Toward Personalized Relational Learning

Jundong Li* Liang Wu* Osmar R. Zaïane† Huan Liu*

Abstract

Relational learning exploits relationships among instances manifested in a network to improve the predictive performance of many network mining tasks. Due to its empirical success, it has been widely applied in myriad domains. In many cases, individuals in a network are highly idiosyncratic. They not only connect to each other with a composite of factors but also are often described by some content information of high dimensionality specific to each individual. For example in social media, as user interests are quite diverse and personal; posts by different users could differ significantly. Moreover, social content of users is often of high dimensionality which may negatively degrade the learning performance. Therefore, it would be more appealing to tailor the prediction for each individual while alleviating the issue related to the curse of dimensionality. In this paper, we study a novel problem of Personalized Relational Learning and propose a principled framework PRL to personalize the prediction for each individual in a network. Specifically, we perform personalized feature selection and employ a small subset of discriminative features customized for each individual and some common features shared by all to build a predictive model. On this account, the proposed personalized model is more human interpretable. Experiments on real-world datasets show the superiority of the proposed PRL framework over traditional relational learning methods.

1 Introduction

With the widespread availability and upsurge of various information systems, networks are becoming increasingly important to our day-to-day life; examples include social networks, communication networks, academic networks and financial transaction networks. Conventional data mining and machine learning techniques cannot be easily applied to networks as they assume data instances are independently and identically distributed (i.i.d.). In reality, instances in a network are explicitly or implicitly correlated, with complex dependencies [4, 27], making the enduring and deeply buried data i.i.d. assumption invalid. On top of that, in many cases, nodes in a network are associated with labels that characterize their behaviors or preferences. For example, the label information could indicate user interests or political polarizations in social media; research interests of scholars in academic networks; and functionalities of genes in protein-protein interaction (PPI) networks. Hence, inferring missing labels of nodes in a network could advance many real-world applications such as recommendation, personalized search and crowdsourcing [11, 32, 38, 33]. However, label information is rather limited on networks as the labeling process requires human attention and maybe very expensive; or itself is naturally unavailable due to some privacy issues. The limited access to label information necessities the usage of relational learning [31, 34, 35], which leverages the network structure that is readily available and a small subset of labeled nodes to assign unlabeled nodes to some predefined groups.

Most, if not all, individuals in a network are highly idiosyncratic. First, they show dependencies to each other due to a composite of complex reasons. As in the case of social networks, users build connections because they are relatives, colleagues, classmates or share some common interests. Second, nodes in many networks often have rich personalized accompanying attributes of high dimensionality. To give a palpable understanding, we can observe that in social media, content information (e.g., blogs, posts, images) by different users could be quite diverse and personal, with a variety of foci. Also, user-generated content is often high-dimensional with irrelevant, redundant and noisy features; it may jeopardize the prediction performance on unseen nodes due to the curse of dimensionality [17, 21]. Therefore, it is desired to tailor the prediction for each node in the network with only a small subset of relevant features. In other words, for each instance, we would like to use a subset of discriminative personalized features in conjunction with some shared features for prediction, while these personalized features could vary for different nodes. Consequently, the model is interpretable as we can explain why we make such a prediction.

In this paper, we study a novel problem of personalized relational learning on networks. This problem has not been previously studied, mainly because of the following challenges: (1) As per the fact that labeled nodes are scarce while network structure is readily observed, it is indispensable to design a relational model such that nodes could borrow strength from its neighbors in building a more accurate predictive model. (2) Social identity theory [28] suggests that individuals in

*Computer Science and Engineering, Arizona State University, Tempe, AZ, USA. {jundongl, wuliang, huan.liu}@asu.edu

†Department of Computing Science, University of Alberta, Edmonton, AB, Canada. zaiane@ualberta.ca

a network often exhibit different personalized patterns, but also, they more or less share some common behaviors to some extent. Relational learning should be able to seize these natures. (3) Traditional relational learning approaches often use a global pattern for the prediction purpose. Thus it is still not clear how to customize the prediction for each individual node. To tackle these challenges, we propose a novel relational learning framework PRL. The main contributions are as follows:

- We formally define the problem of personalized relational learning for networked data.
- We provide a principled relational model such that nodes in the network could borrow strength from its neighbors in fortifying the predictive model.
- We propose an effective way to capture common and personalized behaviors of individuals by selecting a small subset of discriminative features and a subset of shared features.
- We present an effective alternating algorithm to solve the optimization problem of the proposed PRL framework.
- We perform experiments on real-world datasets to corroborate the superiority of the PRL framework.

2 Problem Statement

We first summarize some notations used in this paper. Following commonly adopted notations, we use bold uppercase characters for matrices (e.g., $\mathbf{A}$), bold lowercase characters for vectors (e.g., $\mathbf{a}$). Also, we represent the i-th row of matrix $\mathbf{A} \in \mathbb{R}^{n\times d}$ as $\mathbf{A}_{i*}$, the j-th column as $\mathbf{A}_{*j}$, the (i,j)-th entry as $\mathbf{A}_{ij}$, transpose of matrix $\mathbf{A}$ as $\mathbf{A}^T$, its Frobenius norm is defined as $\|\mathbf{A}\|_F = \sqrt{\sum_{i=1}^{n}\sum_{j=1}^{d}\mathbf{A}_{ij}^2}$, its $\ell_{2,1}$-norm is $\|\mathbf{A}\|_{2,1} = \sum_{i=1}^{n}\sqrt{\sum_{j=1}^{d}\mathbf{A}_{ij}^2}$. The vectorization of a $n\times d$ matrix $\mathbf{A}$ is a column vector of size nd, denoted as $v(\mathbf{A}) = [\mathbf{A}_{11},...,\mathbf{A}_{n1},...,\mathbf{A}_{1d},...,\mathbf{A}_{nd}]^T$. $\mathbf{A}\otimes\mathbf{B}$ denotes the Kronecker product between matrices $\mathbf{A}$ and $\mathbf{B}$. $\mathbf{I}_d$ denotes the identity matrix of size $d\times d$.

Let $\mathcal{U} = \{u_1, u_2, ..., u_N\}$ be the set of N nodes in the network. Assume that $\mathcal{U}^L = \{u_1, u_2, ..., u_n\}$ is the set of n labeled nodes where $n < N$ and $\mathcal{U}^U = \mathcal{U}\backslash\mathcal{U}^L$ is the set of $N-n$ unlabeled nodes. We use the adjacency matrix $\mathbf{A}\in\mathbb{R}^{N\times N}$ to denote the dependencies among nodes in the network such that $\mathbf{A}_{ij} = 1$ if there is a link from node u_i to node u_j; otherwise $\mathbf{A}_{ij} = 0$. In addition, we use $\mathbf{X} = [\mathbf{x}_1;\mathbf{x}_2;...;\mathbf{x}_n] \in \mathbb{R}^{n\times d}$ to denote the feature representation these n labeled nodes, where each node is associated with a d-dimensional features. We also use $\mathcal{Y} = \{c_1, c_2, ..., c_k\}$ to denote label set of these nodes and $\mathbf{Y} = [\mathbf{y}_1;...;\mathbf{y}_n] \in \{0,1\}^{n\times k}$ is the corresponding one-hot label matrix for labeled nodes $\mathcal{U}^L$, where the j-th element in $\mathbf{y}_i$ is 1 if the i-th node is associated with class label c_j, otherwise 0;

With the aforementioned notations, we now formally define the problem of personalized relational learning as follows. Given a network $\mathcal{G}$ with network structure $\mathbf{A}$, feature representation $\mathbf{X}$ and labels $\mathbf{Y}$ for nodes in $\mathcal{U}^L$, the task is to train a classifier to predict the missing labels for nodes in $\mathcal{U}^U$. In particular, during the learning phase, we would like to (1) leverage both network structure and feature information; and (2) tailor the prediction for each node by employing a subset of features locally associated with the node itself and a small subset of features relevant to all nodes.

3 The Proposed Framework

In this section, we show how to build a personalized relation learning model in details. We first formulate the problem as an optimization problem and then present an effective optimization algorithm to solve it.

The workflow of the proposed framework is illustrated in Figure 1. From the figure, we can see that in the training phase, we have three sources of information, i.e., the network structure $\mathbf{A}$, feature matrix $\mathbf{X}$ and labels $\mathbf{Y}$ for nodes in $\mathcal{U}^L$. We first show how the proposed PRL framework finds some relevant features shared by all nodes (e.g., feature f_1) and also, a small subset of discriminative features that are locally associated with each specific node (e.g., features f_3 and f_5 for u_1) to build a personalized predictive model, i.e., a classifier. Second, as label information is rather limited in real-world networks, we show how PRL makes use of rich network structure to make nodes borrow strength from each other to improve the prediction performance. At last, we will provide an alternating optimization algorithm for the proposed PRL.

3.1 Modeling Node Features for Personalized Relational Learning

In order to infer the missing labels of unlabeled nodes, one simple and straightforward way is to build a global model for all nodes on the node features. However, one drawback is that it assumes that all nodes share the exactly same patterns. In other words, it conjectures that all nodes share the same feature weights, and the feature weights derived from labeled nodes could be directly shifted to unlabeled nodes. Despite the fact that nodes in a network share some common patterns to some extent, they are often regarded as being highly idiosyncratic, showing distinct behaviors. The idiosyncrasy of nodes has been heavily observed in reality and also is supported by social identity theory [8] in sociology. It motivates us to build a predictive model to capture both global and personalized behaviors of nodes in the network. Next, we first

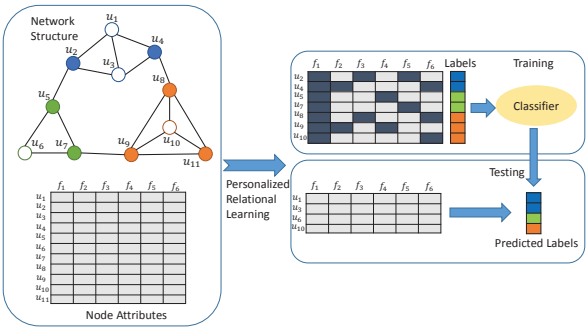


Figure 1: An illustration of the personalized relational learning framework (PRL). During the training phase, PRL leverages the network structure and content information of labeled nodes to build a classifier. For each labeled node, a subset of shared features and some unique personalized features are employed. For example, f_1 is a shared feature for all nodes, while for node u_1 and u_2, their personalized feature set are $\{f_3, f_5\}$ and $\{f_2, f_6\}$, respectively. Afterwards, we make use of the built classifier to make predictions on unlabeled nodes.

introduce the framework to model the common node patterns and then extend it to model the personalized nature of each individual.

Features of nodes in real-world networks are often high-dimensional with noisy, irrelevant and redundant information. Hence, the illusion that all node features are dovetailed with labels is not true. To uncover common behaviors shared by all nodes, we embed feature selection into a linear multi-class classification model, resulting in the following objective function:

$$\min_{\tilde{\mathbf{W}}} \sum_{i=1}^{n} \|\mathbf{x}_i \tilde{\mathbf{W}} - \mathbf{y}_i\|_2^2 + \gamma \|\tilde{\mathbf{W}}\|_{2,1}, \tag{3.1}$$

where $\tilde{\mathbf{W}} \in \mathbb{R}^{d \times k}$ is the global feature weight shared by all nodes, and the term $\gamma \|\tilde{\mathbf{W}}\|_{2,1}$ is imposed to achieve joint feature sparsity across k different classes.

To apprehend personalized behaviors of each single node, we also assume that each node is also associated with a local variable $\mathbf{W}^i$. In this way, the class labels of each node in $\mathcal{U}^L$ can be approximated by a conjunction of global model parameter $\tilde{\mathbf{W}}$ and a localized variable $\mathbf{W}^i$. In this way, Eq. (3.1) can be reformulated as:

$$\min_{\tilde{\mathbf{W}}, \mathbf{W}^i} \sum_{i=1}^{n} \|\mathbf{x}_i (\tilde{\mathbf{W}} + \mathbf{W}^i) - \mathbf{y}_i\|_2^2 + \gamma \|\tilde{\mathbf{W}}\|_{2,1}. \tag{3.2}$$

Similar to the modeling of global behaviors, personalized behavior is also encoded in a small subset of features that is locally associated with each individual. To put it in another way, we would like to achieve feature sparsity within each localized model parameter $\mathbf{W}^i$. It can be mathematically formulated by solving a exclusive group lasso problem [14, 15, 39]. In particular, each $\mathbf{W}^i$ is regarded as a group, exclusive group lasso encourages intra-group level competition but discourages inter-group level competition. As a result, a small subset of discriminative personalized features can be obtained within each $\mathbf{W}^i$. Therefore, we first impose a $\ell_{2,1}$-norm sparse regularization on $\mathbf{W}^i$ for intra-group level feature sparsity across k different class labels. Afterwards, we put ℓ_2 norm at the inter-group level for non-sparsity. With the intra-level sparsity and inter-level non-sparsity regularization term $\sum_{i=1}^{n} \|\mathbf{W}^i\|_{2,1}^2$, the node features $\mathbf{X}$ for personalized relational learning can be formally formulated as:

$$\min_{\tilde{\mathbf{W}}, \mathbf{W}^i} \sum_{i=1}^{n} \|\mathbf{x}_i (\tilde{\mathbf{W}} + \mathbf{W}^i) - \mathbf{y}_i\|_2^2 + \beta \sum_{i=1}^{n} \|\mathbf{W}^i\|_{2,1}^2 + \gamma \|\tilde{\mathbf{W}}\|_{2,1}, \tag{3.3}$$

where parameters β and γ are used to balance the sparsity of personalized and shared features, respectively.

3.2 Modeling Network Information for Personalized Relational Learning

The objective function in Eq. (3.3) builds a predictive learning model with the supervision of node labels $\mathbf{Y}$. However, as mentioned above that in many cases, the portion of labeled nodes is very limited, either because of the labor and time consuming labeling process or labels themselves are just unavailable due to some privacy issues. Fortunately, a rich source of network structure are readily observable and could be potentially helpful to build a more informative predictive model.

Even though individuals in a highly connected network exhibit some unique behaviors, as indicated by social categorization theory [29], these personalized individual behaviors is well organized and can be categorized into various groups. For example, groups can indicate different foci of user interests, such as sports, literature, and arts. Here the challenges center around inferring of personalized patterns and obtaining their group structures simultaneously. In this work, we take advantage of the network structure to cluster the personalized patterns based on the node connectivity. In particular, we force linked nodes to borrow strength from each other in learning personalized patterns to fortify the prediction model by the network lasso regularization term [7]:

$$\min_{\mathbf{W}} \sum_{i,j=1}^{n} \mathbf{A}_{i,j} \|\mathbf{W}^i - \mathbf{W}^j\|_F. \tag{3.4}$$

The advantages of the network lasso regularization term are two folds. First, the Frobenius norm of the difference between $\mathbf{W}^i$ and $\mathbf{W}^j$ not only makes them close

to each other if they are connected, i.e., $\mathbf{A}_{ij} = 1$, but also incentivizes them to be the same. In this way, since many localized feature weights $\mathbf{W}^i$ are made to be the same, they are automatically grouped into several clusters. Second, when the label information cannot provide us enough guide to learn the localized parameter, Eq. (3.4) provides us a way to borrow strength from neighbors for the model parameter learning.

3.3 Personalized Relational Learning By combing the objective function in Eq. (3.3) and Eq. (3.4), the final objective function of the proposed Personalized Relational Learning (PRL) model can be formulated as:

$$\begin{aligned}(3.5)\quad &\min_{\tilde{\mathbf{W}},\mathbf{W}^i} J(\tilde{\mathbf{W}},\mathbf{W}^i) = \sum_{i=1}^{n} \|\mathbf{x}_i(\tilde{\mathbf{W}}+\mathbf{W}^i) - \mathbf{y}_i\|_2^2 \\ &+\alpha \sum_{i,j=1}^{n} \mathbf{A}(i,j)\|\mathbf{W}^i - \mathbf{W}^j\|_F + \beta \sum_{i=1}^{n} \|\mathbf{W}^i\|_{2,1}^2 + \gamma\|\tilde{\mathbf{W}}\|_{2,1},\end{aligned}$$

where α is a model parameter to control the contribution of network structure in helping personalized relational learning. Also, it controls how the nodes are clustered according to their localized feature parameters $\mathbf{W}^i$. By solving the above optimization problem, we can obtain $\tilde{\mathbf{W}}$ that captures the global feature pattern and a set of $\mathbf{W}^i$ $(i = 1, ..., n)$ that capture the personalized feature pattern for each node in $\mathcal{U}^L$.

Now we discuss how to make a prediction for unlabeled nodes by the built classifier which is a conjunction of $\tilde{\mathbf{W}}$ and $\mathbf{W}^i$. During the prediction phase, we first find the linked neighbors for a new unlabeled node u_t in the network $\mathcal{G}$; then if we successfully find some neighbors, we take the averaged feature parameters (conjunction of global and personalized) of its neighbors as the new feature weight $\overline{\mathbf{W}}^t$; otherwise, we use the averaged feature parameters (conjunction of global and personalized) of all nodes in $\mathcal{U}^L$ as the new feature weight $\overline{\mathbf{W}}^t$. After we obtain the feature weight for the new unlabeled node u_t, its class labels can be predicted by $c^* = \arg\max_{c_j \in \mathcal{C}}(|\mathbf{x}_t\overline{\mathbf{W}}^t|_j)$.

3.4 Optimization Solution The objective function of PRL in Eq. (3.5) involves two sets of variables: (1) the global variable $\tilde{\mathbf{W}}$ that captures the global patterns of nodes in the network; and (2) the localized variable $\mathbf{W}^i$ that encodes personalized behaviors of each individual node. The objective function is not convex w.r.t. $\tilde{\mathbf{W}}$ and $\mathbf{W}^i$ $(i = 1, ..., n)$ simultaneously. In addition to that, the objective function is also not smooth. Motivated by [36], we present an effective alternating algorithm to solve it, thus in each iteration, the model parameters could be updated with a closed-form solution.

First, we denote $\mathbf{Z} \in \mathbb{R}^{n \times nd}$ as follow:

$$\mathbf{Z} = \begin{bmatrix} \mathbf{X}_{11} & & & \dots & \mathbf{X}_{1d} & & \\ & \ddots & & \dots & & \ddots & \\ & & \mathbf{X}_{n1} & \dots & & & \mathbf{X}_{nd} \end{bmatrix}.$$

Also, we represent the set of all n localized variables $\mathbf{W}^i$ in a concatenated format as $\mathbf{W} = [\mathbf{W}^1; \mathbf{W}^2; ...; \mathbf{W}^n]$. In this regard, we only two blocks of variables, i.e., $\tilde{\mathbf{W}}$ and $\mathbf{W}$ for the objective function in Eq. (3.5). We iteratively update these two blocks of variables as follows:

3.4.1 Update $\tilde{\mathbf{W}}$ When $\mathbf{W}$ is fixed, we remove the terms that are irrelevant to $\tilde{\mathbf{W}}$ and obtain the following optimization problem:

$$(3.6)\qquad \min_{\tilde{\mathbf{W}}} J(\tilde{\mathbf{W}}) = \|\mathbf{X}\tilde{\mathbf{W}} + \mathbf{Z}\mathbf{W} - \mathbf{Y}\|_F^2 + \gamma\|\tilde{\mathbf{W}}\|_{2,1}.$$

Since the objective function in Eq. (3.6) is convex, we take the derivative of $J(\tilde{\mathbf{W}})$ w.r.t. $\tilde{\mathbf{W}}$ to be zero, then we have the following:

$$(3.7)\qquad \mathbf{X}^T(\mathbf{X}\tilde{\mathbf{W}} + \mathbf{Z}\mathbf{W} - \mathbf{Y}) + \gamma\mathbf{D}\tilde{\mathbf{W}} = 0,$$

where $\mathbf{D} \in \mathbb{R}^{d \times d}$ is a diagonal matrix with its i-th diagonal element formulated as[1]:

$$(3.8)\qquad \mathbf{D}_{ii} = \frac{1}{2\|\tilde{\mathbf{W}}_{i*}\|_2}.$$

$\mathbf{X}^T\mathbf{X}+\gamma\mathbf{D}$ is positive semidefinite since $\mathbf{X}^T\mathbf{X}$ is positive semidefinite and $\gamma\mathbf{D}$ is a positive diagonal matrix. As a result, we obtain a closed-form solution of $\tilde{\mathbf{W}}$ as:

$$(3.9)\qquad \tilde{\mathbf{W}} = -(\mathbf{X}^T\mathbf{X} + \gamma\mathbf{D})^{-1}\mathbf{X}^T(\mathbf{Z}\mathbf{W} - \mathbf{Y})$$

3.4.2 Update $\mathbf{W}$ Next, we show how to update $\mathbf{W}$ when $\tilde{\mathbf{W}}$ is fixed. Similarly, by removing the terms that are not relevant to $\mathbf{W}$, we obtain the following optimization problem:

$$\begin{aligned}(3.10)\quad &\min_{\mathbf{W}} J(\mathbf{W}) = \|\mathbf{Z}\mathbf{W} + \mathbf{X}\tilde{\mathbf{W}} - \mathbf{Y}\|_F^2 \\ &+\alpha \sum_{i,j=1}^{n} \mathbf{A}_{ij}\|\mathbf{W}^i - \mathbf{W}^j\|_F + \beta \sum_{i=1}^{n} \|\mathbf{W}^i\|_{2,1}^2.\end{aligned}$$

We can reformulate the first term in Eq. (3.10) as:

$$(3.11)\qquad J_1(\mathbf{W}) = \|\mathbf{Z}\mathbf{W} + \mathbf{X}\tilde{\mathbf{W}} - \mathbf{Y}\|_F^2.$$

The derivative of $J_1(\mathbf{W})$ w.r.t. $\mathbf{W}$ is as follows:

$$(3.12)\qquad \frac{\partial J_1(\mathbf{W})}{\partial \mathbf{W}} = 2\mathbf{Z}^T(\mathbf{Z}\mathbf{W} + \mathbf{X}\tilde{\mathbf{W}} - \mathbf{Y}).$$

[1] It should be noted that in practice, $\|\tilde{\mathbf{W}}_{i,:}\|_2$ could be very close to zero. Hence, we define $\mathbf{D}_{ii} = \frac{1}{2\|\tilde{\mathbf{W}}_{i,:}\|_2 + \epsilon}$, where ϵ is a very small constant.

The second term $J_2(\mathbf{W})$ in Eq. (3.10) is the network lasso penalty term. Its derivative w.r.t. $\mathbf{W}^k$, i.e., $\frac{\partial J_2(\mathbf{W})}{\partial \mathbf{W}^k}$ can be represented as follows:

$$\begin{aligned}(3.13)\quad &\sum_{i=1}^{n} \frac{\mathbf{A}_{ik}(\mathbf{W}^k - \mathbf{W}^i)}{\|\mathbf{W}^i - \mathbf{W}^k\|_F} + \sum_{j=1}^{n} \frac{\mathbf{A}_{kj}(\mathbf{W}^k - \mathbf{W}^j)}{\|\mathbf{W}^k - \mathbf{W}^j\|_F} \\ =&\mathbf{W}^k \sum_{i=1}^{n} \frac{2\mathbf{A}_{ik}}{\|\mathbf{W}^i - \mathbf{W}^k\|_F} - \sum_{i=1}^{n} \frac{2\mathbf{A}_{ik}\mathbf{W}_i}{\|\mathbf{W}^i - \mathbf{W}^k\|_F}.\end{aligned}$$

By concatenating all $\frac{\partial J_2(\mathbf{W})}{\partial \mathbf{W}^k}$ $(k = 1, ..., n)$ together, we can get the representation of $\frac{\partial J_2(\mathbf{W})}{\partial \mathbf{W}}$ as follows[2]:

$$(3.14)\quad \frac{\partial J_2(\mathbf{W})}{\partial \mathbf{W}} = 2(\mathbf{C} \otimes \mathbf{I}_d)\mathbf{W},$$

where the matrix $\mathbf{C} \in \mathbb{R}^{n \times n}$ is as follows:

(3.15)

$$\mathbf{C}_{ij} = \begin{cases} \sum_{l=1}^{n} \frac{\mathbf{A}_{il}}{\|v(\mathbf{W}^i) - v(\mathbf{W}^l)\|_2} - \frac{\mathbf{A}_{ij}}{\|v(\mathbf{W}^i) - v(\mathbf{W}^j)\|_2} & (i = j) \\ \frac{-\mathbf{A}_{ij}}{\|v(\mathbf{W}^i) - v(\mathbf{W}^j)\|_2} & (i \neq j). \end{cases}$$

The third term $\sum_{i=1}^{n} \|\mathbf{W}^i\|_{2,1}^2$ is the exclusive group lasso penalty term that enables personalized feature selection among each node in the network. Its derivative w.r.t. $\mathbf{W}$ can be represented as follows:

$$(3.16)\quad \frac{J_3(\mathbf{W})}{\partial v(\mathbf{W})} = 2\mathbf{F}\mathbf{W},$$

where $\mathbf{F} \in \mathbb{R}^{nd \times nd}$ is a diagonal matrix with $\mathbf{F}_{11}, \mathbf{F}_{22}, ..., \mathbf{F}_{nd,nd}$ on the diagonal. The diagonal element $\mathbf{F}_{mm}$ in $\mathbf{F}$ can be represented as[2]:

$$(3.17)\quad \begin{aligned}&\mathbf{F}_{mm} = \sum_{i=1}^{n} \frac{\mathbb{I}_{m,i}\|\mathbf{W}^i\|_{2,1}}{\|\mathbf{V}_{m,:}\|_2}, \\ &\mathbf{V} = [\mathbf{W}^1; \mathbf{W}^2, ..., \mathbf{W}^n],\end{aligned}$$

where $\mathbb{I}_{m,i}$ is an indicator function such that $\mathbb{I}_{m,i}$ is 1 if $\mathbf{V}_{m,:}$ belongs to $\mathbf{W}^i$; otherwise $\mathbb{I}_{m,i}$ is 0.

Putting the derivative of $J_1(\mathbf{W})$, $J_2(\mathbf{W})$ and $J_3(\mathbf{W})$ w.r.t. $\mathbf{W}$ together, we can obtain the closed-form solution of $v(\mathbf{W})$ as follows:

(3.18)

$$\mathbf{W} = -(\mathbf{Z}^T\mathbf{Z} + \alpha(\mathbf{C} \otimes \mathbf{I}_d) + \beta\mathbf{F})^{-1}\mathbf{Z}^T(\mathbf{X}\tilde{\mathbf{W}} - \mathbf{Y}).$$

The pseudo code of obtaining the optimal solution of $\tilde{\mathbf{W}}$ and $\mathbf{W}$ can be summarized in Algorithm 1.

[2]The derivative of $J_2(\mathbf{W})$ and $J_3(\mathbf{W})$ w.r.t. $\mathbf{W}$ are not smooth at some certain points. To make them derivative at any points, we approximate them $\mathbf{C}$ and $\mathbf{F}$ as

$$\mathbf{C}_{ij} = \begin{cases} \sum_{l=1}^{n} \frac{\mathbf{A}_{il}}{\|v(\mathbf{W}^i) - v(\mathbf{W}^l) + \epsilon\|_2} - \frac{\mathbf{A}_{ij}}{\|v(\mathbf{W}^i) - v(\mathbf{W}^j) + \epsilon\|_2} & (i = j) \\ \frac{-\mathbf{A}_{ij}}{\|v(\mathbf{W}^i) - v(\mathbf{W}^j) + \epsilon\|_2} & (i \neq j), \end{cases}$$

and $\mathbf{F}_{mm} = \sum_{i=1}^{n} \frac{\mathbb{I}_{m,i}\|\mathbf{W}^i\|_{2,1}}{\|\mathbf{V}_{m,:} + \epsilon\|_2}$, where ϵ is a very small constant.

Algorithm 1 Optimization algorithm for the proposed Personalized Relational Learning (PRL)

Input: $\mathbf{X} \in \mathbb{R}^{n \times d}$, $\mathbf{Z} \in \mathbb{R}^{n \times nd}$, $\mathbf{Y} \in \mathbb{R}^{n \times k}$, α, β, γ.
Output: $\tilde{\mathbf{W}} \in \mathbb{R}^{d \times k}$, $\mathbf{W}^i \in \mathbb{R}^{d \times k}$ $(i = 1, ..., n)$.

```
Initialize W̃, W^i (i = 1, ..., n);
while objective function in Eq. (3.5) not converge do
    Update D by Eq. (3.8);
    Update W̃ by Eq. (3.9);
    Update C as Eq. (3.15);
    Update F as Eq. (3.17);
    Update W by Eq. (3.18);
    Obtain W^i (i = 1, .., n) from W;
end while
```

4 Experimental Results

In this section, we conduct experiments to evaluate the effectiveness of the proposed personalized relational learning framework (PRL). We first introduce the used datasets, compared methods, and experimental settings before presenting detailed results of the experiments. At last, we investigate the parameter sensitivity of the proposed PRL framework.

4.1 Datasets We use three real-world networks for evaluation, and all of them are publicly available. Cora and Citeseer are real-world academic networks[3] while BlogCatalog is a social media network[4].

Cora: The Cora dataset is a citation network with 2708 publications and 5429 citations. Each publication is described by a set of 1433 words which are considered as features. All these features are 0/1-valued. All publications are categorized into 7 classes according to their subjects.

Citeseer: The Citeseer dataset is another citation network with 3312 publications and 4732 links. They are grouped into 6 classes. Similar to Cora, each publication is associated with a total of 3703 0/1-valued features.

BlogCatalog: The BlogCatalog dataset is a social blogging dataset with 5196 users. The tag information of blogs by users are regarded as features; the feature number is 1,638. A total number of 171,743 links are observed. The ground truth is the major category (among 6 categories) of blogs posted by users.

4.2 Comparison Methods We select several representative relational learning methods for a fair comparison. Among them, NMF only considers node features while wvRN and SocDim only exploit network structure. On the other hand, GNMF, FsNet, and the proposed PRL framework can be regarded as approaches

[3]http://linqs.umiacs.umd.edu/projects/projects/lbc/
[4]http://dmml.asu.edu/users/xufei/datasets.html

that make use of both sources of information.

- **NMF**: Non-negative Matrix Factorization (NMF) [16] has proven to be effective in many real-world applications by reducing the feature dimensionality. We consider it as a baseline method to first obtain the low-rank node feature representation and then apply discriminative learning methods.
- **wvRN**: Weighted-Vote Relational Neighbor Classifier (wvRN) [22] is a local neighborhood based classifier. It makes the prediction for unlabeled nodes by a weighted vote score of its labeled neighbors.
- **SocDim**: Social Dimensions (SocDim) [31] is one of the state-of-the-art relational learning approaches with only network information. It first adopts modularity maximization [25] to extract latent representations and then utilize them as features for discriminative learning.
- **GNMF**: Graph Regularized NMF (GNMF) [2] is based on the assumption that latent representations of connected nodes are also similar to each other. After getting the low-rank feature representation, we take them as input to a typical learning method.
- **FsNet**: FsNet [6] aims to select a subset of relevant features on the node feature space. In particular, it exploits a linear regression model to capture the node features and adopt graph regularization to make use of the network structure. We employ discriminative learning methods to build a predictive model based on the selected features.

4.3 Experimental Settings The vast majority of relational learning methods heavily depend on the extracted feature representations. Among these comparison methods, NMF, SocDim, GNMF, and FsNet are typical feature-based relational learning methods. They first extract latent features and then employ typical discriminative methods to build a classifier to enable the prediction on unlabeled nodes. In the experiments, we follow a commonly adopted setting [31] to use linear SVM for discriminative learning.

For each method, we randomly choose $p\%$ of nodes for training and the rest $1-p\%$ for testing. As we often have limited access to labeled nodes in practice, we choose a relatively small value for p by varying it in the range of $\{1, 2, ..., 9, 10\}$. For each p, we run the experiments 10 times and report the average classification performance. Two widely used evaluation criteria based on F1-measure, i.e., Micro-F1 and Macro-F1 are used to measure the multi-class and multi-label classification problems [13].

4.4 Classification Performance Comparison In this subsection, we evaluate the performance of the proposed PRL relational learning framework by comparing its classification performance with other methods on the three above mentioned datasets. The comparison results are shown in Table 1, Table 2 and Table 3. The model parameters of the proposed PRL framework could be determined by cross-validation, and a detailed sensitivity study will be investigated in the following subsection. We make the following observations from these three tables:

- In most cases, when we gradually increase the number of labeled nodes from 1% to 10%, the classification performance increases for all methods in the table.
- Our proposed personalized relational learning framework PRL outperforms all baseline methods in almost all cases. We also perform a pairwise Wilcoxon signed-rank test [5] between PRL and these baseline methods, the comparison results show that PRL is significantly better, with a significance level in both 0.01 and 0.05.
- Both wvRN and SocDim are relational learning methods with only network information; their classification performance is inferior to relational learning approaches incorporating node features such as GNMF, FsNet, and PRL. The results support the importance of leveraging both sources of information for relational learning.
- GNMF is an extension of NMF that uses graph regularization to make the latent representation consistent with the network topological structure. It obtains higher Micro-F1 and Macro-F1 than NMF in most cases, suggesting that exploration of rich network information is helpful and could improve relational learning.
- FsNet selects a common set of relevant features, while our proposed method could be regarded as a personalized feature selection framework. The improvement of PRL over FsNet validates the necessity of employing personalized features for relational learning, which has an added value over a set of shared features.

4.5 Parameter Study PRL has three important parameters. The first parameter α balances the contribution of node features and network structure for relational learning. The second parameter β and the third parameter γ controls the sparsity of personalized features of each individual and the common feature in the

Training Ratio		1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Micro-F1	NMF	0.3914	0.4531	0.4748	0.4980	0.5229	0.5342	0.5402	0.5460	0.5639	0.5494
	wvRN	0.3230	0.3402	0.3626	0.3751	0.3929	0.4109	0.4221	0.4399	0.4502	0.4643
	SocDim	0.3322	0.3942	0.4414	0.4797	0.4996	0.5315	0.5467	0.5636	0.5872	0.5945
	GNMF	0.3936	0.4510	0.4798	0.5137	0.5216	0.5415	0.5477	0.5586	0.5726	0.5740
	FsNet	0.3880	0.4516	0.4829	0.5079	0.5231	0.5274	0.5384	0.5413	0.5444	0.5399
	PRL	**0.4254**	**0.4908**	**0.5324**	**0.5506**	**0.5688**	**0.5811**	**0.5989**	**0.6170**	**0.6266**	**0.6315**
Macro-F1	NMF	0.3133	0.3874	0.4178	0.4409	0.4829	0.4960	0.5041	0.5053	0.5262	0.5038
	wvRN	0.1198	0.1617	0.2064	0.2374	0.2721	0.3045	0.3273	0.3556	0.3755	0.3979
	SocDim	0.3077	0.3808	0.4256	0.4628	0.4814	0.5123	0.5311	0.5469	0.5688	0.5769
	GNMF	0.3173	0.3906	0.4300	0.4674	0.4793	0.4999	0.5061	0.5212	0.5340	0.5404
	FsNet	0.3074	0.3905	0.4269	0.4626	0.4836	0.4892	0.5040	0.5074	0.5133	0.5109
	PRL	**0.3833**	**0.4098**	**0.4881**	**0.4968**	**0.5324**	**0.5539**	**0.5637**	**0.5791**	**0.5906**	**0.6039**

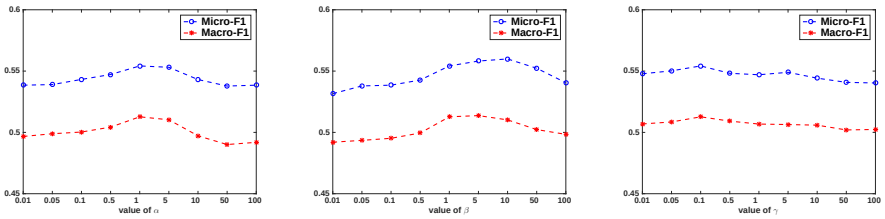
Table 1: Classification performance comparison on Cora dataset with different portions of training data.

Training Ratio		1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Micro-F1	NMF	0.4236	0.4704	0.4749	0.4926	0.4978	0.5062	0.5264	0.5329	0.5363	0.5412
	wvRN	0.2264	0.2412	0.2548	0.2655	0.2779	0.2884	0.3003	0.3118	0.3206	0.3313
	SocDim	0.2701	0.2996	0.3254	0.3447	0.3523	0.3682	0.3750	0.3855	0.3934	0.4044
	GNMF	0.4296	0.4768	0.4981	0.5124	0.5235	0.5243	0.5253	0.5357	0.5435	0.5535
	FsNet	0.4301	0.4657	0.5125	0.5142	0.5202	0.5301	0.5344	0.5417	0.5524	0.5576
	PRL	**0.4356**	**0.4851**	**0.5296**	**0.5307**	**0.5505**	**0.5535**	**0.5568**	**0.5691**	**0.5725**	**0.5762**
Macro-F1	NMF	0.3732	0.4271	0.4347	0.4548	0.4589	0.4671	0.4881	0.4961	0.4977	0.5021
	wvRN	0.0887	0.1172	0.1421	0.1626	0.1843	0.2023	0.2221	0.2393	0.2532	0.2700
	SocDim	0.2453	0.2815	0.3056	0.3264	0.3333	0.3476	0.3544	0.3644	0.3712	0.3821
	GNMF	0.3820	0.4346	0.4565	0.4723	0.4837	0.4862	0.4865	0.4967	0.5061	0.5141
	FsNet	0.3677	0.4183	0.4683	0.4714	0.4797	0.4835	0.4949	0.5030	0.5089	0.5167
	PRL	**0.3993**	**0.4356**	**0.4751**	**0.4862**	**0.5103**	**0.5142**	**0.5220**	**0.5231**	**0.5287**	**0.5295**

Table 2: Classification performance comparison on Citeseer dataset with different portions of training data.

Training Ratio		1%	2%	3%	4%	5%	6%	7%	8%	9%	10%
Micro-F1	NMF	0.5342	0.5938	0.6235	0.6531	0.6570	0.6617	0.6739	0.6787	0.6854	0.6938
	wvRN	0.2516	0.3181	0.3511	0.3928	0.4204	0.4372	0.4598	0.4727	0.4849	0.5026
	SocDim	0.3697	0.4419	0.4741	0.5078	0.5340	0.5502	0.5680	0.5831	0.5947	0.5952
	GNMF	0.5632	0.6063	**0.6504**	0.6587	0.6634	0.6743	0.6771	0.6873	0.6880	0.6927
	FsNet	0.5363	0.6096	0.6240	0.6308	0.6467	0.6359	0.6422	0.6444	0.6408	0.6433
	PRL	**0.6009**	**0.6127**	0.6341	**0.6622**	**0.6767**	**0.6939**	**0.7117**	**0.7184**	**0.7235**	**0.7365**
Macro-F1	NMF	0.5279	0.5856	0.6184	0.6479	0.6529	0.6579	0.6693	0.6748	0.6804	0.6885
	wvRN	0.2276	0.3043	0.3416	0.3836	0.4123	0.4299	0.4495	0.4607	0.4722	0.4902
	SocDim	0.3651	0.4372	0.4690	0.5023	0.5293	0.5429	0.5599	0.5754	0.5863	0.5869
	GNMF	0.5533	0.6006	0.6236	0.6544	0.6571	0.6689	0.6733	0.6819	0.6925	0.6963
	FsNet	0.5189	0.6010	0.6175	0.6306	0.6452	0.6338	0.6417	0.6436	0.6398	0.6426
	PRL	**0.5720**	**0.6153**	**0.6447**	**0.6697**	**0.6661**	**0.6923**	**0.7143**	**0.7153**	**0.7228**	**0.7335**

Table 3: Classification performance comparison on Blogcatalog dataset with different portions of training data.



(a) Effect of parameter α. (b) Effect of parameter β. (c) Effect of parameter γ.

Figure 2: Parameter study on Citeseer dataset.

model learning phase. To investigate the effects of these three parameters, we fix one parameter each time and vary the other two to see how it affects the classification performance. We only show the parameter study results on Citeseer dataset to save space as we have the similar observations on the other two datasets. The portion of

training data in the study is set to be 5%. First, we fix the parameters β as 1 and γ as 0.1 and vary the value of α among $\{0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100\}$. As can be observed from Figure 2(a), when we gradually increase α, the classification performance first increases and reaches its peak and then gradually decreases. The best performance is achieved when α is between 1 and 5, which is consistent with the suggestions from [7, 36]. Next, we fix $\alpha = 1$ and $\gamma = 0.1$ and vary β as $\{0.01, 0.05, 0.1, 0.5, 1, 5, 10, 50, 100\}$. The study results are shown in Figure 2(b). We observe that when β is small, the classification performance is relatively lower. The reason is that when β is small, the contribution of the personalized feature selection is limited; on the other hand, a large β enables us to find better localized features customized for each node, which in turn benefit the prediction performance. At last, we fix $\alpha = 1$ and $\beta = 1$, and vary the third variable γ. In particular, when γ is small, personalized features will dominate the objective function, the performance is high. As we continuously increase γ, the performance gradually decreases, but the change is small. In a nutshell, PRL is not very sensitive to these three parameters, and we can tune them in a wide range in practice.

5 Related Work

In this section, we review some related work from two perspectives: (1) traditional relational learning on plain networks; and (2) relational learning with node features.

5.1 Traditional Relational Learning Relational learning refers to the problem of learning on relational data that can be naturally represented as a network. Due to the prevalence of networks in many high impact domains, relational learning has received considerable attention in the past decade. Most of existing attempts are mainly based on the Markov assumption that labels of connected nodes show autocorrelation. Collective inference [12, 24] is a typical example that first derives relational features from labeled nodes, and then iteratively predicts the labels for unlabeled nodes. To enable the prediction on each unlabeled node, Weighted-vote relational neighbor (wvRN) [22] adopts a weighted voting score of the class probabilities from its labeled neighbors. Above mentioned approaches focus on one hub autocorrelation on networks and are therefore limited. Label propagation [40], graph regularization [37], spectral partitioning [23], and graphical model [35] based methods are developed to take advantage of long-distance autocorrelation for the prediction task. As connections in social networks are often multi-dimensional, Tang and Liu [31] extract latent social dimensions from the network first, with each dimension depicting a plausible affiliation among social actors. Then they resort to social dimensions and fed them into a typical discriminative learning approach such as SVM. Another category of methods that is related to relational learning is network embedding [9, 10, 26], which aims to learn vector representation of nodes that can well preserve the network topological structure. Discriminative learning approaches are employed afterwards on the learned embeddings. Our proposed PRL framework differs from these approaches as they overwhelmingly focus on plain networks while our framework tackles a more complex scenario with node features.

5.2 Relational Learning with Node Features As node features are naturally observed in many real-world networks, there is a surge of research on relational learning with node features. It is an interesting yet challenging topic, mainly because of the bewildering combination of heterogeneous contents and structures. Since node features are often noisy and of high dimensionality, vast of existing methods try to learn a low-rank representation on node attributes for relational learning. GNMF [2] take advantages of NMF to reduce feature dimensionality and then employs graph regularization to capture the network structure. Wu et al. [34] investigate whether social status of nodes in the network could be a complement to network structure and node features to improve the relational learning performance. In the proposed RESA framework, they also exploit the NMF model to reduce the feature dimensionality of node features. Another line of work is feature selection on networked data [6, 18, 19, 20, 30]. However, these work are also distinct from the proposed PRL framework as we attempt to tailor the prediction for each node in the network by finding a set of personalized features while these methods perform relational learning with the same set of feature representations.

6 Conclusions and Future Work

In addition to the readily observed network information, nodes in many real-world networks are often described by a rich set of features of high dimensionality. Recent studies show that the exploration of node features could advance a variety of learning tasks. Relational learning is one among these learning tasks. It targets to use network structure and node features of a small number of labeled nodes to build a predictive learning model; then employ the built model to infer missing labels for unlabeled nodes. Existing methods on this line assume that all nodes have a common pattern by sharing the same feature weight. However, as nodes in networks are highly idiosyncratic, their associated node features are quite diverse and personalized. Hence, it would

be appealing to tailor the prediction by using a set of personalized features tightly hinged with the node, and a set of common features shared by all nodes. Toward this goal, we propose a novel personalized relational learning framework PRL. As we can customize the prediction for each individual, the proposed model is also human interpretable. Experiments on real-world networks show the effectiveness of the proposed model.

Future work can be focused on two aspects. First, we would like to investigate if we can further employ the personalized model to other network mining tasks like social recommendation, community detection, and link prediction. Second, real-world networks are naturally dynamic with network structure changes and content drifts [1, 3, 18]. Therefore, we will study how to make the proposed personalized relational learning framework to handle dynamic networks.

Acknowledgements

This material is, in part, supported by National Science Foundation (NSF) under grant number 1614576.

References

[1] C. Aggarwal and K. Subbian. Evolutionary network analysis: A survey. *CSUR*, 47(1):10, 2014.
[2] D. Cai, X. He, J. Han, and T. S. Huang. Graph regularized nonnegative matrix factorization for data representation. *TPAMI*, 33(8):1548–1560, 2011.
[3] C. Chen and H. Tong. Fast eigen-functions tracking on dynamic graphs. In *SDM*, 2015.
[4] C. Chen, H. Tong, L. Xie, L. Ying, and Q. He. Fascinate: Fast cross-layer dependency inference on multi-layered networks. In *KDD*, 2016.
[5] J. Demšar. Statistical comparisons of classifiers over multiple data sets. *JMLR*, 7:1–30, 2006.
[6] Q. Gu and J. Han. Towards feature selection in network. In *CIKM*, 2011.
[7] D. Hallac, J. Leskovec, and S. Boyd. Network lasso: Clustering and optimization in large graphs. In *KDD*, 2015.
[8] M. J. Hornsey. Social identity theory and self-categorization theory: A historical review. *Social and Personality Psychology Compass*, 2(1):204–222, 2008.
[9] X. Huang, J. Li, and X. Hu. Accelerated attributed network embedding. In *SDM*, 2017.
[10] X. Huang, J. Li, and X. Hu. Label informed attributed network embedding. In *WSDM*, 2017.
[11] G. Jeh and J. Widom. Scaling personalized web search. In *WWW*, 2003.
[12] D. Jensen, J. Neville, and B. Gallagher. Why collective inference improves relational classification. In *KDD*, 2004.
[13] L. Jian, J. Li, K. Shu, and H. Liu. Multi-label informed feature selection. In *IJCAI*, 2016.
[14] D. Kong, R. Fujimaki, J. Liu, F. Nie, and C. Ding. Exclusive feature learning on arbitrary structures via $\ell_{1,2}$ -norm. In *NIPS*, 2014.
[15] D. Kong, J. Liu, B. Liu, and X. Bao. Uncorrelated group lasso. In *AAAI*, 2016.
[16] D. D. Lee and H. S. Seung. Algorithms for non-negative matrix factorization. In *NIPS*, 2001.
[17] J. Li, K. Cheng, S. Wang, F. Morstatter, T. Robert, J. Tang, and H. Liu. Feature selection: A data perspective. *arXiv:1601.07996*, 2016.
[18] J. Li, X. Hu, L. Jian, and H. Liu. Toward time-evolving feature selection on dynamic networks. In *ICDM*, 2016.
[19] J. Li, X. Hu, J. Tang, and H. Liu. Unsupervised streaming feature selection in social media. In *CIKM*, 2015.
[20] J. Li, X. Hu, L. Wu, and H. Liu. Robust unsupervised feature selection on networked data. In *SDM*, 2016.
[21] J. Li and H. Liu. Challenges of feature selection for big data analytics. *arXiv preprint arXiv:1611.01875*, 2016.
[22] S. A. Macskassy and F. Provost. A simple relational classifier. Technical report, DTIC Document, 2003.
[23] F. McSherry. Spectral partitioning of random graphs. In *FOCS*, 2001.
[24] J. Neville and D. Jensen. Collective classification with relational dependency networks. In *MRDM*, 2003.
[25] M. E. Newman. Modularity and community structure in networks. *PNAS*, 103(23):8577–8582, 2006.
[26] B. Perozzi, R. Al-Rfou, and S. Skiena. Deepwalk: Online learning of social representations. In *KDD*, 2014.
[27] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. 2011.
[28] H. Tajfel. *Social identity and intergroup relations*. 2010.
[29] H. Tajfel, M. G. Billig, R. P. Bundy, and C. Flament. Social categorization and intergroup behaviour. *European journal of social psychology*, 1(2):149–178, 1971.
[30] J. Tang and H. Liu. Feature selection with linked data in social media. In *SDM*, 2012.
[31] L. Tang and H. Liu. Relational learning via latent social dimensions. In *KDD*, 2009.
[32] D. Wang, D. Pedreschi, C. Song, F. Giannotti, and A.-L. Barabasi. Human mobility, social ties, and link prediction. In *KDD*, 2011.
[33] X. Wang, W. Lu, M. Ester, C. Wang, and C. Chen. Social recommendation with strong and weak ties. In *CIKM*, 2016.
[34] L. Wu, X. Hu, and H. