness of other distributions such as Dirichlet distribution which is widely applied in graph analysis (Xie, Kelley, and Szymanski 2013). One of the main obstacles is the technical difficulty on tackling distance between distributions. For instance, although the KL divergence or Wasserstein distance between Gaussian distributions has a closed form, it becomes intractable when other kinds of distributions are considered.

In this paper, to address two open problems in the distribution-based embedding, we focus on Wasserstein embedding and propose to minimize the expected loss rather than the empirical loss to alleviate the influence of additional noises. With the aid of the PAC-Bayesian theory, we first derive an upper bound for the expected loss. Then we propose an algorithm called Adversarial PAC-Bayesian Learning (APBL) for minimizing the upper bound. Moreover, we

[*]Corresponding author

propose a neural network based instance of APBL called Regularized Adversarial Wasserstein Embedding Network (RAWEN), the robustness of which is further certificated by our analytic results. Moreover, we are the first to investigate the effectiveness of more kinds of embedded distributions such as Dirichlet distribution and truncated Gaussian distribution. Finally, we conduct extensive experiments to validate the effectiveness and robustness of RAWEN, compared with the state-of-the-art node embedding methods.

Preliminaries

In this section we briefly introduce some useful notations in node embedding and present the framework of Wasserstein embedding.

Node Embedding Problem

We first introduce the basic notations in node embedding. We focus on an undirected graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ are nodes, $\mathcal{E}$ are edges and $N = |\mathcal{V}|$, i.e. the number of nodes. Edges are represented by the adjacency matrix $E \in \{0,1\}^{N\times N}$ such that $e_{ij} = 1$ means there exists an edge between v_i and v_j, while $e_{ij} = 0$ means not.

We use $y_{ij} \in \{0, 1\}$ to denote the linkage relation between nodes. To clarify, linkage relation is a kind of artificial signal of supervision, which is mainly constructed from the observed edges according to a certain predefined criterion. For instance, Tang et al. (2015) uses the two-order proximity on graph to construct y_{ij}, i.e. if both $e_{ik} = 1$ and $e_{jk} = 1$ are satisfied, then $y_{ij} = 1$. Ω is the set of observed linkage relation y_{ij} with size M.

Given a metric space $\mathcal{Z} \subseteq \mathbb{R}^K$, for $z_i, z_j \in \mathcal{Z}$, the distance metric is denoted as d_{ij}. The goal of node embedding is to relate each node $v_i \in \mathcal{V}$ with a latent representation $z_i \in \mathcal{Z}$, and learn effective $Z = \{z_1, \cdots, z_N\} \subseteq \mathcal{Z}$ given the observations Ω. The optimization objective is to minimize the following empirical loss function

$$\hat{\ell}(Z, \Omega) = \frac{1}{M} \sum_{y_{ij}\in\Omega} \ell(d_{ij}, y_{ij}) \tag{1}$$

where ℓ is a certain loss function for embedding learning. In existing state-of-the-art models, the choice of ℓ is usually the binary cross entropy (Perozzi, Al-Rfou, and Skiena 2014; Tang et al. 2015; Grover and Leskovec 2016; Wang, Cui, and Zhu 2016),

$$\ell(d_{ij}, y_{ij}) = -y_{ij} \log \sigma(d_{ij}) - (1-y_{ij}) \log(1-\sigma(d_{ij})) \tag{2}$$

where the activation function is $\sigma(d) : \mathbb{R} \to [0, 1]$.

Wasserstein Embedding

Recently, some researches attempt to use distributions to represent embeddings rather than real vectors (He et al. 2015; Bojchevski and Günnemann 2018). Specifically, they suppose a metric space $\mathcal{Z} \subseteq \mathbb{R}^K$, where each node $v_i \in \mathcal{V}$ is related with a distribution $q(z|v_i)$ defined on $\mathcal{Z}$. Then the distance d_{ij} can be defined as the distance between two distributions $q(z|v_i)$ and $q(z|v_j)$, e.g., Kullback-Leibler (KL) divergence and Wasserstein distance. Previous work found that the Wasserstein distance is more effective in node embedding problem (Zhu et al. 2018). Therefore in this paper we focus on the latter kind of distance,

$$d_{ij} = \inf_{\Lambda\in\Pi(q(z|v_i),q(z|v_j))} \mathbb{E}_{(z_i,z_j)\sim\Lambda}[c(z_i, z_j)] \tag{3}$$

where $\Pi(q(z|v_i), q(z|v_j))$ is a set of joint distributions on $\mathbb{R}^K \times \mathbb{R}^K$ s.t. the element of which has marginal distributions equal to $q(z|v_i)$ and $q(z|v_j)$. The cost function $c : \mathbb{R}^K \times \mathbb{R}^K \to \mathbb{R}_+$ is a predefined function, e.g. the Euclidean distance.

Due to the complexity of the distance form, current methods only considered the embedded distributions as Gaussian. It is mainly because distance like KL-divergence or Wasserstein metric has an analytic form for Guassian distributions. Specifically, in the Gaussian approach, each node v_i is related with the distribution $q(z|v_i) = \mathcal{N}(\mu_i, \Sigma_i \boldsymbol{I})$, where $\mu_i \in \mathbb{R}^K$, $\Sigma_i \in \mathbb{R}^K$ are respectively the mean and variance vectors. Compared with the traditional vector-based embedding methods, distributions contain more information. For example, Zhu et al. (2018) interprets the mean μ_i as the position and the variance Σ_i as the degree of node v_i, which enables the model to learn more effective embeddings. However, there still exist several open problems:

- Embeddings of nodes can be easily influenced by external noises on observations Ω. (Zügner, Akbarnejad, and Günnemann 2018). Currently most vector-based embedding methods impose prior information on embedded space to improve the robustness (Dai et al. 2018a). However, due to the inherent complexity of distribution-based embedding, it is technically hard to impose such information for distribution-based methods.
- The choice of embedding space is only the family of Gaussian distributions. The effectiveness of Wasserstein embedding with other kinds of distributions is far from well-studied.

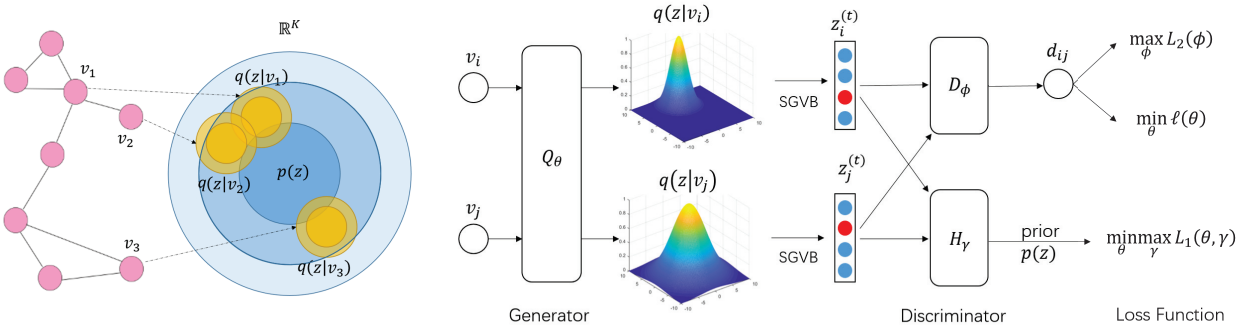
In the remainder of this paper, we present our novel methods to tackle these concerns of Wasserstein embeddings.

Proposed Method

In this section we propose to minimize the expected loss rather than the empirical loss to alleviate the influence of additional noises. To achieve this, we first study the expected loss in Wasserstein embedding and derive an upper bound of the expected loss when the optimized Wasserstein distance is given. Then we propose an adversarial PAC-Bayesian learning strategy to infer the parameters in both the Wasserstein distance and the embedded distributions. As a comprehensive solution, we present Regularized Adversarial Wasserstein Embedding Network (RAWEN) with different kinds of embedded distributions.

Optimizing the Expected Loss

If we only minimize the empirical loss in Eq. 1, additional noises or perturbance on Ω can easily let the model learn wrong information and therefore perform poorly even when the empirical loss is small (Zügner, Akbarnejad, and



(a) Prior on distribution based embedding.

(b) Structure of proposed RAWEN.

Figure 1: Overview of the proposed framework.

Günnemann 2018). Therefore we focus on the loss calculated on the whole dataset rather than that only on the observations. Formally, we concerntrate on the following expected loss

$$\ell(Q) = \mathbb{E}_{y_{ij} \sim P_Y}[\ell(d_{ij}, y_{ij})] \tag{4}$$

where P_Y is the unknown distribution over y_{ij} and $Q = \{q(z|v_1), \cdots, q(z|v_N)\}$. The expected loss measures the effectiveness of embeddings not only on observations Ω but also on those unseen data. Therefore those noises on observations Ω will have less influence on the model if we minimize the expected loss. However, the term is difficult to be directly computed due to the unseen linkage information on the graph. In this paper, one of our main contributions is the following theorem that bounds the difference between the expected loss and empirical loss for Wasserstein embedding.

We first rewrite the Wasserstein distance with its dual form (Villani 2003):

$$d_{ij} = \sup_{\|D\|_L \leq 1} \mathbb{E}_{z \sim q(z|v_i)}[D(z)] - \mathbb{E}_{z \sim q(z|v_j)}[D(z)] \tag{5}$$

where function $D : \mathbb{R}^K \to \mathbb{R}$ is a 1-Lipschitz function. Then we prove the following theorem under the framework of PAC-Bayesian. A detailed proof can be found in Appendix.

Theorem 1. *Given a graph with N nodes, $\Omega = \{(v_i, v_j, y_{ij})\}$ is the observed linkage information sampled from P_Y with size $M \geq 8$. Suppose $\mathcal{Z} \subseteq \mathbb{R}^K$ is a metric space, and $\mathcal{Q}$ is a family of distribution defined on $\mathcal{Z}$. Each node v_i is related with a distribution $q(z|v_i) \in \mathcal{Q}$, where distance between embedded distributions d_{ij} is Wasserstein distance. Loss function $\ell(d_{ij}, y_{ij}) \in [0, 1]$ is pre-defined. Then given optimal function D^* defined in the dual form of Wasserstein distance, prior distribution $p(z)$ defined on $\mathcal{Z}$ and $\delta \in (0, 1)$, with probability at least $1 - \delta$ we have for all $Q = \{q(z|v_1), \cdots, q(z|v_N)\} \subseteq \mathcal{Q}$:*

$$\ell(Q) \leq \hat{\ell}(Q, \Omega) + \frac{\sum_{i=1}^N KL(q(z|v_i)\|p(z)) + \ln M - \ln \delta}{M} \tag{6}$$

where $\hat{\ell}(Q, \Omega)$ is the empirical loss on Ω and the distance d_{ij} between node v_i and v_j is calculated as,

$$d_{ij} = \mathbb{E}_{z \sim q(z|v_i)}[D^*(z)] - \mathbb{E}_{z \sim q(z|v_j)}[D^*(z)] \tag{7}$$

Intuitively, this theorem states that, if the optimal function D^* in the Wasserstein distance is given, the expected loss in the Wasserstein embedding can be approximated by the sum of the empirical error and an additional generalization error term, where the latter one is proportional to the KL divergence between the embedded distribution $q(z|v_i)$ and prior distribution $p(z)$.

Adversarial PAC-Bayesian Learning

Based on Theorem 1, we should optimize both the generalization bound and the function D in the Wasserstein distance simultaneously. To solve this problem, we propose an adversarial training strategy below

$$\max_D \min_Q \ell(Q) + \lambda_2 \ell(D) \tag{8}$$

where

$$\ell(Q) = \hat{\ell}(Q, \Omega) + \lambda_1 \sum_{i=1}^N KL(q(z|v_i)\|p(z))$$
$$\ell(D) = \sum_{y_{ij} \in \Omega} \mathbb{E}_{z \sim q(z|v_i)}[D(z)] - \mathbb{E}_{z \sim q(z|v_j)}[D(z)] \tag{9}$$

The general idea behind our adversarial learning process is summarized as follows. On one hand, given the optimal function D^*, we can minimize the expected loss $\ell(Q)$ by the upper bound in Theorem 1. On the other hand, we can derive an optimal function D^* by maximizing $\ell(D)$ given optimal $q(z|v_i)$. Therefore if we update Q and D by alternatively minimizing $\ell(Q)$ and maximizing $\ell(D)$, we can derive an approximated solution to this problem. We name the proposed adversarial learning process as adversarial PAC-Bayesian learning (APBL).

Regularized Adversarial Wasserstein Embedding Network

However, there still exist two challenges during implementation: (1) How to calculate the expectation term in the Wasserstein distance? (2) How to minimize the KL divergence term? To tackle, we propose a neural network based

solution called Regularized Adversarial Wasserstein Embedding Network (RAWEN) as follows.

We define a function $Q_\theta : \mathcal{V} \to \mathcal{Z}$, where for each node $v_i \in \mathcal{V}$, the output is a distribution $q(z|v_i) = Q_\theta(v_i)$. In practice, Q_θ is implemented as a multi-layer neural network, whose outputs are the parameters of the distribution $q(z|v_i)$. Then for the first challenge, the distance can be approximated by

$$d_{ij} \approx \frac{1}{T}\sum_{t=1}^{T}\left[D_\phi(z_i^{(t)}) - D_\phi(z_j^{(t)})\right] \tag{10}$$

where $z_i^{(t)}$ and $z_j^{(t)}$ are sampled independently from $q(z|v_i) = Q_\theta(v_i)$ and $q(z|v_j) = Q_\theta(v_j)$. As the complexity of the model would increase exponentially if we use independent function D for each pair, we propose to approximate the functions by $D_\phi : \mathcal{Z} \to \mathbb{R}$, which is a multi-layer neural network parameterized by ϕ. For sampling, we use Stochastic Gradient Variational Bayesian (SGVB) estimator with the reparameterization trick (Kipf and Welling 2016b).

For the second challenge brought by the intractable KL divergence term, we propose to incorporate a GAN-like loss (Goodfellow et al. 2014)

$$L_1(\theta,\gamma) = \frac{1}{NT}\sum_{v_i\in\mathcal{V}}\sum_{t=1}^{T}\log\left(1 - H_\gamma(z_i^{(t)})\right) + \log H_\gamma(z^{(t)}) \tag{11}$$

where $H_\gamma(z) : \mathcal{Z} \to [0,1]$ is a multi-layer neural network parameterized by γ and $z_i^{(t)}$ and $z^{(t)}$ are independently sampled from $q(z|v_i)$ and $p(z)$ by SGVB respectively. Based on the definition above, the adversarial loss function in Eq. 8 can be written as

$$\min_\theta \max_{\gamma,\phi} \hat{\ell}(\theta) + \lambda_1 L_1(\theta,\gamma) + \lambda_2 L_2(\phi) \tag{12}$$

where $\hat{\ell}(\theta) = \frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(d_{ij}, y_{ij})$ is the empirical loss for θ, and $L_2(\phi) = \frac{1}{M}\sum_{y_{ij}\in\Omega} d_{ij}$. To regularize function D as 1-Lipschitz, we add term $\|\nabla_\phi d_{ij} - 1\|^2$ to L_2. Fig. 1b illustrates the overall structure of RAWEN, with the detailed learning process in Alg. 1. In summary, there are two kinds of adversarial learning processes in this model

- **Adversarial PAC-Bayesian Learning:** We can regard Q_θ as a generator and D_ϕ as a discriminator. The generator minimizes the approximated expected loss for embeddings based on the given Wasserstein distance, while the discriminator tries to learn the correct Wasserstein distance.
- **GAN Loss for KL Divergence:** We can regard Q_θ as a generator and H_γ as a discriminator. The generator minimizes the KL divergence term based on the given function H_γ, while the discriminator tries to learn the correct KL divergence.

SGVB for Different Distributions

With the aid of our approximation to the Wasserstein distance and KL divergence, the embedding space $\mathcal{Q}$ in RAWEN is not necessarily limited to Gaussian any longer. In fact, the list of applicable distributions are the same as those of SGVB (Kingma and Welling 2013). As a demonstration, we also implement two other kinds of $\mathcal{Q}$ besides the Gaussian distribution to validate the effectiveness of RAWEN, namely Dirichlet and truncated Gaussian distribution.

Algorithm 1 Adversarial Training Strategy

Require: Input Ω, regularizing coefficient λ_1, λ_2 and learning rate
repeat
  Random choose y_{ij}
  Output parameters of $q(z|v_i), q(z|v_j)$ by Q_θ
  Sample $\tilde{z}^{(t)}$ from prior $p(z)$
  Sample $z_i^{(t)}, z_j^{(t)}$ from $q(z|v_i), q(z|v_j)$ by SGVB
  Update ϕ, γ by maximizing the loss function below with the Adam optimizer (Kingma and Ba 2015),

$$\lambda_2 L_2(\phi) + \lambda_1 L_1(\theta,\gamma)$$

  Update θ by minimizing the loss function below with the Adam optimizer

$$\hat{\ell}(\theta) + \lambda_1 L_1(\theta,\gamma)$$

until Convergence

- **Gaussian Distribution:** For node v_i, the embedded distribution can be represented as $q(z|v_i) = \mathcal{N}(\mu_i, \Sigma_i \odot \boldsymbol{I})$, where the prior $p(z) = \mathcal{N}(0, \boldsymbol{I})$. The SGVB estimator with reparameterization trick (Kingma and Welling 2013) is derived as

$$z_i^{(t)} = \mu_i + \Sigma_i \odot \psi^{(t)}, \text{ where } \psi^{(t)} \sim \mathcal{N}(0, \boldsymbol{I}) \tag{13}$$

  where $\mu_i, \Sigma_i \in \mathbb{R}^K$ are outputs of $Q_\theta(v_i)$.
- **Dirichlet Distribution:** In the setting of Dirichlet distribution, the variable of $q(z|v_i)$ can be regarded as parameters of a multinomial distribution. Illustratively, in graph analysis, it can be used to describe a node's affiliations to different communities (Xie, Kelley, and Szymanski 2013). For node v_i, the embedded distribution can be represented as,

$$q(z|v_i) = \text{Dir}(\eta_1, \cdots, \eta_K) \tag{14}$$

  where the prior $p(z) = \text{Dir}(\frac{1}{K}, \cdots, \frac{1}{K})$. We use Logistic-Normal distribution to approximate the sampling in SGVB estimator, which is widely used to approximate Dirichlet distribution (Lafferty and Blei 2006). Formally,

$$z_i^{(t)} = \text{softmax}[\mu_i + \Sigma_i \odot \psi^{(t)}] \text{ , where } \psi^{(t)} \sim \mathcal{N}(0, \boldsymbol{I}) \tag{15}$$

  where $\mu_i, \Sigma_i \in \mathbb{R}^K$ are outputs of $Q_\theta(v_i)$.
- **Truncated Gaussian Distribution:** For truncated Gaussian we set the domain of $q(z|v_i)$ as $\mathbb{R}_+$, which allows the generative model to learn non-negative features (Brouwer and Lio 2017). The prior $H(z)$ in this case can be defined

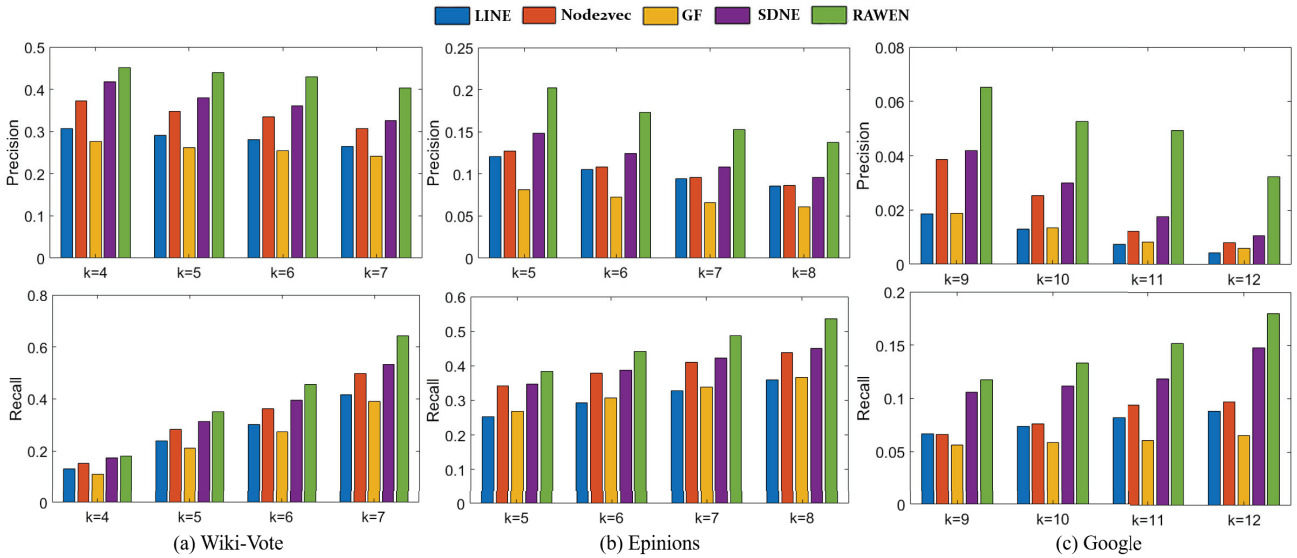


Figure 2: Evaluation of precision and recall for link prediction on three datasets.

as the truncated Gaussian with mean 0 and the standard variance. As for the SGVB estimator, for node v_i,

$$z_i^{(t)} = \sigma(\mu_i + \Sigma_i \odot \psi^{(t)}), \quad \text{where} \quad \psi^{(t)} \sim \mathcal{N}(0, \boldsymbol{I}) \quad (16)$$

We set $\phi(z) = \log(1 + e^z)$ (i.e.the softplus function) and $\mu_i, \Sigma_i \in \mathbb{R}^K$ are outputs of $Q_\theta(v_i)$. Discriminators H_γ and D_ϕ are all implemented as one-layer fully connected neural networks.

Certification of Robustness

According to the above descriptions, RAWEN can be applied to more kinds of distributions besides Gaussian, which addresses the second concerns on Wasserstein embedding. In this section, we conduct a formal analysis on the robustness of RAWEN to address the remaining one.

Let us take $\Omega = \{(v_i, v_j, y_{ij})\}$ as samples from the true distribution P_Y and $M := |\Omega|$. The number of $y_{ij} = 1$ is $\rho_M M$. By denoting the empirical distribution with its support on Ω as $\hat{P}_Y$, we view our node embedding framework as a generative model for recovering the true distribution P_Y with $\hat{P}_Y$. To introduce noises to the distribution $\tilde{P}_Y$, we randomly add noises to R samples in Ω by $\tilde{y}_{ij} = 1 - y_{ij}$. We assume for each v_i there exists a node v_j ($v_{j'}$) s.t. $y_{ij} = 1$ ($\tilde{y}_{ij'} = 1$), and a node v_k ($v_{k'}$) s.t. $y_{ik} = 0$ ($\tilde{y}_{ik'} = 0$).

Let θ^* be the optimal solution of the mapping function Q_θ after its being trained with the observations Ω in RAWEN, and $d_{ij}(\theta^*)$ is the corresponding distance. Similarly, we define $\tilde{\theta}^*$ as the optimal solution after perturbation. Suppose for each $y_{ij} = 1$ ($\tilde{y}_{ij} = 1$), we have $d_{ij}(\theta^*) \leq \delta_1$ ($d_{ij}(\tilde{\theta}^*) \leq \tilde{\delta}_1$). While for each $y_{ij} = 0$ ($\tilde{y}_{ij} = 0$), we have $d_{i,j}(\theta^*) \geq \delta_0$ ($d_{i,j}(\tilde{\theta}^*) \geq \tilde{\delta}_0$). Here δ_1 and $\tilde{\delta}_1$ are chosen to be sufficiently small s.t. $\delta_1 \ll \delta_0$, $\tilde{\delta}_1 \ll \tilde{\delta}_0$.

We start our analysis by providing a data-dependent definition of *path* as an extension of its conventional notion in classical graph theory, alongside a main assumption which will be used during our proofs.

Definition 1 (Path w.r.t. Ω)**.** *A path of length Δ from v_i to v_j w.r.t. Ω is defined as a $(\Delta+1)$-length sequence of nodes $(v_{i_0}, \ldots, v_{i_\Delta})$ ($i_0 := i$, $i_\Delta := j$) s.t. for any $1 \leq k \leq \Delta$, $y_{i_{k-1} i_k} = 1$.*

Assumption 1. *If $y_{ij} = 1$ and $d_{ij}(\theta^*) < \hat{\delta}_1$, then there exists a path from v_i to v_j w.r.t. Ω.*

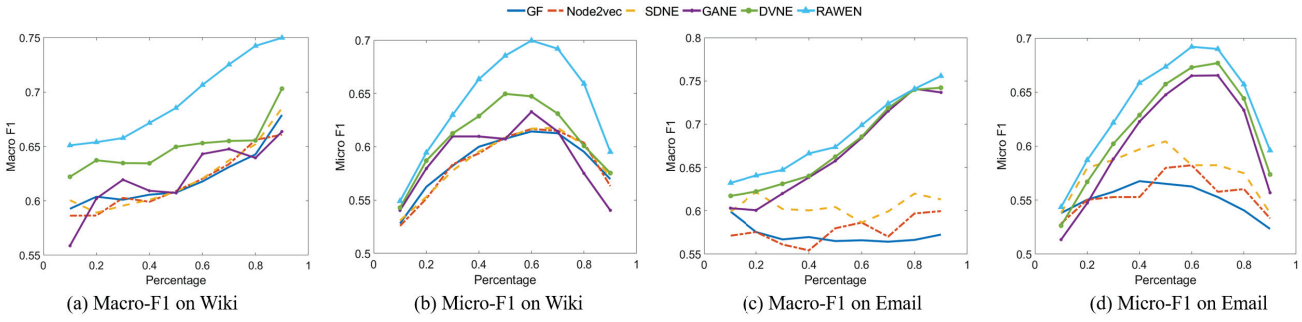
This assumption has been supported by many reported empirical results on graph analysis. For example, a previous work using path-based similarity to predict potential links on graph (Katz 1953). Another justification comes from several experiments which states that node embeddings learned from path-related observations can always achieve satisfying results on a wide range of graph-based tasks (Jeh and Widom 2002; Cao, Lu, and Xu 2015; Tsitsulin et al. 2018).

With preparations above, we are now able to state our main theorems, which provide a quantitative certification on the robustness of our proposed node embedding framework. Detailed proofs can be found in Appendix.

Theorem 2. *Under Assumption 1, for an arbitrary pair of nodes v_i, v_j s.t. $y_{ij} = 1$, if $d_{ij}(\theta^*) \leq \hat{\delta}_1$, then the expectation of $d_{ij}(\tilde{\theta}^*)$ satisfies*

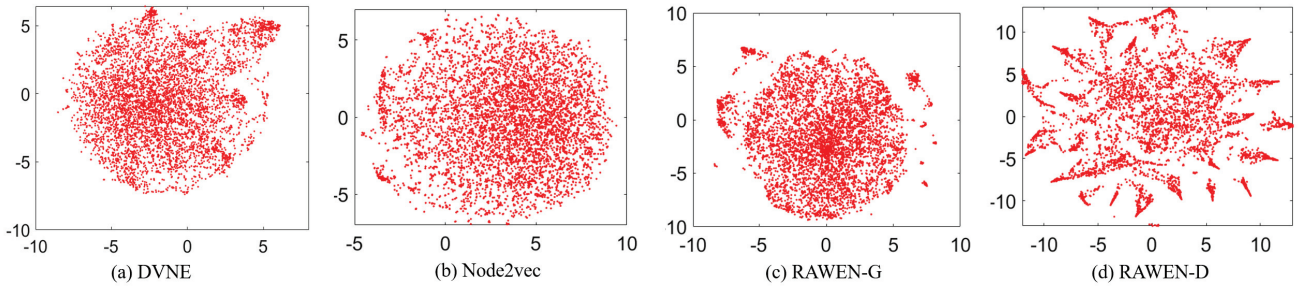
$$\mathbb{E}_{\tilde{P}_Y}[d_{i,j}(\tilde{\theta}^*)] \leq \mathbb{E}_{P_Y}[\Delta] \cdot \Big[\frac{R}{2\rho_M M}\sqrt{2J} + (1 - \frac{R}{2\rho_M M}) \cdot \tilde{\delta}_1\Big] \quad (17)$$

where $\mathbb{E}_{P_Y}[\Delta]$ is the expectation of the length of path in P_Y and $J = \mathbb{E}_{p(z)}[\|z\|^2] < \infty$.



(a) Macro-F1 on Wiki (b) Micro-F1 on Wiki (c) Macro-F1 on Email (d) Micro-F1 on Email

Figure 3: Macro- and Micro-F1 value for classification task.



(a) DVNE (b) Node2vec (c) RAWEN-G (d) RAWEN-D

Figure 4: Visualization of node embeddings with t-SNE.

Theorem 3. *Under Assumption 1, for an arbitrary pair of nodes v_i, v_j s.t. $y_{ij} = 0$, if $d_{ij}(\theta^*) \geq \hat{\delta}_0$, then the expectation of $d_{ij}(\tilde{\theta}^*)$ satisfies*

$$\mathbb{E}_{\tilde{P}_Y}[d_{ij}(\tilde{\theta}^*)] \geq \hat{\delta}_0 \cdot (1 - \frac{R}{2(1-\rho_M)M}) \qquad (18)$$

Evaluations

In this section, we compare our proposed RAWEN with the state-of-the-art node embedding methods in terms of effectiveness and robustness. In particular, our main research questions are

- **RQ1.** Is RAWEN an effective node embedding method?
- **RQ2.** Is RAWEN a robust node embedding method?
- **RQ3.** What is the influence of different $\mathcal{Q}$?

Experiment Settings

We validate the effectiveness of our embedding on two benchmark tasks, i.e. multi-label node classification and link prediction tasks. For each run of experiment, we conduct a 10-fold cross validation on each dataset and report the average results. We validate the expressiveness of our node embedding framework on the following public graph datasets of various scale. For link prediction, we use Wiki-Vote, Epinions and Google datasets, which respectively contain 2846, 5488, 44000 nodes and 184376, 279480, 445618 edges. For node classification we use Email and Wiki dataset, which respectively contain 1005, 19933 nodes and 25571, 1003686 edges.

We compare the performance of RAWEN with several state-of-the-art node embedding methods. For vector-based embedding model, we choose Graph Factorization (GF) (Ahmed et al. 2013), Large-scale Information Network Embedding (LINE) (Tang et al. 2015), Node2vec (Grover and Leskovec 2016) and Structural Deep Network Embeddings (SDNE) (Wang, Cui, and Zhu 2016) as baselines. Furthermore, we compare RAWEN with Generative Adversarial Network Embedding (GANE) (Hong, Li, and Wang 2018), which is a robust vector-based embedding models. For distribution-based embedding models, we choose deep variational network embedding in Wasserstein space (DVNE) (Zhu et al. 2018) as the baseline. We also conduct self-comparisons among different choices of $\mathcal{Q}$, namely RAWEN-G, RAWEN-D and RAWEN-T which respectively stand for Gaussian, Dirichlet and Truncated Gaussian.

Similar to the evaluations in node2vec, we construct the observed relations for training node embeddings with the following criterion, i.e. $y_{ij} = 1$ if v_i, v_j are in a common random walk path; otherwise, $y_{ij} = 0$. For each methods, we set the embedding size as 20, 30, 50 for Wiki-vote, Epinions and Google respectively and set the batch size as 200 in each case. For our model, the learning rate is set as 0.002, sample size T as 10 and the regularization coefficient as 0.1.

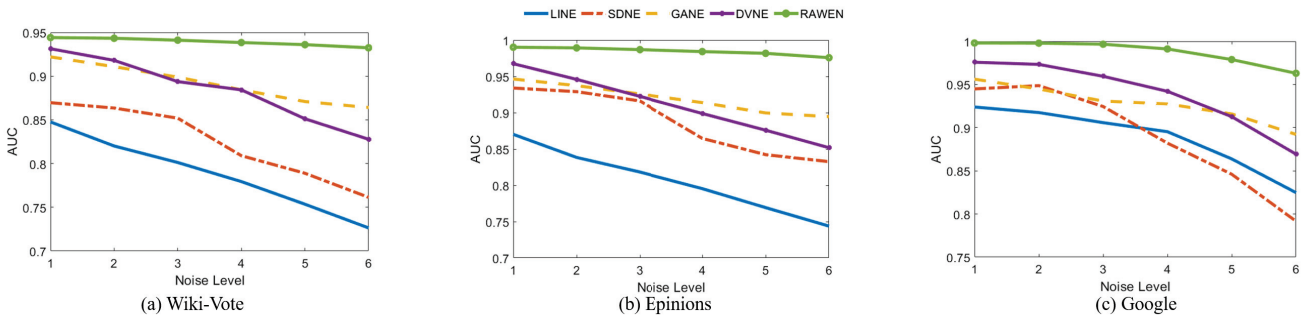


Figure 5: AUC results on link prediction tasks after repeatedly replacing an existing edge in the training set with a random edge. We set six noise levels, i.e. by replacing 3%, 5%, 8%, 12%, 15%, 18% edges in the training set.

Empirical Results

For RQ1, we present the precision and recall results on the link prediction tasks in Fig. 2. For baseline models, Node2vec, LINE and GF are worse than SDNE due to their commonly shallow architectures. With the help of adversarial learning, GANE achieves a similar performance to SDNE without deep architecture. DVNE outperforms all the vector-based embedding models, which proves the advantage of distribution-based embedding. As for the three specifications of our framework, all of them outperform the state-of-the-art models with a noticeable margin on each dataset. For example, RAWEN outperforms the best baseline DVNE by 5% in precision on Google. We also evaluate each method above on node classification tasks. Fig. 3 plots the results of RAWEN-G because three kinds of distributions show similar performance.

For RQ2, we empirically validate the robustness of RAWEN in Fig. 5. For the sake of clarity we omit the results of GF and Node2vec, which show a similar performance to LINE. For baseline models, only GANE shows slight robustness against noises, probably due to the adversarial learning. As a comparison, stronger robustness effect is observed on RAWEN after we artificially inject noise into the ground-truth graph with diverse noise levels. For example, on wiki-vote dataset with noise level 6 (the largest), our framework only degrades by 2% decrease in AUC, while other baseline models averagely degrade by 10%. In other words, RAWEN indeed has the ability to learn robust representations.

For RQ3, RAWEN-G achieves the best performance over the other two specifications. Meanwhile, we visualize the node embeddings obtained from our model in Fig. 4 using t-SNE algorithm (Maaten and Hinton 2008). Interestingly, the embeddings from RAWEN-D successfully capture the structural information of the given graph, as in Fig. 4. It is mainly because, the distribution-based embedding in the Dirichlet case can be regarded as the parameters of certain multinomial distribution. Therefore, $q(z|v)$ is expected to capture the clustering information in the graph.

