IEICE TRANS. INF. & SYST., VOL.E105–D, NO.10 OCTOBER 2022

LETTER

Graph Embedding with Outlier-Robust Ratio Estimation

Kaito SATTA[†], ***Nonmember and*** **Hiroaki SASAKI**[†a)], ***Member***

SUMMARY The purpose of graph embedding is to learn a lower-dimensional embedding function for graph data. Existing methods usually rely on maximum likelihood estimation (MLE), and often learn an embedding function through conditional mean estimation (CME). However, MLE is well-known to be vulnerable to the contamination of outliers. Furthermore, CME might restrict the applicability of the graph embedding methods to a limited range of graph data. To cope with these problems, this paper proposes a novel method for graph embedding called the *robust ratio graph embedding* (RRGE). RRGE is based on the ratio estimation between the conditional and marginal probability distributions of link weights given data vectors, and would be applicable to a wider-range of graph data than CME-based methods. Moreover, to achieve outlier-robust estimation, the ratio is estimated with the γ-cross entropy, which is a robust alternative to the standard cross entropy. Numerical experiments on artificial data show that RRGE is robust against outliers and performs well even when CME-based methods do not work at all. Finally, the performance of the proposed method is demonstrated on realworld datasets using neural networks.

key words: *graph embedding, representation learning, graph data, outlier-robustness, ratio estimation*

1. Introduction

Graph embedding is aimed at learning an embedding function from data vectors associated with nodes and their link weights (i.e., graph data). The learned embedding function converts data vectors to useful lower-dimensional feature vectors, which enable us to easily use a variety of statistical methods, while it is not straightforward to apply them to graph data without learning embedding functions. A number of tasks can be performed via graph embedding such as node classification, link prediction, node clustering, etc. For references of these tasks and more examples, we refer to a recent survey paper for graph embedding [1]

To develop scalable methods to the graph size or dimensionality of data vectors, recent works for graph embedding have employed neural networks and stochastic optimization [2]–[5]. A common approach is based on probabilistic models. In [6], the conditional distribution of link weights given data vectors is modeled by the Poisson distribution, and then an embedding function is learned by maximizing the likelihood function with stochastic optimization. The likelihood-based approach is simple and has been demonstrated to work well, but includes two limitations: First, we need to specify a probabilistic model for the conditional distribution. Thus, a misspecification of the conditional distribution would lead to degenerated graph embedding. The second limitation is that the maximum likelihood estimation (MLE) is known to be sensitive to outliers. Thus, when link weights are contaminated by outliers, the likelihood-based approach can be inappropriate.

To overcome these limitations, a robust method for graph embedding is proposed in [7]. In this method, an embedding function is learned through conditional mean estimation (CME) of link weights given data vectors using the β-cross entropy [8]. β-cross entropy is known as a density power cross entropy and often produces more robust estimation against outliers than MLE. In addition, any specific probabilistic model is not used for CME in this method. However, the conditional mean (CM) only describes a limited statistical dependency between link weights and data vectors, and thus there should be still a room for improvement over CME. For instance, when CM is constant, then the embedding function learned through CME can be almost a constant function, which is useless in general.

This paper proposes a novel method for graph embedding called the *robust ratio graph embedding* (RRGE). RRGE robustly learns an embedding function through estimation of the ratio between the conditional and marginal distributions of link weights given data vectors. This approach should be more appropriate than CME because even when CM is constant, the conditional distribution in the ratio is not necessarily constant and captures more general statistical dependencies between link weights and data vectors. Thus, the proposed method would be able to learn a useful embedding function on a wider-range of graph data. Furthermore, for robust estimation of the ratio, we propose to use the γ-cross entropy [9], which has a favorable robustness property so called the *strong robustness*: Contamination ratio of outliers is not necessarily assumed to be small for robust estimation. Previously, a similar method has been proposed for representation learning on Euclidean data based on density ratio estimation with the γ-cross entropy [10]. Our work can be regarded as an application of the Euclidean method for representation learning to graph embedding. Finally, we numerically demonstrate that RRGE is robust to outliers and works well on a wide-range of graph data even when CM is constant.

Manuscript received April 17, 2022.
Manuscript revised June 17, 2022.
Manuscript publicized July 4, 2022.

