On Learning Community-specific Similarity Metrics for Cold-start Link Prediction

Linchuan Xu[1], Xiaokai Wei[2], Jiannong Cao[1] and Philip S. Yu[23]
[1]The Hong Kong Polytechnic University, Hung Hom, Kowloon, Hong Kong
Email: {cslcxu,csjcao}@comp.polyu.edu.hk
[2]University of Illinois at Chicago, Chicago, Illinois, USA
Email: weixiaokai@gmail.com, psyu@uic.edu
[3]Institute for Data Science, Tsinghua University, Beijing, China

***Abstract*—This paper studies a cold-start problem of inferring new edges between vertices with no demonstrated edges but vertex content by learning vertex-based similarity metrics. Existing metric learning methods for link prediction fail to consider communities which can be observed in real-world social networks. Because communities imply the existence of local homogeneities, learning a global similarity metric is not appropriate. In this paper, we thus learn community-specific similarity metrics by proposing a community-weighted formulation of metric learning model. To better illustrate the community-weighted formulation, we instantiate it in two models, which are community-weighted ranking (CWR) model and community-weighted probability (CWP) model. Experiments on three real-world networks show that community-specific similarity metrics are meaningful and that both models perform better than those leaning global metrics in terms of prediction accuracy.**

I. Introduction

In big data era, networks are common structures of objects simply because objects are dependent and have interactions with each other, e.g., users make friends in online social networks, researchers form collaboration networks by co-authoring papers, and biological molecules have interaction in biological processes. Hence, there have been increasing interests from both industries and academia in analyzing networks. Among various network applications, link prediction [1], [2], which aims to predict potential links among network nodes, is an important step to understand and study the characteristics of networks. For example, in bioinformatics, by obtaining potential interactions, one does not need to conduct expensive experiments on all possible pairs of nodes and can spend the resource wisely on the most likely interactions [3]. For social network applications, such as Facebook and Twitter, it is fundamental to grow the user base and enhance user experience with link prediction techniques.

The link prediction [4] is usually performed by measuring the similarities between them. The similarities can be measured on two types of information, edges between vertices and vertex content. Previously, edges are used to construct various edge-based similarity metrics, such as Common Neighbors [5], [4] and more sophisticated ones for heterogeneous networks [6]. Recently, edges are encoded in node representations by network embedding methods, such as DeepWalk [7], LINE [8] and node2vec [9]. Vertex-based metrics can be constructed on different content in different domains, such as tweets in Twitter, and abstract of academic papers.

But in some scenarios, the edges of certain nodes may be unavailable, e.g.,

- Social network users may set a privacy policy that limits the visibility of their connections.
- At the time of registration, a new user has no connections within the social network.
- When a social network is made open to public for the first time, there may be plenty of vertices without any connections.
- When an academic paper is being drafted, it may have no references to other papers.

The particular challenge of link prediction for these vertices without edges is the cold-start problem.

In the cold-start link prediction problem, edge-based similarity metrics and network embedding methods are all out of usage. Hence one can only rely on vertex content to devise vertex-based similarity metrics. Although similarity metric learning has been studied for link prediction [10], [11], almost all studies fail to consider communities. In this case, vertices of social networks are assumed to share global homogeneities. However, plenty of studies [12], [13] have revealed that vertices of real-world social networks usually tend to form clusters or communities, which are groups of vertices having more intra-group interactions than inter-group interactions. For example, co-authorships are more likely to take place within each domain than across different domains. Accordingly, real-world social networks actually display a mixture of intra-community homogeneities.

Intra-community homogeneities have two implications to metric learning. Firstly, the communities to which vertices belong have to be considered because intra-community interactions are more likely to occur than inter-community ones as discussed above. Secondly, the similarities measured on the same vertex content but in different communities may be different. This is because semantic meanings of certain terms may vary from one community to another, e.g., “language” may refer to a type of programming language in Software Engineering community while in Natural Language Processing, it may refer to a type of spoken language. Hence, each community should be associated with a corresponding

978-1-5090-6014-6/18/$31.00 ©2018 IEEE

similarity metric, which is referred to as community-specific metric.

In this paper, we thus propose a community-weighted formulation of metric learning model to learn community-specific metrics. To better illustrate the formulation, we instantiate it in two models, i.e., community-weighted ranking (CWR) model and community-weighted probability (CWP) model. The CWR learns the metrics by enforcing similarities of vertices connected by edges to be larger than that of those not connected, which follows the original approach to link prediction [4], i.e., inferring interactions between vertices with similarities ranked in the top. The CWP formulates the link prediction as a binary classification problem by estimating the probabilities of edges based on the similarities. We study the two models because ranking and classification are the two major streams of formulation of the link prediction problem [14].

The contributions of the paper are summarized as follows:

- To the best of our knowledge, this is the first attempt to learn community-specific similarity metrics for the cold-start link prediction problem.
- This paper shows the community-weighted formulation as an effective way to learn community-specific similarity metrics, which are demonstrated in two models, i.e., the CWR model and the CWP model.
- We present comprehensive evidence showing the community-specific metrics outperform global ones.

II. Related Work

The first category of related work should be studying the cold-start link prediction problem where the given network provides limited edge information, and in extreme cases [15], no edges at all. Although various studies [15] [16] proposed different approaches to handling the cold-start problem, we are different from them in two aspects. Firstly, we propose a similarity metric learning method while previous methods perform no metric learning at all. Secondly, previous studies target at problems where there exist heterogeneous auxiliary networks, which are unavailable in our problem settings.

The second category is similarity metric learning for link prediction. Similarity metric learning is the key to link prediction problem, which is initially solved by measuring the similarity between vertices [4] based on the intuition that links are usually formed between "close" vertices. And afterwards, there emerged a large number of approaches to constructing and learning similarity metrics. Here we introduce a couple of representatives.