Related Work

Node embedding is a crucial task in graph analysis, where traditional methods viewed the node embedding problem as how to represent nodes as real vectors (Hamilton, Ying, and Leskovec 2017), e.g., by directly embedding nodes with vectors (Perozzi, Al-Rfou, and Skiena 2014; Tang et al. 2015; Grover and Leskovec 2016) and calculating vectors by linkage relation (Wang, Cui, and Zhu 2016; Kipf and Welling 2016a). Recently, some methods were proposed to embed nodes with distributions instead of vectors and empirical results proved the effectiveness of the embeddings (He et al. 2015; Bojchevski and Günnemann 2018; Zhu et al. 2018). However, to the best of our knowledge, only Gaussian distribution has been studied as embedding space previously.

Robustness is a common concern in node embedding, which can be defined as the sensitiveness of embeddings to external noises on the graph (Ribeiro, Saverese, and Figueiredo 2017; Bojcheski and Günnemann 2018; Dai et al. 2018a). Efforts have been made on vector-based embedding models, by e.g. imposing prior on the vector space (Kipf and Welling 2016b). Adversarial learning on the graph (Wang et al. 2018) is also a practical way to improve the robustness of vector-based embeddings (Dai et al. 2018b; Hong, Li, and Wang 2018). However, there still lacks a robust solution for distribution-based embedding.

Conclusion

In this paper, we focus on distribution-based node embedding and propose a novel method for improving the robustness of embedded distributions, based on a derived generalization bound for Wasserstein embedding in a PAC-Bayesian framework (Theorem 1). Furthermore, in order to minimize the upper bound, we propose an algorithm called adversarial PAC-Bayesian learning. A neural network based instance of this algorithm is RAWEN, the robustness of which is further certificated by theoretical analysis. RAWEN is also the first attempt to extend embedding space of distributions to incorporate more kinds of distributions, including Dirichlet distribution and truncated Gaussian distribution. Empirical results prove the effectiveness and robustness of our proposed method. In the future, we would like to gen-

eralize the framework to more kinds of distributions such as Gamma distribution. We suggest it is also promising to explore the interpretations of the embedded results.

Acknowledgement

This work was supported in part by National Natural Science Foundation of China (61972099, U1636204, U1836213, 61602121, U1736208, 61602123, U1836210, 61972372), Natural Science Foundation of Shanghai (19ZR1404800), and National Program on Key Basic Research (NO. 2015CB358800). Min Yang is a member of Shanghai Institute of Intelligent Electronics & Systems, Shanghai Institute for Advanced Communication and Data Science, and Engineering Research Center of CyberSecurity Auditing and Monitoring, Ministry of Education, China.

Appendix A: Proof of Theorem 1

In the proof we will make use of the following lemma:

Lemma 1. *(Maurer 2004) Let $X \in [0,1]$ be a real-valued random variable with expectation μ. If we i.i.d sample $x_1, \cdots, x_n$ and $\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$, then for $n \geq 8$:*

$$\mathbb{E}_X[e^{nKL(\mu,\bar{x})}] \leq 2\sqrt{n} \tag{19}$$

Theorem 1. *Given a graph with N nodes, $\Omega = \{(v_i, v_j, y_{ij})\}$ is the observed linkage information sampled from P_Y with size $M \geq 8$. Suppose $\mathcal{Z} \subseteq \mathbb{R}^K$ is a metric space, and $\mathcal{Q}$ is a family of distribution defined on $\mathcal{Z}$. Each node v_i is related with a distribution $q(z|v_i) \in \mathcal{Q}$, where distance between embedded distributions d_{ij} is Wasserstein distance. Loss function $\ell(d_{ij}, y_{ij}) \in [0,1]$ is pre-defined. Then given optimal function D^* defined in the dual form of Wasserstein distance, prior distribution $p(z)$ defined on $\mathcal{Z}$ and $\delta \in (0,1)$, with probability at least $1-\delta$ we have for all $Q = \{q(z|v_1), \cdots, q(z|v_N)\} \subseteq \mathcal{Q}$:*

$$\ell(Q) \leq \hat{\ell}(Q,\Omega) + \frac{\sum_{i=1}^{N} KL(q(z|v_i)\|p(z)) + \ln M - \ln\delta}{M} \tag{20}$$

where $\hat{\ell}(Q,\Omega)$ is the empirical loss on Ω and the distance d_{ij} between node v_i and v_j is calculated as,

$$d_{ij} = \mathbb{E}_{z\sim q(z|v_i)}[D^*(z)] - \mathbb{E}_{z\sim q(z|v_j)}[D^*(z)] \tag{21}$$

Proof. According to the framework of PAC-Bayesian theory (McAllester 1999), we first define the hypothesis space $\mathcal{H} = \mathcal{Z} \times \cdots \times \mathcal{Z}$ in Wasserstein embedding, with a hypothesis is $h = (z_1, \cdots, z_n) \in \mathcal{H}$, where $z_i \in \mathcal{Z}$. Given optimal function D^*, we further define the loss function of $\ell(h, y_{ij})$ as,

$$\ell(h, y_{ij}) = \ell(D^*(z_i) - D^*(z_j), y_{ij}) \tag{22}$$

where $\ell(D^*(z_i) - D^*(z_j), y_{ij}) \in [0,1]$. Then we define a distribution Q on the hypothesis space $\mathcal{H}$ by $Q(h) = \prod_{i=1}^{N} q(z_i|v_i)$, where $q(z|v_i)$ is the embedded distribution as defined.

We also define a prior distribution on the hypothesis space by $P(h) = \prod_{i=1}^{N} p(z_i)$, where $p(z_i)$ is the prior distribution on $\mathbb{R}^K$ as defined. Then the loss function of a distribution Q can be written by,

$$\ell(Q, y_{ij}) = \int [\ell(D^*(z_i) - D^*(z_j), y_{ij})] q(z_i|v_i) q(z_j|v_j) dz_i dz_j$$

Given two Bernoulli variable a, b, the KL divergence between then can be written as,

$$KL(a\|b) = a\ln\frac{a}{b} + (1-a)\ln\frac{1-a}{1-b} \tag{23}$$

If we define $a = \mathbb{E}_{h\sim Q}\mathbb{E}_{y_{ij}}[\ell(h, y_{ij})]$ and $b = \mathbb{E}_{h\sim Q}\big[\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h, y_{ij})\big]$, then according to the convexity of KL divergence (Seeger 2003), we have:

$$KL(a\|b) \leq \mathbb{E}_{h\sim Q}\big[KL(\mathbb{E}_{y_{ij}}[\ell(h, y_{ij})]\|\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h, y_{ij}))\big]$$

According to Jensen inequality, for any prior P defined on $\mathcal{H}$, we have:

$$\begin{aligned}&\mathbb{E}_{h\sim Q}\big[M \cdot KL(\mathbb{E}_{y_{ij}}[\ell(h, y_{ij})]\|\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h, y_{ij}))\big] \\ &\leq KL(Q\|P) + \ln\mathbb{E}_{h\sim P}\big[e^{M\cdot KL(\mathbb{E}_{y_{ij}}[\ell(h,y_{ij})]\|\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h,y_{ij}))}\big]\end{aligned} \tag{24}$$

When $M \geq 8$, the term has the following bound by Lemma 1:

$$\mathbb{E}_{h\sim P}\big[e^{M\cdot KL(\mathbb{E}_{y_{ij}}[\ell(h,y_{ij})]\|\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h,y_{ij}))}\big] \leq 2\sqrt{M}$$