[†]The authors are with Future University Hakodate, Hakodate-shi, 041–8655 Japan.
a) E-mail: hsasaki@fun.ac.jp

DOI: 10.1587/transinf.2022EDL8033

2. Problem Setting and Background

We consider an undirected graph which consists of n nodes and links between them. The i-th node is equipped with a d-dimensional *data vector* $\boldsymbol{x}_i \in \mathbb{R}^d$, and the *link weight* between i-th and j-th nodes is expressed as a nonnegative and symmetric weight $w_{ij} = w_{ji} \in \{0, 1, 2, \dots, \}$ where $w_{ii} = 0$ for all $i = 1, \dots, n$. The goal of *graph embedding* is to estimate an *embedding* function $\boldsymbol{f}(\boldsymbol{x}) : \mathbb{R}^d \to \mathbb{R}^K$ from $\{w_{ij}\}_{1 \le i < j \le n}$ and $\{\boldsymbol{x}_i\}_{i=1}^n$ where $K > 0$ denotes the feature dimension.

In order to take the link information into account, the fundamental task is to estimate $\boldsymbol{f}(\boldsymbol{x})$ such that the statistical dependency between the link weights w_{ij} and data vectors $(\boldsymbol{x}_i, \boldsymbol{x}_j)$ is captured. To this end, let us denote the conditional distribution of the link weight w given data vectors $(\boldsymbol{x}, \boldsymbol{x}')$ by $p(w|\boldsymbol{x}, \boldsymbol{x}')$. Then, we assume that the data vectors $\boldsymbol{x}_i$ are independently and identically distributed, and the link weights w_{ij} in a random undirected graph are independently generated as $w_{ij}|\boldsymbol{x}_i, \boldsymbol{x}_j \overset{\text{indep.}}{\sim} p(w|\boldsymbol{x}_i, \boldsymbol{x}_j)$.

Maximum likelihood estimation (MLE) is one of the simplest approach to learn $\boldsymbol{f}(\boldsymbol{x})$ [2], [3], [6]. For instance, [6] substitutes a conditional Poisson distribution for $p(w|\boldsymbol{x}_i, \boldsymbol{x}_j)$, and formulates the log-likelihood function as

$$\sum_{(i,j)\in\mathcal{I}_n} \left[w_{ij} \log \mu_{\boldsymbol{f},\alpha}(\boldsymbol{x}_i, \boldsymbol{x}_j) - \mu_{\boldsymbol{f},\alpha}(\boldsymbol{x}_i, \boldsymbol{x}_j) \right], \tag{1}$$

where $\mathcal{I}_n := \{(i, j) | 1 \le i < j \le n\}$, and $\mu_{\boldsymbol{f},\alpha}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ denotes the conditional mean in the conditional Poisson distribution and is modeled as $\log \mu_{\boldsymbol{f},\alpha}(\boldsymbol{x}_i, \boldsymbol{x}_j) := \langle \boldsymbol{f}(\boldsymbol{x}_i), \boldsymbol{f}(\boldsymbol{x}_j) \rangle - \alpha$ with the inner product $\langle \cdot, \cdot \rangle$ and a parameter α. Then, the embedding function $\boldsymbol{f}(\boldsymbol{x})$ is learned by maximizing (1). This approach is simple and appealing, but includes two issues: First, it assumes that the underlying conditional distribution is Poisson, which may not be fulfilled in practice. Second, MLE is often hampered by the contamination of outliers.

To alleviate these issues, [7] uses the β-cross entropy [8] with which an embedding function $\boldsymbol{f}(\boldsymbol{x})$ is learned through conditional mean estimation (CME). By modeling the conditional mean $E[W|\boldsymbol{x}_i, \boldsymbol{x}_j] := \sum_w w p(w|\boldsymbol{x}_i, \boldsymbol{x}_j)$ by $\mu_{\boldsymbol{f},\alpha}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ as in (1), the empirical β-cross entropy for CME is formulated as

$$\sum_{(i,j)\in\mathcal{I}_n} \left[-w_{ij} \frac{\mu_{\boldsymbol{f},\alpha}(\boldsymbol{x}_i, \boldsymbol{x}_j)^\beta - 1}{\beta} + \frac{\mu_{\boldsymbol{f},\alpha}(\boldsymbol{x}_i, \boldsymbol{x}_j)^{1+\beta}}{1+\beta} \right]. \tag{2}$$