There are major two categories of similarity metrics, edge-based ones and vertex-based ones. Edge-based similarity metrics are constructed from edges between vertices, such as Common Neighbors [5] and Rooted PageRank [4]. These similarity metrics are generic. Vertex-based similarity metrics, by contrast, are specific to the domain, such as Keyword Match Count [17] in academic social networks. These domain-specific metrics have been demonstrated to have significant improvements on link prediction tasks [17]. Some studies [18] even combine edges and vertex content into a single similarity metric. All these similarity metrics are initially unweighted. Afterwards, studies revealed weighted similarity metrics perform better than unweighted [19]. Hence, studies trying to learn an appropriate similarity metric emerged.

A stream of metric learning is to learn relationship strengths based on edges. Some studies [10] formulated the strengths as output of non-linear functions with similarity metrics mentioned above as inputs. More recent studies [20] ignored the commonly used similarity metrics, but proposed more sophisticated models to learn the relationship strengths.

The proposed models learn vertex-based similarity metrics. The major difference from previous studies is the learning of community-specific metrics instead of global ones.

III. Methodology

A. The Studied Problem

Given a network $G(V_1, F_1, E)$, where V_1 is a set of vertices, F_1 is a set of features, and E is the set of edges among V_1, the objective is to learn feature-based similarity metrics to infer interactions among V_2, where V_2 and V_1 belong to the same network, and $V_2 \cap V_1 = \emptyset$.

B. The Similarity Metric

The studied similarity metric is weighted feature similarity (WFS) which is in the same form of commonly used weighted Jarccard coefficient, and it is denoted as follows:

$$\text{WFS}(\boldsymbol{x}_i, \boldsymbol{x}_j | \boldsymbol{w}) = \frac{\boldsymbol{w}^\top (\boldsymbol{x}_i \otimes \boldsymbol{x}_j)}{\boldsymbol{w}^\top (\boldsymbol{x}_i \oplus \boldsymbol{x}_j)}, \tag{1}$$

where $\boldsymbol{w} \in \mathbb{R}^d$ denotes the column vector of feature weights, d is the number of features, $\boldsymbol{x}_i \in \mathbb{R}^d$ and $\boldsymbol{x}_j \in \mathbb{R}^d$ denotes the column feature vector of vertex i and vertex j, respectively, and $x_{ij} = 1$ if vertex i contains feature j or $x_{ij} = 0$ otherwise. $\otimes$ is a binary function which returns a vector with each element equal to logical "and" of two corresponding elements of input vectors while $\oplus$ returns a vector with each element equal to logical "or" of input vectors. In the rest of paper, we let $s(\boldsymbol{x}_i, \boldsymbol{x}_j)$ denote $\text{WFS}(\boldsymbol{x}_i, \boldsymbol{x}_j)$. It is worthy of noting that the paper uses weighted feature similarity as an example, and can be applied to other feature-based similarities.

C. Ranking Models

This section starts with global ranking model, and then extends the global model to community-weighted ranking model.

1) Global Ranking Model: The similarity metric should fulfill the expectation that similarities of linked pairs of vertices should be larger than those of non-linked pairs, which can be formulated as follows:

$$s(\boldsymbol{x}_i, \boldsymbol{x}_j | \boldsymbol{w}) > s(\boldsymbol{x}_h, \boldsymbol{x}_k | \boldsymbol{w}), \tag{2}$$

where $(i, j) \in E$ and $(h, k) \notin E$ here and in the rest of the paper.

If and only if Eq. (2) is violated, $\boldsymbol{w}$ should be penalized. Accordingly, total penalties can be quantified as follows:

$$L(\boldsymbol{w}) = \sum g(s(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}) - s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w})), \tag{3}$$

which is summarized on the number of edges. Non-linked pairs are randomly sampled, and $g(x)$ is defined below.

Firstly, the sigmoid function is introduced to the similarity difference as follows:

$$h(x) = \frac{1}{1 + \exp\{-\left[s(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}) - s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w})\right]\}}, \tag{4}$$

In this way, the violation condition, i.e., violating Eq. (2), does not need to be checked on each pair of edge and non-existing edge, which makes the penalty function continuous. More importantly, the subsequent optimization is simplified.

Then the regularized logistic loss is employed as the penalty function:

$$L(\boldsymbol{w}) = -\sum \log(1 - \frac{1}{1 + \exp\{-d(s_f, s_t)\}}) + \eta||\boldsymbol{w}||_2^2, \tag{5}$$

where $d(s_f, s_t) = s(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}) - s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w})$, $\eta \in \mathbb{R}$, and $||\cdot||_2^2$ is F_2 norm. Since $L(\boldsymbol{w})$ is a convex function, we may solve this minimization problem by gradient descent.

2) Community-weighted Formulation of Similarity: As discussed in the introduction, the community-weighted formulation of similarity should be different from the global formulation in two aspects. Firstly, the community information should be associated with each vertex. Secondly, community-specific similarity metrics should be adopted to measure the similarities instead of the global one.

Considering the fact that each vertex may belong to multiple communities as a result of overlapping communities [21] in real-world social networks, we adopt the soft community assignment, which means that every community is assigned to each vertex with a probability. This soft community assignment can be achieved by the commonly used multi-class logistic regression model defined as follows:

$$\pi_a(\boldsymbol{x}_i) = \frac{\exp\{\boldsymbol{v}_a^\top \boldsymbol{x}_i + b_a\}}{\sum_{t=1}^{A} \exp\{\boldsymbol{v}_t^\top \boldsymbol{x}_i + b_t\}}, \tag{6}$$

where $\pi_a(\boldsymbol{x}_i)$ is the probability that vertex i belongs to community a, $\boldsymbol{v}_a \in \mathbb{R}^d$ is the vector of centroid a to be estimated [22], A is the total number of communities, and $b \in \mathbb{R}$ is a bias term. According to Eq. (6), larger probabilities would be assigned to communities to which vertex i has more features belonging.

With soft community assignment defined, when computing the similarity of two vertices, every possible community of each vertex should be considered. It is intuitive that two vertices may belong to the same community. But they may belong to different communities as well. Taking the paper citation network as an example, papers from various research fields have cited papers from machine learning field, which is because machine learning techniques have been widely employed for knowledge management in other fields, such as retrieval in database [23], and network traffic management in networking [24].

Hence, considering all possible combinations of communities, the similarity between two vertices using community-specific similarity metrics is quantified as follows:

$$s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{W}) = \sum_a \sum_b \pi_a(\boldsymbol{x}_i)\pi_b(\boldsymbol{x}_j)s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{ab}), \tag{7}$$

where $\boldsymbol{W}$ is a collection of community-specific weight vectors $\{\boldsymbol{w}_{ab}\}$. When $a = b$, $\boldsymbol{w}_{ab}$ is the weight vector for community a. Otherwise, $\boldsymbol{w}_{ab}$ is the weight vector for measuring similarities of vertices from community a and community b.

3) Community-weighted Ranking (CWR) Model: The regularized loss for the CWR model extended from the global ranking model can be drawn as follows:

$$\begin{aligned} L(\boldsymbol{V}, \boldsymbol{W}) = &-\sum \log(1 - \frac{1}{1 + \exp\{-d(s_f, s_t|\boldsymbol{W})\}}) \\ &+ \lambda \sum_{a=1}^{A} ||\boldsymbol{v}_a||_2^2 + \beta \sum_{ab} ||\boldsymbol{w}_{ab}||_2^2, \end{aligned} \tag{8}$$

where $\lambda \in \mathbb{R}$ and $\beta \in \mathbb{R}$.

Eq. (8) is not jointly convex on all the variables, i.e., centroids and weight vectors. We thus may solve it alternatingly by gradient-based algorithms, such as gradient descent and L-BFGS, with the other type of variables fixed. The derivative w.r.t weight vectors can be obtained as follows: $\frac{\partial L(V,W)}{\partial \boldsymbol{w}_{ab}} =$

$$\begin{aligned} &\sum \Bigg[\frac{\mathrm{d}s_{hk}^{ab}}{\mathrm{d}\boldsymbol{w}_{ab}}\pi_a(\boldsymbol{x}_h)\pi_b(\boldsymbol{x}_k) + I \times \frac{\mathrm{d}s_{hk}^{ab}}{\mathrm{d}\boldsymbol{w}_{ab}}\pi_b(\boldsymbol{x}_h)\pi_a(\boldsymbol{x}_k) \\ &- \frac{\mathrm{d}s_{ij}^{ab}}{\mathrm{d}\boldsymbol{w}_{ab}}\pi_a(\boldsymbol{x}_i)\pi_b(\boldsymbol{x}_j) + I \times \frac{\mathrm{d}s_{ij}^{ab}}{\mathrm{d}\boldsymbol{w}_{ab}}\pi_b(\boldsymbol{x}_i)\pi_a(\boldsymbol{x}_j)\Bigg] \\ &\times \frac{1}{1 + \exp\{-d(s_f, s_t)\}} + 2\eta\boldsymbol{w}_{ab}, \end{aligned} \tag{9}$$

where $I = 1$ when $a \neq b$, $I = 0$ otherwise, and $\frac{\mathrm{d}s_{hk}^{ab}}{\mathrm{d}\boldsymbol{w}_{ab}}$ is the abbreviations of $\frac{\mathrm{d}s(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}_{ab})}{\mathrm{d}\boldsymbol{w}_{ab}}$, which is as follows:

$$\left\{\frac{(\boldsymbol{x}_i \otimes \boldsymbol{x}_j)\boldsymbol{w}_{ab}^\top(\boldsymbol{x}_i \oplus \boldsymbol{x}_j) - \boldsymbol{w}_{ab}^\top(\boldsymbol{x}_i \otimes \boldsymbol{x}_j)(\boldsymbol{x}_i \oplus \boldsymbol{x}_j)}{\left[\boldsymbol{w}_{ab}^\top(\boldsymbol{x}_i \oplus \boldsymbol{x}_j)\right]^2}\right\}, \tag{10}$$

, and $\frac{\mathrm{d}s_{ij}^{ab}}{\mathrm{d}\boldsymbol{w}_{ab}}$ is similarly defined.

The derivative w.r.t centroids are as follows: $\frac{\partial L(\boldsymbol{V},\boldsymbol{W})}{\partial \boldsymbol{v}_a} =$

$$\begin{aligned} &\sum \Bigg\{\sum_{b\neq a}\Big[\frac{\mathrm{d}\pi_a(\boldsymbol{x}_h)}{\mathrm{d}\boldsymbol{v}_a}\pi_b(\boldsymbol{x}_k) + \frac{\mathrm{d}\pi_a(\boldsymbol{x}_k)}{\mathrm{d}\boldsymbol{v}_a}\pi_b(\boldsymbol{x}_h)\Big]s_{hk}^{ab} \\ &+ \frac{\mathrm{d}\left[\pi_a(\boldsymbol{x}_h)\pi_a(\boldsymbol{x}_k)\right]}{\mathrm{d}\boldsymbol{v}_a}s_{hk}^{aa} - \sum_{b\neq a}\Big[\frac{\mathrm{d}\pi_a(\boldsymbol{x}_i)}{\mathrm{d}\boldsymbol{v}_a}\pi_b(\boldsymbol{x}_j) \\ &+ \frac{\mathrm{d}\pi_a(\boldsymbol{x}_j)}{\mathrm{d}\boldsymbol{v}_a}\pi_b(\boldsymbol{x}_i)\Big]s_{ij}^{ab} - \frac{\mathrm{d}\left[\pi_a(\boldsymbol{x}_i)\pi_a(\boldsymbol{x}_j)\right]}{\mathrm{d}\boldsymbol{v}_a}s_{ij}^{aa}\Bigg\} \\ &\times \frac{1}{1 + \exp\{-d(s_f, s_t|W)\}} + 2\lambda\boldsymbol{v}_a, \end{aligned} \tag{11}$$