Therefore by Markov inequality, with probability $1-\delta$:

$$\mathbb{E}_{h\sim P}\big[e^{M\cdot KL(\mathbb{E}_{y_{ij}}[\ell(h,y_{ij})]\|\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h,y_{ij}))}\big] \leq \frac{2\sqrt{M}}{\delta}$$

Together with Eq. 24, we obtain:

$$\begin{aligned}&KL\Big(\mathbb{E}_{h\sim Q}\mathbb{E}_{y_{ij}}[\ell(h, y_{ij})]\|\mathbb{E}_{h\sim Q}\big[\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h, y_{ij})\big]\Big) \\ &\leq \frac{KL(Q\|P) + \ln 2\sqrt{M} - \ln\delta}{M}\end{aligned}$$

with probability $1-\delta$. Note that $KL(a,b) \geq 2(a-b)^2$, we obtain,

$$\begin{aligned}\mathbb{E}_{h\sim Q}\mathbb{E}_{y_{ij}}[\ell(h, y_{ij})] \leq\ &\mathbb{E}_{h\sim Q}\big[\frac{1}{M}\sum_{y_{ij}\in\Omega}\ell(h, y_{ij})\big] \\ &+ \sqrt{\frac{KL(Q\|P) + \ln 2\sqrt{M} - \ln\delta}{M}}\end{aligned} \tag{25}$$

By taking the expectation on $KL(Q\|P)$:

$$\begin{aligned}KL(Q\|P) &= \int \big[\prod_{i=j}^{N} q(z_j|v_j)\big] \ln\big[\prod_{i=1}^{N}\frac{q(z_i|v_i)}{p(z_i)}\big] dz_1 \cdots dz_N \\ &= \sum_{i=1}^{N}\int \big[\prod_{j=1}^{N} q(z_j|v_j)\big] \ln\frac{q(z_i|v_i)}{p(z_i)} dz_1 \cdots dz_N \\ &= \sum_{i=1}^{N}\int q(z|v_i)\ln\frac{q(z|v_i)}{p(z)} \\ &= \sum_{i=1}^{N} KL(q(z|v_i)\|p(z))\end{aligned} \tag{26}$$

and the following inequality (Seldin and Tishby 2010),

$$\sqrt{\frac{\sum_{i=1}^{N} KL(q(z|v_i)\|p(z)) + \ln 2\sqrt{M} - \ln\delta}{2M}} \leq \frac{\sum_{i=1}^{N} KL(q(z|v_i)\|p(z)) + \ln(M+1) - \ln\delta}{M} \quad (27)$$

we finish the proof. □

Appendix B: Proof of Theorem 2 and 3

Theorem 2. *Under Assumption 1, for an arbitrary pair of nodes v_i, v_j s.t. $y_{ij} = 1$, if $d_{ij}(\theta^*) \leq \hat{\delta}_1$, then the expectation of $d_{ij}(\tilde{\theta}^*)$ satisfies*

$$\mathbb{E}_{\tilde{P}_Y}[d_{i,j}(\tilde{\theta}^*)] \leq \mathbb{E}_{P_Y}[\Delta] \cdot \left[\frac{R}{2\rho_M M}\sqrt{2J} + (1 - \frac{R}{2\rho_M M}) \cdot \tilde{\delta}_1\right] \quad (28)$$

where $\mathbb{E}_{P_Y}[\Delta]$ is the expectation of the length of path in P_Y and $J = \mathbb{E}_{p(z)}[\|z\|^2] < \infty$.

Proof. First, if the KL term in the upper bound is minimized, we obtain,

$$\begin{aligned} &\mathbb{E}_{\tilde{P}_Y}\left[\inf_{\gamma \in \prod(q(z|v_i), q(z|v_j))} \mathbb{E}_{(z_i,z_j)\sim\gamma}[\|z_i - z_j\|^2]\right] \\ &\leq \mathbb{E}_{\tilde{P}_Y}\mathbb{E}_{z_i\sim q(z|v_i)}\mathbb{E}_{z_j\sim q(z|v_j)}[\|z_i - z_j\|^2] \\ &\leq \mathbb{E}_{p(z)}[\|Z\|^2 + \|Z\|^2] \end{aligned} \quad (29)$$

Therefore we obtain,

$$\mathbb{E}_{\tilde{P}_Y}[d_{ij}(\tilde{\theta}^*)] \leq \sqrt{2J} \quad (30)$$

According to Assumption 1, there exists a path w.r.t Ω from v_i to v_j. Then for each $y_{uv} = 1$ in the path, $\tilde{y}_{ij} = 0$ with probability $\frac{R}{2\rho_M M}$. If y_{uv} is not influenced by the noise, we have $d_{ij}(\tilde{\theta}^*) \leq \tilde{\delta}_1$. Otherwise, we have $\mathbb{E}_{\tilde{P}_Y}[d_{ij}(\tilde{\theta}^*)] \leq \sqrt{2J}$.

For each y_{uv} in the path, whether it is influenced by noise or not can be regarded as an event respecting Binomial distribution with parameter $\frac{R}{2\rho_M M}$. Then with the property of metric space, we sum $d_{uv}(\tilde{\theta}^*)$ in the path and obtain the following inequality

$$\mathbb{E}_{\tilde{P}_Y}\left[d_{ij}(\theta^*)\right] \leq \sum_{u,v} \mathbb{E}_{\tilde{P}_Y}\left[d_{uv}(\tilde{\theta}^*)\right] \quad (31)$$

For each $\mathbb{E}_{\tilde{P}_Y}\left[d_{uv}(\tilde{\theta}^*)\right]$ we have:

$$\mathbb{E}_{\tilde{P}_Y}\left[d_{uv}(\tilde{\theta}^*)\right] \leq \frac{R}{2\rho_M M} \cdot \sqrt{2J} + (1 - \frac{R}{2\rho_M M}) \cdot \tilde{\delta}_1 \quad (32)$$

□

Theorem 3. *Under Assumption 1, for an arbitrary pair of nodes v_i, v_j s.t. $y_{ij} = 0$, if $d_{ij}(\theta^*) \geq \hat{\delta}_0$, then the expectation of $d_{ij}(\tilde{\theta}^*)$ satisfies*

$$\mathbb{E}_{\tilde{P}_Y}[d_{ij}(\tilde{\theta}^*)] \geq \hat{\delta}_0 \cdot (1 - \frac{R}{2(1-\rho_M)M}) \quad (33)$$

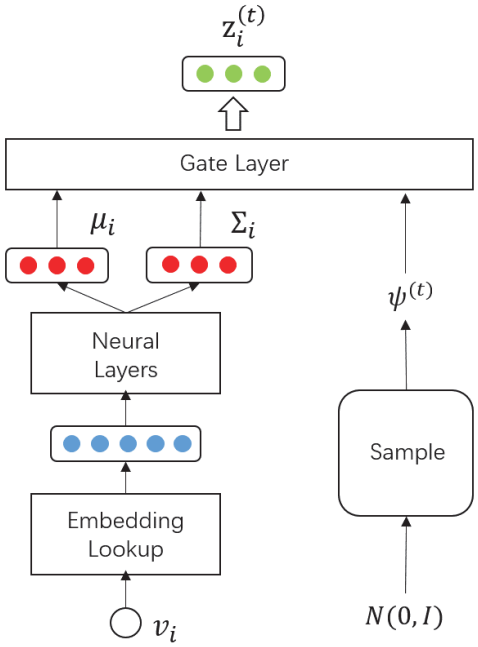


Figure 6: Structure of the generator

Proof. It is straightforward to see that for v_i and v_j, there is assumed to have no path w.r.t Ω. Otherwise, we have $d_{ij}(\theta^*) \leq \sum_{u,v} d_{uv}(\theta^*) \leq \Delta\delta_1 \ll \hat{\delta}_0$, which brings a contradiction.

Then as a worst case analysis, there will be a path between v_i, v_j with probability $\frac{R}{2(1-\rho_M)M}$ influenced by the noise while there is still no path between v_i and v_j with probability $1 - \frac{R}{2(1-\rho_M)M}$. In other words, distance between v_i, v_j is not influenced by the noise. Thus we have $d_{ij}(\tilde{\theta}^*) \geq \hat{\delta}_0$ and by a direct calculation of the expectation of Bernoulli distribution,

$$\mathbb{E}_{\tilde{P}_Y}[d_{ij}(\tilde{\theta}^*)] \geq 0 \cdot \frac{R}{2(1-\rho_M)M} + \hat{\delta}_0 \cdot (1 - \frac{R}{2(1-\rho_N)N})$$

□

Appendix C: Detailed Implementation

For the implementation of Q_θ, D_ϕ and H_γ, discriminator H_γ and D_ϕ are one-layer full connection layer. We present details on implementation of function Q_θ and the corresponding sampling process in Fig. 6. The choices of gate layer, as we have discussed above, are listed as follows.

- If $q(z|v)$ is Multivariate Gaussian, the gate function is identity function.
- If $q(z|v)$ is Dirichlet distribution, the gate function is softmax.
- If $q(z|v)$ is Multivariate Truncated Normal distribution, the gate function is softplus, where the μ_i is also an output from a softplus function.

References

Ahmed, A.; Shervashidze, N.; Narayanamurthy, S.; Josifovski, V.; and Smola, A. J. 2013. Distributed large-scale natural graph factorization. In *WWW*.

Bojcheski, A., and Günnemann, S. 2018. Adversarial attacks on node embeddings. *arXiv preprint arXiv:1809.01093*.

Bojchevski, A., and Günnemann, S. 2018. Deep gaussian embedding of attributed graphs: Unsupervised inductive learning via ranking. *ICLR*.

Brouwer, T., and Lio, P. 2017. Prior and likelihood choices for bayesian matrix factorisation on small datasets. *arXiv preprint arXiv:1712.00288*.

Cao, S.; Lu, W.; and Xu, Q. 2015. Grarep: Learning graph representations with global structural information. In *CIKM*. ACM.

Chamberlain, B. P.; Clough, J.; and Deisenroth, M. P. 2017. Neural embeddings of graphs in hyperbolic space. *arXiv preprint arXiv:1705.10359*.

Dai, H.; Li, H.; Tian, T.; Huang, X.; Wang, L.; Zhu, J.; and Song, L. 2018a. Adversarial attack on graph structured data. In *ICML*, 1123–1132.

Dai, Q.; Li, Q.; Tang, J.; and Wang, D. 2018b. Adversarial network embedding. In *AAAI*.

Goodfellow, I. J.; Pougetabadie, J.; Mirza, M.; Xu, B.; Wardefarley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2014. Generative adversarial networks. *NIPS*.

Goyal, P., and Ferrara, E. 2018. Graph embedding techniques, applications, and performance: A survey. *Knowledge-Based Systems* 151:78–94.

Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *SIGKDD*.

Hamilton, W. L.; Ying, R.; and Leskovec, J. 2017. Representation learning on graphs: Methods and applications. *IEEE Data Engineering Bulletin*.

He, S.; Liu, K.; Ji, G.; and Zhao, J. 2015. Learning to represent knowledge graphs with gaussian embedding. In *CIKM*.

Hong, H.; Li, X.; and Wang, M. 2018. Gane: A generative adversarial network embedding.

Jeh, G., and Widom, J. 2002. Simrank: a measure of structural-context similarity. In *SIGKDD*.

Katz, L. 1953. A new status index derived from sociometric analysis. *Psychometrika* 18(1):39–43.

Kingma, D., and Ba, J. 2015. Adam: A method for stochastic optimization. *ICLR*.

Kingma, D. P., and Welling, M. 2013. Auto-encoding variational bayes. *arXiv preprint arXiv:1312.6114*.

Kipf, T. N., and Welling, M. 2016a. Semi-supervised classification with graph convolutional networks. *ICLR*.

Kipf, T. N., and Welling, M. 2016b. Variational graph auto-encoders. *NIPS Workshop on Bayesian Deep Learning*.

Lafferty, J. D., and Blei, D. M. 2006. Correlated topic models. In *NIPS*.

Maaten, L. v. d., and Hinton, G. 2008. Visualizing data using t-sne. *JMLR*.

Maurer, A. 2004. A note on the pac bayesian theorem. *arXiv preprint cs/0411099*.

McAllester, D. A. 1999. Pac-bayesian model averaging. In *COLT*, volume 99, 164–170. Citeseer.

Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: Online learning of social representations. In *SIGKDD*.

Ribeiro, L. F.; Saverese, P. H.; and Figueiredo, D. R. 2017. struc2vec: Learning node representations from structural identity. In *SIGKDD*.

Seeger, M. 2003. Bayesian gaussian process models: Pac-bayesian generalisation error bounds and sparse approximations. Technical report, University of Edinburgh.

Seldin, Y., and Tishby, N. 2010. A pac-bayesian approach to unsupervised learning with application to co-clustering analysis. *JMLR*.

Tang, J.; Qu, M.; Wang, M.; Zhang, M.; Yan, J.; and Mei, Q. 2015. Line: Large-scale information network embedding. In *WWW'05*, 1067–1077.

Tsitsulin, A.; Mottin, D.; Karras, P.; and Müller, E. 2018. Verse: Versatile graph embeddings from similarity measures. In *WWW*.

Villani, C. 2003. *Topics in optimal transportation*. American Mathematical Soc.

Wang, H.; Wang, J.; Wang, J.; Zhao, M.; Zhang, W.; Zhang, F.; Xie, X.; and Guo, M. 2018. Graphgan: graph representation learning with generative adversarial nets. In *AAAI*.

Wang, D.; Cui, P.; and Zhu, W. 2016. Structural deep network embedding. In *SIGKDD'16*.

Xie, J.; Kelley, S.; and Szymanski, B. K. 2013. Overlapping community detection in networks: The state-of-the-art and comparative study. *Acm computing surveys (csur)* 45(4):43.

Zhu, D.; Cui, P.; Wang, D.; and Zhu, W. 2018. Deep variational network embedding in wasserstein space. In *SIGKDD*.

Zügner, D.; Akbarnejad, A.; and Günnemann, S. 2018. Adversarial attacks on neural networks for graph data. In *SIGKDD*.