This graph embedding method based on the β-cross entropy has been shown to be robust against outliers. In addition, this model does not employ a particular probabilistic model for CME. However, the conditional mean $E[W|\boldsymbol{x}_i, \boldsymbol{x}_j]$ only captures a limited dependency between the link weight w_{ij} and data vectors $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. For instance, when the true conditional mean is constant (i.e., $E[W|\boldsymbol{x}_i, \boldsymbol{x}_j] = \text{const}$), then the estimated conditional mean $\mu_{\widehat{\boldsymbol{f}},\widehat{\alpha}}(\boldsymbol{x}_i, \boldsymbol{x}_j)$ could be close to constant (i.e., $\mu_{\widehat{\boldsymbol{f}},\widehat{\alpha}}(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \widehat{\boldsymbol{f}}(\boldsymbol{x}_i), \widehat{\boldsymbol{f}}(\boldsymbol{x}_j) \rangle - \widehat{\alpha} \approx \text{const}$), implying that the learned embedding function $\widehat{\boldsymbol{f}}(\boldsymbol{x})$ can be almost a useless constant function.

3. Graph Embedding with Robust Ratio Estimation

3.1 Ratio Estimation for Graph Embedding

A more general approach over CME should be directly based on estimation of the conditional distribution $p(w|\boldsymbol{x}, \boldsymbol{x}')$ without specifying any probabilistic models. However, as already discussed in Sect. 3.2 of [7] with the β-cross entropy, this general approach might be intractable because it involves an infinite summation with respect to $w \in \{0, 1, 2, \dots\}$, which is essentially the same as the partition function problem in MLE. Alternatively, we estimate the following ratio of the conditional and marginal distributions for graph embedding:

$$\log \frac{p(w|\boldsymbol{x}, \boldsymbol{x}')}{p(w)} = \log \frac{p(w, \boldsymbol{x}, \boldsymbol{x}')}{p(w)p(\boldsymbol{x}, \boldsymbol{x}')}, \tag{3}$$

where $p(\boldsymbol{x}, \boldsymbol{x}')$ denotes the joint density for $\boldsymbol{x}$ and $\boldsymbol{x}'$ and $p(w)$ is the marginal distribution of the link weight w. Unlike the conditional expectation $E[W|\boldsymbol{x}_i, \boldsymbol{x}_j]$, the distribution ratio (3) includes the conditional distribution more directly and thus would capture more general statistical dependency between w and $(\boldsymbol{x}, \boldsymbol{x}')$. Thus, even when $E[W|\boldsymbol{x}_i, \boldsymbol{x}_j]$ is constant, the distribution ratio (3) is not necessarily constant and must be dependent to $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ due to the conditional distribution in the numerator. Moreover, by slightly modifying Theorem 1 in [10], this ratio estimation can be seen as maximizing mutual information between link weighs and feature vectors $\boldsymbol{f}(\boldsymbol{x})$ under some conditions, ensuring the embedding function learned through the ratio estimation would include large amount of information to link weights.

3.2 Outlier-Robust Ratio Estimation

Next, we develop a practical robust method such that the intractable infinite summation does not appear. To this end, we apply a representation learning method for Euclidean data proposed in [10] to graph data, which begins with the following binary classification problem:

$$\begin{aligned} &\mathcal{D}_+ := \{(w_{ij}, \boldsymbol{x}_i, \boldsymbol{x}_j)\}_{i,j=1}^n \sim p(w, \boldsymbol{x}, \boldsymbol{x}') \\ \text{vs.} \quad &\mathcal{D}_- := \{(w^*_{ij}, \boldsymbol{x}_i, \boldsymbol{x}_j)\}_{i,j=1}^n \sim p(w)p(\boldsymbol{x}, \boldsymbol{x}'), \end{aligned}$$

where w^*_{ij} is a random permutation of w_{ij} with respect to i, j and can be regarded as samples from the marginal distribution $p(w)$ because the random permutation deletes the dependency to $(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Then, we assign *pseudo* labels $y = 0$ and $y = 1$ to $\mathcal{D}_+$ and $\mathcal{D}_-$ respectively, and employ a γ-cross entropy [9] to estimate the posterior probability $p(y|w, \boldsymbol{x}, \boldsymbol{x}')$. To this end, we express a model for $p(y|w, \boldsymbol{x}, \boldsymbol{x}')$ as

$$f(y|w, \boldsymbol{x}, \boldsymbol{x}') = \{F(r(w, \boldsymbol{x}, \boldsymbol{x}'))\}^{1-y} \{1 - F(r(w, \boldsymbol{x}, \boldsymbol{x}'))\}^y, \tag{4}$$

where $F(r(w,\boldsymbol{x},\boldsymbol{x}')) := \frac{e^{r(w,\boldsymbol{x},\boldsymbol{x}')}}{1+e^{r(w,\boldsymbol{x},\boldsymbol{x}')}}$. Since it holds from (4) that

$$r(w,\boldsymbol{x},\boldsymbol{x}') = \log \frac{f(y=0|w,\boldsymbol{x},\boldsymbol{x}')}{f(y=1|w,\boldsymbol{x},\boldsymbol{x}')},$$

$r(w,\boldsymbol{x},\boldsymbol{x}')$ can be seen as a model for

$$\begin{aligned}\log \frac{p(y=0|w,\boldsymbol{x},\boldsymbol{x}')}{p(y=1|w,\boldsymbol{x},\boldsymbol{x}')} &= \log \frac{p(w,\boldsymbol{x},\boldsymbol{x}'|y=0)p(y=0)}{p(w,\boldsymbol{x},\boldsymbol{x}'|y=1)p(y=1)} \\ &= \log \frac{p(w,\boldsymbol{x},\boldsymbol{x}')}{p(w)p(\boldsymbol{x},\boldsymbol{x}')},\end{aligned}$$

where we applied the Bayes theorem with the symmetric class probabilities (i.e., $p(y=0) = p(y=1) = \frac{1}{2}$) and following relation from $\mathcal{D}_+$ and $\mathcal{D}_-$:

$$\begin{aligned}p(w,\boldsymbol{x},\boldsymbol{x}'|y=0) &= p(w,\boldsymbol{x},\boldsymbol{x}') \\ p(w,\boldsymbol{x},\boldsymbol{x}'|y=1) &= p(w)p(\boldsymbol{x},\boldsymbol{x}'). \end{aligned} \tag{5}$$

Thus, we can obtain an estimate of the distribution ratio (3) through the posterior probability estimation. In fact, the same approach has been taken in density ratio estimation using logistic regression [11].

To fit the model $r(w,\boldsymbol{x},\boldsymbol{x}')$, our objective function $D_\gamma(r)$ is formulated from the γ-cross entropy for posterior probability estimation in [9], [12] as follows:

$$\begin{aligned}D_\gamma(r) &:= -\frac{1}{\gamma}\log\left[\sum_w \iint \sum_{y=0}^{1} p(y,w,\boldsymbol{x},\boldsymbol{x}')\right. \\ &\quad \left.\times\left(\frac{f(y|w,\boldsymbol{x},\boldsymbol{x}')^{\gamma+1}}{\sum_{y'=0}^{1} f(y'|w,\boldsymbol{x},\boldsymbol{x}')^{\gamma+1}}\right)^{\frac{\gamma}{\gamma+1}} \mathrm{d}\boldsymbol{x}\mathrm{d}\boldsymbol{x}'\right] \\ &:= -\frac{1}{\gamma}\log\left[\frac{1}{2}E_{\mathrm{WXX}'}\left[\left\{F_{\gamma+1}(r(W,\boldsymbol{X},\boldsymbol{X}'))\right\}^{\frac{\gamma}{\gamma+1}}\right]\right. \\ &\quad \left.+\frac{1}{2}E_{\mathrm{W}\times\mathrm{XX}'}\left[\left\{1-F_{\gamma+1}(r(W,\boldsymbol{X},\boldsymbol{X}'))\right\}^{\frac{\gamma}{\gamma+1}}\right]\right],\end{aligned} \tag{6}$$

where $E_{\mathrm{WXX}'}$ and $E_{\mathrm{W}\times\mathrm{XX}'}$ denote the expectations over $p(w,\boldsymbol{x},\boldsymbol{x}')$ and $p(w)p(\boldsymbol{x},\boldsymbol{x}')$ respectively, $F_{\gamma+1}(r(w,\boldsymbol{x},\boldsymbol{x}')) := \frac{e^{(\gamma+1)r(w,\boldsymbol{x},\boldsymbol{x}')}}{1+e^{(\gamma+1)r(w,\boldsymbol{x},\boldsymbol{x}')}}$, and we assumed $p(y=0) = p(y=1) = \frac{1}{2}$ again and used (5). In (6), γ is a positive parameter and controls the robustness to outliers: A higher value of γ implies more robust to outliers. A simple calculation ensures that $D_\gamma(r)$ is minimized at $r(w,\boldsymbol{x},\boldsymbol{x}') = \log\frac{p(w,\boldsymbol{x},\boldsymbol{x}')}{p(w)p(\boldsymbol{x},\boldsymbol{x}')}$.

In practice, we need to approximate (6) from data samples in $\mathcal{D}_+$ and $\mathcal{D}_-$. By applying the law of large numbers for doubly-indexed partially dependent random variables [7, Theorem A.1] , an empirical approximation of (6) is given up to a constant by

$$\begin{aligned}\widehat{D}_\gamma(r) &:= -\frac{1}{\gamma}\log\left[\frac{1}{|\mathcal{I}_n|}\sum_{(i,j)\in\mathcal{I}_n}\left\{F_{\gamma+1}(r(w_{ij},\boldsymbol{x}_i,\boldsymbol{x}_j))\right\}^{\frac{\gamma}{\gamma+1}}\right. \\ &\quad \left.+\frac{1}{|\mathcal{I}_n|}\sum_{(i,j)\in\mathcal{I}_n}\left\{1-F_{\gamma+1}(r(w^*_{ij},\boldsymbol{x}_i,\boldsymbol{x}_j))\right\}^{\frac{\gamma}{\gamma+1}}\right],\end{aligned} \tag{7}$$

where $|\mathcal{I}_n|$ denotes the number of elements in $\mathcal{I}_n$. The key point is that $\widehat{D}_\gamma(r)$ does not include any infinite summation with respect to w and thus easy to use in practice. We call the proposed method based on $\widehat{D}_\gamma(r)$ as the *robust ratio graph embedding* (RRGE). Theoretical analysis to the outlier-robustness of using the γ-cross entropy has been thoroughly performed in the context of density ratio estimation. We refer to Sect. 4.3 in [10].

4. Numerical Illustration

4.1 Illustration on Artificial Data

By following the experimental setting in [7], we first independently generated data vectors $\{\boldsymbol{x}_i \in \mathbb{R}^{20}\}_{i=1}^{200}$ from the mixture of four Gaussians: 50 data vectors $\boldsymbol{x}_i$ were generated from each Gaussian density, and the k-th Gaussian density for $k = 1,\ldots,4$ has mean $\boldsymbol{A}\boldsymbol{\mu}_k$ and the identity covariance matrix where the elements in $\boldsymbol{\mu}_k \in \mathbb{R}^5$ and $\boldsymbol{A} \in \mathbb{R}^{20\times 5}$ were independently sampled from the normal density. Then, every generated data vector $\boldsymbol{x}_i$ was rescaled such that $\sum_{i=1}^{200}\boldsymbol{x}_i/200 = 4$. Finally, the link weights w_{ij} were generated as follows:

Bernoulli link with outliers: We randomly generated w_{ij} from a Bernoulli distribution $B(p)$ where p denotes the probability: If $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are generated from the same Gaussian, then $w_{ij} \sim B(0.05)$, while $w_{ij} \sim B(q)$ if $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$ are generated from different Gaussians.
Categorical link with a constant mean: To generate graph data such that the conditional mean $E[W|\boldsymbol{x}_i,\boldsymbol{x}_j]$ is a constant, we first randomly generated $w_{ij} \in \{0,1,2\}$ from a categorical distribution as follows: Denoting by p_k the probability that $w_{ij} = k$ $(k = 0,1,2)$, $p_1 = 1 - 0.1/(1 + \|\boldsymbol{x}_i - \boldsymbol{x}_j\|/\delta)$ where $\delta := \max_{i,j}\|\boldsymbol{x}_i - \boldsymbol{x}_j\|$, while $p_0 = p_2 = (1-p_1)/2$. Thus, the conditional mean $E[W|\boldsymbol{x}_i,\boldsymbol{x}_j]$ is exactly one, while the conditional variance of w_{ij} depends on $\boldsymbol{x}_i$ and $\boldsymbol{x}_j$.

By expressing the embedding function by $\boldsymbol{f}(\boldsymbol{x}) = \boldsymbol{B}\boldsymbol{x}$ where $\boldsymbol{B} \in \mathbb{R}^{5\times 20}$ is a matrix, we applied the following three graph embedding methods to the generated graph data:

ML-GE: Embedding function $\boldsymbol{f}(\boldsymbol{x})$ is learned by maximizing the likelihood function (1) w.r.t. $\boldsymbol{B}$ and α.
β-GE: Embedding function $\boldsymbol{f}(\boldsymbol{x})$ is learned by minimizing the β-cross entropy (2) w.r.t $\boldsymbol{B}$ and α.
RRGE: With a parameter w_0, we express $r(w,\boldsymbol{x},\boldsymbol{x}') := |w - w_0|\langle\boldsymbol{f}(\boldsymbol{x}),\boldsymbol{f}(\boldsymbol{x}')\rangle - \alpha$ in (7), and the embedding function $\boldsymbol{f}(\boldsymbol{x})$ as well as the parameters (α, w_0) are learned by minimizing the γ-cross entropy (7).

The objective functions in all methods were optimized by using BFGS, and the ridge regularization was performed to $\boldsymbol{B}$. Due to BFGS, the absolute function $|w - w_0|$ in RRGE is smoothly approximated by $\log(\cosh(w - w_0))$. With the optimized matrix $\widehat{\boldsymbol{B}}$, we applied the k-means clustering to the feature vectors $\{\boldsymbol{y}_i := \widehat{\boldsymbol{B}}\boldsymbol{x}_i\}_{i=1}^{200}$. The clustering performance was measured by the adjusted Rand index (ARI) [13]: The maximum value of ARI is one, and a higher value means a better clustering result.

Table 1 Averages of ARI values over 100 runs. Numbers in the parentheses are standard errors. The best and comparable methods judged by the t-test at the significance level 1% are described in boldface.

ML-GE	$\beta = 0.1$	$\beta = 0.5$	$\beta = 1.0$	$\beta = 10$	$\gamma = 1$	$\gamma = 5$	$\gamma = 10$	$\gamma = 30$
Bernoulli link ($q = 0.01$)								
0.930(0.012)	**0.932(0.012)**	**0.945(0.011)**	**0.932(0.015)**	0.729(0.031)	**0.937(0.012)**	**0.954(0.010)**	0.905(0.021)	0.878(0.021)
Bernoulli link ($q = 0.02$)								
0.819(0.017)	0.833(0.016)	**0.887(0.014)**	**0.921(0.016)**	0.675(0.030)	0.838(0.017)	**0.896(0.015)**	**0.893(0.017)**	**0.876(0.022)**
Bernoulli link ($q = 0.03$)								
0.535(0.024)	0.557(0.024)	0.684(0.025)	**0.851(0.022)**	0.686(0.032)	0.650(0.027)	0.687(0.027)	**0.804(0.025)**	**0.846(0.021)**
Categorical link								
0.075(0.008)	0.075(0.008)	0.075(0.008)	0.075(0.008)	0.077(0.008)	0.529(0.031)	0.581(0.031)	0.753(0.028)	**0.875(0.017)**

The results for Bernoulli links with outliers are presented in Table 1. When $q = 0.01$ (i.e., small contamination of outliers), all methods work well. However, as q is increased, the ARI values of ML-GE are strongly decreased, while both β-GE and RRGE still keep high ARI values. Thus, RRGE also achieves robust graph embedding as β-GE. Regarding categorical links with a constant conditional mean (Table 1), RRGE with $\gamma = 30$ shows the best performance. Even for the other γ values, RRGE still performs clearly better than ML-GE and β-GE. This would be because the distribution ratio (3) captures more general statistical dependency between link weights and data vectors. On the other hand, since β-GE is based on CME and ML-GE assumes the Poisson distribution, their performance is not so good. In short, these results indicate that RRGE is robust against outliers as β-GE yet possibly applicable on a wide-range of graph data.

4.2 Illustration on Realworld Datasets

Next, we demonstrate the performance of RRGE on realworld datasets, and follow the experimental setting in [14][†]. We employ the following datasets with binary link weights whose details are given in [14]:

WebKB: 877 nodes, 1,480 links and $d = 1{,}703$[††].
WordNet: 37,623 nodes, 312,885 links and $d = 300$[†††].
DBLP: 41,328 nodes, 210,320 links and $d = 33$ [15].

Following the shifted inner product similarity [16], we used a similarity function, $d(\boldsymbol{x}_i, \boldsymbol{x}_j) = \langle \boldsymbol{f}(\boldsymbol{x}_i), \boldsymbol{f}(\boldsymbol{x}_j)\rangle + h(\boldsymbol{x}_i) + h(\boldsymbol{x}_j)$, where $h(\boldsymbol{x})$ was modeled as a neural network. Regarding the embedding function $\boldsymbol{f}(\boldsymbol{x})$, we employed a 2-hidden layer fully-connected neural network where each hidden layer has 2,000 units and ReLU activation function. All parameters were optimized by the Adam optimizer with the learning rate $5e-4$ and minibatch size $n_{\mathrm{B}} = 64$. For WebKB, the learning rate was decreased to $5e-5$ after 100 iterations. To optimize the parameters, the following methods for binary wights $w_{ij} \in \{0, 1\}$ were employed:

LR: Since a similar form of the objective function has been used in a number of works and shown promising performance (e.g., [14], [16]), we minimized the following objective function akin to the cross entropy in logistic regression (LR):

$$\sum_{(i,j)\in\mathcal{W}'_n} \log F(d(\boldsymbol{x}_i, \boldsymbol{x}_j)) + \sum_{(i,j)\in\mathcal{I}'_{i,r}} \log\{1 - F(d(\boldsymbol{x}_i, \boldsymbol{x}_j))\}$$

where F is the logistic function, $\mathcal{W}'_n$ denotes a random subset of node pairs with *strictly* positive link wights, and $\mathcal{I}'_{i,r}$ is a random subset of $\mathcal{I}_n$ and includes r nodes randomly selected to the i-th node as in the negative sampling. Thus, the numbers of nodes in $\mathcal{W}'_n$ and $\mathcal{I}'_{i,r}$ are n_{B}^2 and rn_{B}^2, respectively. We used $r = 5$ as in [14].

RRGE: Here, we simply expressed the ratio model as $r(w_{ij}, \boldsymbol{x}_i, \boldsymbol{x}_j) = w_{ij} d(\boldsymbol{x}_i, \boldsymbol{x}_j)$. Since w_{ij} is binary, this model form simplifies a learning procedure with stochastic gradient descent. Inspired by LR, the γ-cross entropy was modified as follows:

$$\widehat{D}_\gamma(d) := -\frac{1}{\gamma}\log\Bigg[\sum_{(i,j)\in\mathcal{W}'_n}\Big\{F_{\gamma+1}(d(\boldsymbol{x}_i, \boldsymbol{x}_j))\Big\}^{\frac{\gamma}{\gamma+1}} + \sum_{(i,j)\in\mathcal{I}'_{i,1}}\Big\{1 - F_{\gamma+1}(d(\boldsymbol{x}_i, \boldsymbol{x}_j))\Big\}^{\frac{\gamma}{\gamma+1}}\Bigg].$$

We randomly selected a single data vector $\boldsymbol{x}_j$ to $\boldsymbol{x}_i$ in the second summation. This random node selection removes the statistical dependency to link weights w_{ij} and works similarly as the random permutation w^*_{ij} in (7) because w_{ij} are binary.

Evaluation task was *link prediction task*: Data samples are split into training (64%), validation (16%), test (20%) sets. Then, we optimized the parameters on the training sets, and the best model was selected using the validation set. For DBLP and WordNet (WebKB), the maximum number of iterations was 10,000 (2,000), and the model was validated at every 500 (100) iterations. Based on the validated models, we predicted links of unseen nodes on the test set, and the performance score was *area under the curve* (AUC).

Table 2 shows that the proposed method compares favorably with LR, and thus is promising on realworld datasets.

5. Conclusion

This paper proposed a novel method for graph embedding. In addition to the robustness against outliers, the key feature is that an embedding function is learned through estimation

[†]https://github.com/kdrl/WIPS
[††]https://linqs.soe.ucsc.edu/data
[†††]https://code.google.com/archive/p/word2vec

Table 2 Averages of AUC values over 10 runs. The best score for each dataset and K is described in boldface.

		LR	$\gamma = 0.1$	$\gamma = 0.3$	$\gamma = 0.5$	$\gamma = 1.0$
WebKB	$K = 10$	0.806	**0.832**	0.828	0.827	0.816
	$K = 50$	0.792	**0.815**	0.805	0.800	0.793
	$K = 100$	0.822	**0.842**	0.838	0.836	0.828
WordNet	$K = 10$	0.871	0.874	0.882	0.886	**0.893**
	$K = 50$	0.892	0.893	0.896	**0.897**	0.896
	$K = 100$	0.901	0.901	0.902	**0.904**	0.900
DBLP	$K = 10$	**0.870**	0.861	0.863	0.863	0.866
	$K = 50$	**0.876**	0.867	0.868	0.867	0.866
	$K = 100$	**0.875**	0.867	0.868	0.868	0.866

of the ratio between the conditional and marginal distributions of link weights, which possibly promotes applicability to a wider-range of graph data. The usefulness of the proposed method is demonstrated through numerical experiments both on artificial and realworld datasets.

References

[1] H. Cai, V.W. Zheng, and K.C.-C. Chang, "A comprehensive survey of graph embedding: Problems, techniques, and applications," IEEE Trans. Knowl. Data Eng., vol.30, no.9, pp.1616–1637, 2018.

[2] B. Perozzi and R. Al-Rfou, and S. Skiena, "DeepWalk: Online learning of social representations," Proc. 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.701–710, 2014.

[3] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," Proc. 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp.855–864, 2016.

[4] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," Proc. Advances in Neural Information Processing Systems (NeurIPS), 2016.

[5] W. Hamilton, Z. Ying, and J. Leskovec, "Inductive representation learning on large graphs," Proc. Advances in Neural Information Processing Systems (NeurIPS), 2017.

[6] A. Okuno, T. Hada, and H. Shimodaira, "A probabilistic framework for multi-view feature learning with many-to-many associations via neural networks," Proc. 35th International Conference on Machine Learning (ICML), pp.3888–3897, 2018.

[7] A. Okuno and H. Shimodaira, "Robust graph embedding with noisy link weights," Proc. 22nd International Conference on Artificial Intelligence and Statistics (AISTATS), pp.664–673, 2019.

[8] A. Basu, I.R. Harris, N.L. Hjort, and M.C. Jones, "Robust and efficient estimation by minimising a density power divergence," Biometrika, vol.85, no.3, pp.549–559, 1998.

[9] H. Fujisawa and S. Eguchi, "Robust parameter estimation with a small bias against heavy contamination," Journal of Multivariate Analysis, vol.99, no.9, pp.2053–2081, 2008.

[10] H. Sasaki and T. Takenouchi, "Representation learning for maximization of MI, nonlinear ICA and nonlinear subspaces with robust density ratio estimation," Journal of Machine Learning Research, in press.

[11] M. Sugiyama, T. Suzuki, and T. Kanamori, Density Ratio Estimation in Machine Learning, Cambridge University Press, 2012.

[12] H. Hung, Z.-Y. Jou, and S.-Y. Huang, "Robust mislabel logistic regression without modeling mislabel probabilities," Biometrics, vol.74, no.1, pp.145–154, 2018.

[13] L. Hubert and P. Arabie, "Comparing partitions," Journal of Classification, vol.2, no.1, pp.193–218, 1985.

[14] G. Kim, A. Okuno, K. Fukui, and H. Shimodaira, "Representation learning with weighted inner product for universal approximation of general similarities," Proc. 28th International Joint Conference on Artificial Intelligence (IJCAI), pp.5031–5038, 2019.

[15] A. Prado, M. Plantevit, C. Robardet, and J.-F. Boulicaut, "Mining graph topological patterns: Finding covariations among vertex descriptors," IEEE Trans. Knowl. Data Eng., vol.25, no.9, pp.2090–2104, 2013.

[16] A. Okuno, G. Kim, and H. Shimodaira, "Graph embedding with shifted inner product similarity and its improved approximation capability," Proc. 22nd International Conference on Artificial Intelligence and Statistics (AISTATS), pp.644–653, 2019.