where $\frac{\mathrm{d}\pi_a(\boldsymbol{x}_i)}{\mathrm{d}\boldsymbol{v}_a} = \left[1 - \pi_a(\boldsymbol{x}_i)\right]\pi_a(\boldsymbol{x}_i)\boldsymbol{x}_i$, and

$$\frac{\mathrm{d}\left[\pi_a(\boldsymbol{x}_i)\pi_a(\boldsymbol{x}_j)\right]}{\mathrm{d}\boldsymbol{v}_a} = \frac{\mathrm{d}\pi_a(\boldsymbol{x}_i)}{\mathrm{d}\boldsymbol{v}_a}\pi_a(\boldsymbol{x}_j) + \frac{\mathrm{d}\pi_a(\boldsymbol{x}_j)}{\mathrm{d}\boldsymbol{v}_a}\pi_a(\boldsymbol{x}_i), \tag{12}$$

and $\frac{\mathrm{d}\pi_a(\boldsymbol{x}_h)}{\mathrm{d}\boldsymbol{v}_a}$, $\frac{\mathrm{d}\pi_a(\boldsymbol{x}_k)}{\mathrm{d}\boldsymbol{v}_a}$, $\frac{\mathrm{d}\pi_a(\boldsymbol{x}_j)}{\mathrm{d}\boldsymbol{v}_a}$, $\frac{\mathrm{d}\pi_b(\boldsymbol{x}_i)}{\mathrm{d}\boldsymbol{v}_b}$, $\frac{\mathrm{d}\pi_b(\boldsymbol{x}_j)}{\mathrm{d}\boldsymbol{v}_b}$, $\frac{\mathrm{d}[\pi_a(\boldsymbol{x}_h)\pi_a(\boldsymbol{x}_k)]}{\mathrm{d}\boldsymbol{v}_a}$ are similarly defined.

D. Probability Models

1) Global Probability Model: The probability of a link estimated on the similarity can be obtained by the sigmoid function, which is defined as follows:

$$p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}) = \frac{1}{1 + \exp\{-s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w})\}}, \tag{13}$$

To learn an appropriate $\boldsymbol{w}$, both small probabilities of existing links and large ones of non-existing links should be penalized. The logistic loss is employed to perform the penalty. With an ℓ_2-norm regularization term on $\boldsymbol{w}$ to control the complexity, the objective function is formulated as follows:

$$\begin{aligned} L(\boldsymbol{w}) = &-\sum_{(i,j)\in E} \log(p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w})) \\ &- \sum_{(h,k)\notin E} \log(1 - p(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w})) + \eta||\boldsymbol{w}||_2^2, \end{aligned} \tag{14}$$

which is convex, and can be solved by gradient descent.

2) Community-weighted Formulation of Probability: Similar to the way of extending from the global formulation of similarity to the community-weighted formulation, the community-weighted formulation of probability should be summarized on community-specific probabilities weighted for the communities. Hence, the community-weighted formulation of probability is as follows:

$$p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{W}) = \sum_a \sum_b \pi_a(\boldsymbol{x}_i)\pi_b(\boldsymbol{x}_j)p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{ab}) \tag{15}$$

3) Community-weighted Probability (CWP) Model: The loss function of the CWP model extended from the global probability model is formulated as $L(V, W) =$

$$\begin{aligned} &-\sum_{(i,j)\in E} \left[\log \sum_a \sum_b \pi_a(\boldsymbol{x}_i)\pi_b(\boldsymbol{x}_j)p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{ab})\right] \\ &- \sum_{(h,k)\notin E} \left[\log(1 - \sum_a \sum_b \pi_a(\boldsymbol{x}_h)\pi_b(\boldsymbol{x}_k)p(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}_{ab}))\right] \\ &+ \lambda \sum_{a=1}^{A} ||\boldsymbol{v}_a||_2^2 + \eta \sum_{ab} ||\boldsymbol{w}_{ab}||_2^2, \end{aligned} \tag{16}$$

The loss function can also be minimized in a similar way of the loss function of the CWR model. The derivative w.r.t weight vectors can be obtained as follows: $\frac{\partial L(V,W)}{\partial \boldsymbol{w}_{ab}} =$

$$\begin{aligned} &-\sum \Bigg[\frac{p^2(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{ab})}{p(\boldsymbol{x}_i, \boldsymbol{x}_j|W)} \exp\{-s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{ab})\}[\pi_a(\boldsymbol{x}_i)\pi_b(\boldsymbol{x}_j) \\ &+ I \times \pi_b(\boldsymbol{x}_i)\pi_a(\boldsymbol{x}_j)] \frac{\mathrm{d}s(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{ab})}{\mathrm{d}\boldsymbol{w}_{ab}}\Bigg] \\ &-\sum \Bigg[\frac{p^2(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}_{ab})}{p(\boldsymbol{x}_h, \boldsymbol{x}_k|W) - 1} \exp\{-s(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}_{ab})\}[\pi_a(\boldsymbol{x}_h)\pi_b(\boldsymbol{x}_k) \\ &+ I \times \pi_b(\boldsymbol{x}_h)\pi_a(\boldsymbol{x}_k)] \times \frac{\mathrm{d}s(\boldsymbol{x}_h, \boldsymbol{x}_k|\boldsymbol{w}_{ab})}{\mathrm{d}\boldsymbol{w}_{ab}}\Bigg] + 2\eta\boldsymbol{w}_{ab}, \end{aligned} \tag{17}$$

where subscripts of the summation are omit due to space consideration here and in the following equation.

Algorithm 1 Alternating Optimization Algorithm

```
Input:$G(V, E)$, A, $\lambda$, and $\eta$
Output:Weight vectors $\boldsymbol{W}$ and centroid vectors $\boldsymbol{V}$

Initializing $\boldsymbol{V} = \{\boldsymbol{v}_1, ..., \boldsymbol{v}_a, ..., \boldsymbol{v}_A\} \leftarrow$ k-means
repeat(not converge)
  Fix $\boldsymbol{V}$, find optimal $\boldsymbol{W}$ with gradient descent;
  Fix $\boldsymbol{W}$, find optimal $\boldsymbol{V}$ with gradient descent;
until $iteration = iteration_{max}$ or converge
end return $\boldsymbol{W}$,$\boldsymbol{V}$
```

	BlogCatalog	PubMed	DBLP
# communities	4	3	3
# vertices	5567	19717	11512
Ave. non-zero features per vertex	89.83	50.11	60.57
# total features	8675	500	8172
# links	21775	44338	11996

TABLE I: Network statistics

The derivative w.r.t centroids is as follows: $\frac{\partial L(V,W)}{\partial \boldsymbol{v}_a} =$

$$\begin{aligned} &-\sum \Bigg[\frac{1}{p_{ij}} \times \Bigg\{\sum_{b\neq a} \Big[\frac{\mathrm{d}\pi_a(\boldsymbol{x}_i)}{\mathrm{d}\boldsymbol{v}_a}\pi_b(\boldsymbol{x}_j) + \frac{\mathrm{d}\pi_a(\boldsymbol{x}_j)}{\mathrm{d}\boldsymbol{v}_a}\pi_b(\boldsymbol{x}_i)\Big] p_{ij}^{ab} \\ &+ \frac{\mathrm{d}\left[\pi_a(\boldsymbol{x}_i)\pi_a(\boldsymbol{x}_j)\right]}{\mathrm{d}\boldsymbol{v}_a} p_{ij}^{aa}\Bigg\}\Bigg] \\ &-\sum \Bigg[\frac{1}{(p_{hk} - 1)} \times \Bigg\{\sum_{b\neq a} \Big[\frac{\mathrm{d}\pi_a(\boldsymbol{x}_h)}{\mathrm{d}\boldsymbol{v}_a} \times \pi_b(\boldsymbol{x}_k) \\ &+ \frac{\mathrm{d}\pi_a(\boldsymbol{x}_k)}{\mathrm{d}\boldsymbol{v}_a}\pi_b(\boldsymbol{x}_h)\Big] p_{hk}^{ab} + \frac{\mathrm{d}\left[\pi_a(\boldsymbol{x}_h)\pi_a(\boldsymbol{x}_k)\right]}{\mathrm{d}\boldsymbol{v}_a} p_{hk}^{aa}\Bigg\}\Bigg] + 2\lambda\boldsymbol{v}_a, \end{aligned} \tag{18}$$

where $p_{ij} = p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{W})$, $p_{ij}^{aa} = p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{aa})$, and p_{ij}^{ab} is equal to $p(\boldsymbol{x}_i, \boldsymbol{x}_j|\boldsymbol{w}_{ab})$.

4) The Optimization Algorithm: The optimization algorithm for the CWR model and the CWP model is similar, and hence we present a unified one for both of them. It implements the block-coordinate decent [25] algorithm, which solves a joint optimization problem by fixing a subset of variables, and then optimizing the objective w.r.t the rest subset of variables. This process is iteratively executed until convergence. This algorithm is employed because firstly, the optimization objective is to jointly optimize community centroids and community-specific similarity metrics. Secondly, community-specific similarity metrics are affiliated with communities, and hence they can only be determined when the communities are fixed. The pseudo-codes are presented in Algorithm 1.

The centroids are initialized by k-means clustering on vertex feature vectors, which acts as pre-training to render the learning process more effective. Hence, even though Algorithm 1 cannot guarantee the global optimal, it can work out an effective local optimal, which performs well in the experiments. Weight vectors are initialized as all one vectors, and backtracking line search [26] is used to determine learning rates for iterations of gradient descent. During the learning process, non-linked pairs of vertices in both ranking models and probability models are randomly sampled to the same number of linked-pairs so that neither of the two types of pairs dominates the data space.

Referring to the derivatives derived above, the complexity of Algorithm 1 is $O(|E|A^2m^2)$, where m is the number of features. Moreover, the convergence of the algorithm can be guaranteed by general proof of block-wise coordinate descent algorithm [27]. In the following experiments, Algorithm 1 can usually converge after around 10 outer iterations.

IV. Empirical Evaluation

A. Datasets

One social network, BlogCatalog [28]. and two citation networks, PubMed Diabetes [29] and DBLP [30], is used to evaluate the models.

Table 1 presents statistics for the three networks used in our experiments. For BlogCatalog, we sample four popular communities, Art, Technology, Development and Growth, and Finance. Moreover, the friends of each user are also restricted in one of these four communities. The threshold of frequency for filtering rare words in blogs is set as 10. For PubMed, we use the entire network provided by the dataset.

From DBLP, we select three communities, database, machine learning, and networking. And from each community, we select a couple of popular conferences, including but not limited to SIGMOD, VLDB, ICDE for database, AAAI, ICML, NIPS for machine learning, and SIGCOMM, GLOBECOM, INFOCOM for networking. For the time span, we sample a sub-network from 2000 and 2012, which means that not only papers selected have to be published within the time span, but also the references counted have to be published in this time span. For feature extraction, we use all the keywords of abstract for each paper except for stop words, such as "there" and "where". Moreover, we set a threshold of frequency of keyword as 7 to filter out those rare keywords.

B. Experiment Settings

For baselines, we may consider existing approaches [15], [16] to handling the cold-start link prediction problem, but both of them rely on heterogeneous auxiliary networks, which are unavailable in the studied problem. With respect to general methods, such as edge-based similarity metrics, matrix [31] or tensor factorization, or even more recent techniques [32], [7], [8], they are not applicable as well since there are no edges demonstrated by test vertices. As a result, no existing baselines are suitable for the proposed cold-star link prediction problem. Nevertheless, the global models may act as alternatives.

For the implementation of Algorithm 1, regularization coefficients are set as 1, commonly used settings are used in backtracking line search, and relative loss of 0.001 is set as the converge criterion for gradient descent. For the number of communities, we set it as the ground-truth number for each network for simplification. Other methods such as cross-validation, or more sophisticated clustering methods [33] can be employed to learn an appropriate number of communities when the ground truth number is unknown.

C. Overview of Community-Specific Metrics

To demonstrate the effective of learning community-specific similarity metrics, we present important features learned in the global similarity metrics and community-specific similarity metrics in Table 2. Due to limited space, we only present top 10 features ranked according to the weights learned by probability models on the DBLP dataset. From Table 2, we obtain three important observations. Firstly, we can see that for each community, the top 10 features are community-specific, i.e., they are strongly related to the nature of the community. Hence, the CWP model can well learn the community-specific similarity metrics.

Secondly, top 10 features of the global model is a mixture of features from the three communities. This observation demonstrates that learning a global similarity metric is not appropriate for link prediction when there are multiple communities. For a simple example, a certain feature, such as "internet" ranking highly due to its importance in networking in the global model may rank low in the database community, and this feature would definitely bring negative influence to link prediction in the database community.

Thirdly, top 10 features in the similarity metrics for measuring similarities between two papers from different communities, such as Database (1)&Machine Learning (2), are a mixture of features from the two communities. Given that the community-specific similarity metrics are correctly learned by the CWP model, these features may appear frequently in the both communities, and indicate potentials of cross-community citations.

D. Performance of Link Prediction

We conduct 4 runs of experiments for all the three networks, and use different percentage of links as training data in each run. An extra model, equal weight ranking (EWR), is employed as a baseline in which an equal weight is assigned to each feature and is directly used in the similarity measurement. The performance measured on area under the curve (AUC) is presented in Table 3. Comparing the equal weight ranking (EWR) model with the global ranking (GR) model and community-weighted ranking (CWR) model, we see the proposed metric learning method is necessary and very effective.

Among all the models, the CWR model and the CWP model perform the best. The reason behind the advantage can be obtained from the observation that features are actually of different importance in difference communities as illustrated in the section above. We provide a case study to demonstrate why learning community-specific similarity metrics have such benefits in the following section. The reason why all the models perform better on PubMed and DBLP than on BlogCatalog may be that features in professional social networks are more

Global Model	Community-weighted Probability Model					
	Database (1)	1&2	**Machine learning (2)**	2&3	**Networking (3)**	1&3
ranking	**XML**	ranking	**tracking**	query	**internet**	tracking
retrieval	**TREC**	clustering	**shape**	distributed	**routing**	localization
routing	**language**	privacy	**clustering**	stream	**mobility**	predictive
internet	**ranking**	stream	**face**	streams	**traffic**	informative
mobile	**learning**	schema	**learning**	privacy	**sensor**	images
traffic	**search**	collaborative	**matching**	queries	**mobile**	representations
clustering	**test**	matrix	**stereo**	peer-to-peer	**packet**	scene
wireless	**expansion**	image	**vision**	answer	**throughput**	state
mobility	**measures**	language	**motion**	database	**energy**	inference
XML	**complexity**	XML	**recognition**	localization	**interference**	property

TABLE II: Top 10 features learned from probability models

AUC	BlogCatalog				PubMed				DBLP			
	10%	30%	50%	70%	10%	30%	50%	70%	10%	30%	50%	70%
EWR	53.92	54.85	54.36	54.62	65.56	66.34	66.01	65.39	71.55	72.32	71.69	72.22
GR	54.66	56.21	58.72	60.03	81.86	84.12	86.05	87.61	81.55	87.26	89.01	89.95
CWR	57.02	59.67	61.12	**63.05**	**84.56**	**86.89**	**89.01**	**90.16**	**85.67**	**91.89**	**92.82**	**92.15**
GP	54.58	56.66	58.82	59.07	77.22	83.63	84.15	85.36	72.48	86.21	87.51	87.46
CWP	**57.74**	**60.76**	**62.01**	62.65	79.08	85.63	86.05	87.22	76.18	88.38	88.69	90.30

TABLE III: Performance comparison on AUC scores (%), where percentage numbers are the percentage of links used as training data. EWR, GR and GP are abbreviations of equal weight ranking, global ranking and global probability, respectively.

Paper 1	Database : 0.09 Machine Learning: 0.81 Networking : 0.08	**Paper abstract**: trajectory planning and optimization is a fundamental problem in articulated robotics. Algorithms used typically for this problem compute optimal trajectories from scratch in a new situation. In effect, extensive data is accumulated containing situations ...
Common Words	Global	optimization, find, extensive, effect, show, propose, paper
	Database	optimization, effect, extensive, find, show, propose, paper
	Machine Learning	optimization, extensive, find, paper, effect, show, propose
	Networking	find, effect, show, optimization, extensive, paper, propose
Paper 2	Database: 0.32 Machine Learning : 0.04 Networking : 0.64	**Paper abstract**: in a peer-to-peer (P2P) live streaming system, the streaming quality of an end user is much affected by the aggregate download bandwidth from the partners. In this paper, we propose a stochastic model for the P2P streaming system to analyze ...

TABLE IV: A case study on prediction where "Database: 0.09" denotes the estimated probability that this paper belongs to Database is 0.09, "Machine Learning: 0.81" and "Networking: 0.08" are similarly defined.

formal and thus more informative than those in the online social networks.

E. Community-specific Metrics on Prediction

In this section, we explore the reason why community-weighted models outperform global models by examining how they produce predictions on the same test data. Due to the limited space, we only present one representative in Table 4, where the second column presents the probabilities of paper field membership, and common words of abstract of the two papers. The common words are decreasingly ranked according to the weights in the corresponding similarity metric. The ground truth is that there is no citation between these two papers.

Firstly, by looking at field membership probabilities and the abstracts, the soft community assignment is consistent with the ground truth. It is intuitively that this two papers are not much likely to have a citation relationship since they are from different research fields. Secondly, features such as "optimization" have different semantic meanings between machine learning and networking, which have been well preserved in the community-specific metrics. However, the global model assigns very high importance to "optimization" regardless of its low importance in networking field. As a result, the probability of citation given by the global model is 0.55, and the probability given by the CWP model is 0.06. It is not difficult to find that the high probability given by the global model is because it fails to consider research fields of papers and making no difference between semantic meanings of the same feature in different fields.

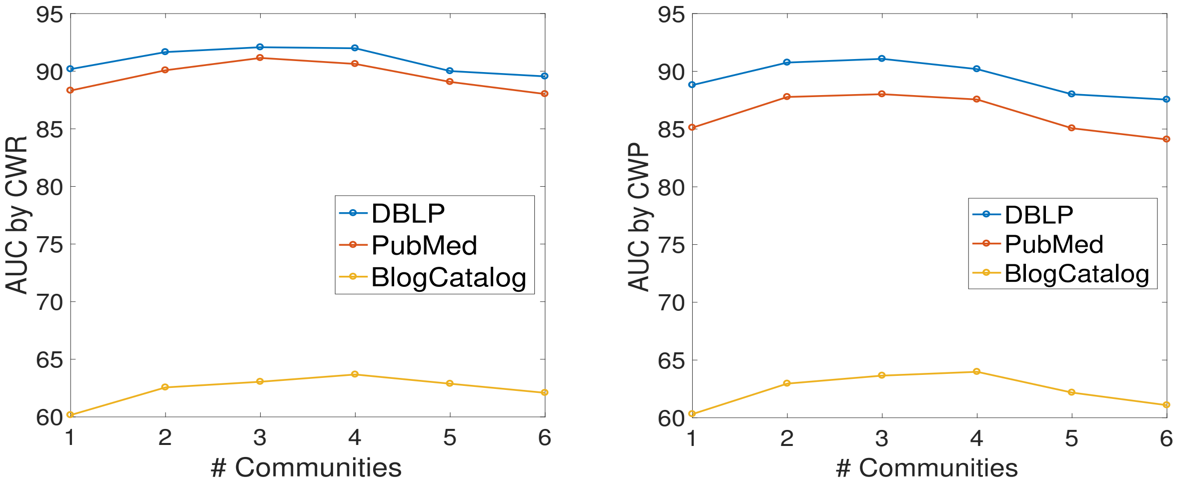

Fig. 1: Parameter sensitivity

F. *Parameter Sensitivity*

The performance w.r.t the number of communities specified in the models is studied, and corresponding AUC scores of tasks where 90% of links are used as training data are presented in Fig. 1. The ground-truth number of communities in BlogCatalog, PubMed, and DBLP is 4, 3, and 3 respectively. Fig. 1 shows the performance of the proposed models gets better as the number of communities grows, reaches the top at the ground-truth number, but starts to drop when it is larger than the ground truth, and is even worse than the global models on the DBLP and PubMed datasets.

The first half of the pattern is expected because the community is the central idea of this paper, and for which the community-weighted models outperform global models. The second half suggests it is not always the case that a larger number of communities would bring better performance, especially when the number is considerably larger than the ground-truth number (e.g. 2 times). Hence, the number of communities should be carefully specified, and methods such as cross-validation and sophisticated clustering methods [33] are suggested to learn an appropriate one when the ground truth is unknown.

V. Conclusion and Future Work

In this paper, we propose a community-weighted formulation to learn community-specific similarity metrics for the cold-start link prediction problem, which is instantiated in two models, i.e., community-weighted probability (CWP) model and community-weighted similarity ranking (CWR) model. Experimental results show that intra-community homogeneities can be well preserved, and that community-weighted models outperform global models in terms of accuracy. In the future, we plan to enable our models to automatically learn an optimal number of communities.

Acknowledgment

The work described in this paper was partially supported by the funding for Project of Strategic Importance provided by The Hong Kong Polytechnic University (Project Code: 1-ZE26), RGC General Research Fund under Grant PolyU 152199/17E, NSFC Key Grant with Project No. 61332004, NSF through grants IIS-1526499, and CNS-1626432, and NSFC 61672313.

References

[1] L. A. Adamic and E. Adar, "Friends and neighbors on the web," *Social networks*, vol. 25, no. 3, pp. 211–230, 2003.

[2] L. Backstrom and J. Leskovec, "Supervised random walks: predicting and recommending links in social networks," in *Proceedings of the fourth ACM international conference on Web search and data mining*. ACM, 2011, pp. 635–644.

[3] X. Wei, L. Xu, B. Cao, and P. S. Yu, "Cross view link prediction by learning noise-resilient representation consensus," in *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017, pp. 1611–1619.

[4] D. Liben-Nowell and J. Kleinberg, "The link-prediction problem for social networks," *Journal of the American society for information science and technology*, vol. 58, no. 7, pp. 1019–1031, 2007.

[5] M. E. Newman, "Clustering and preferential attachment in growing networks," *Physical Review E*, vol. 64, no. 2, p. 025102, 2001.

[6] B. Cao, X. Kong, and P. S. Yu, "Collective prediction of multiple types of links in heterogeneous information networks," in *Proceedings of IEEE International Conference on Data Mining (ICDM)*, 2014.

[7] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 701–710.

[8] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "Line: Large-scale information network embedding," in *Proceedings of the 24th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2015, pp. 1067–1077.

[9] A. Grover and J. Leskovec, "node2vec: Scalable feature learning for networks," in *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2016, pp. 855–864.

[10] N. Li, X. Feng, S. Ji, and K. Xu, "Modeling relationship strength for link prediction," in *Intelligence and Security Informatics*. Springer, 2013, pp. 62–74.

[11] I. Kahanda and J. Neville, "Using transactional information to predict link strength in online social networks." *ICWSM*, vol. 9, pp. 74–81, 2009.

[12] M. Girvan and M. E. Newman, "Community structure in social and biological networks," *Proceedings of the national academy of sciences*, vol. 99, no. 12, pp. 7821–7826, 2002.

[13] M. E. Newman, "The structure and function of complex networks," *SIAM review*, vol. 45, no. 2, pp. 167–256, 2003.

[14] L. Lü and T. Zhou, "Link prediction in complex networks: A survey," *Physica A: Statistical Mechanics and its Applications*, vol. 390, no. 6, pp. 1150–1170, 2011.

[15] V. Leroy, B. B. Cambazoglu, and F. Bonchi, "Cold start link prediction," in *Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2010, pp. 393–402.

[16] L. Ge and A. Zhang, "Pseudo cold start link prediction with multiple sources in social networks." SIAM.

[17] M. Al Hasan, V. Chaoji, S. Salem, and M. Zaki, "Link prediction using supervised learning," in *SDM06: Workshop on Link Analysis, Counter-terrorism and Security*, 2006.

[18] N. Z. Gong, A. Talwalkar, L. Mackey, L. Huang, E. C. R. Shin, E. Stefanov, E. R. Shi, and D. Song, "Joint link prediction and attribute inference using a social-attribute network," *ACM Transactions on Intelligent Systems and Technology (TIST)*, vol. 5, no. 2, p. 27, 2014.

[19] T. Murata and S. Moriyasu, "Link prediction of social networks based on weighted proximity measures," in *Web Intelligence, IEEE/WIC/ACM international conference on*. IEEE, 2007, pp. 85–88.

[20] R. Xiang, J. Neville, and M. Rogati, "Modeling relationship strength in online social networks," in *Proceedings of the 19th international conference on World wide web*. ACM, 2010, pp. 981–990.

[21] G. Palla, I. Derényi, I. Farkas, and T. Vicsek, "Uncovering the overlapping community structure of complex networks in nature and society," *Nature*, vol. 435, no. 7043, pp. 814–818, 2005.

[22] M. Gönen and E. Alpaydin, "Localized multiple kernel learning," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 352–359.

[23] C. Yang and T. Lozano-Perez, "Image database retrieval with multiple-instance learning techniques," in *Data Engineering, 2000. Proceedings. 16th International Conference on*. IEEE, 2000, pp. 233–243.

[24] S. Zander, T. Nguyen, and G. Armitage, "Automated traffic classification and application identification using machine learning," in *Local Computer Networks, 2005. 30th Anniversary. The IEEE Conference on*. IEEE, 2005, pp. 250–257.

[25] P. Richtárik and M. Takáč, "Iteration complexity of randomized block-coordinate descent methods for minimizing a composite function," *Mathematical Programming*, vol. 144, no. 1-2, pp. 1–38, 2014.

[26] L. Armijo *et al.*, "Minimization of functions having lipschitz continuous first partial derivatives," *Pacific Journal of mathematics*, vol. 16, no. 1, pp. 1–3, 1966.

[27] P. Tseng, "Convergence of a block coordinate descent method for nondifferentiable minimization," *Journal of optimization theory and applications*, vol. 109, no. 3, pp. 475–494, 2001.

[28] X. Wang, L. Tang, H. Gao, and H. Liu, "Discovering overlapping groups in social media," in *the 10th IEEE International Conference on Data Mining series (ICDM2010)*, Sydney, Australia, December 14 - 17 2010.

[29] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, and T. Eliassi-Rad, "Collective classification in network data," *AI magazine*, vol. 29, no. 3, p. 93, 2008.

[30] J. Tang, J. Zhang, L. Yao, J. Li, L. Zhang, and Z. Su, "Arnetminer: extraction and mining of academic social networks," in *Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2008, pp. 990–998.

[31] A. K. Menon and C. Elkan, "Link prediction via matrix factorization," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2011, pp. 437–452.

[32] N. Barbieri, F. Bonchi, and G. Manco, "Who to follow and why: link prediction with explanations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 1266–1275.

[33] L. Xu, A. Krzyżak, and E. Oja, "Rival penalized competitive learning for clustering analysis, rbf net, and curve detection," *Neural Networks, IEEE Transactions on*, vol. 4, no. 4, pp. 636–649, 1993.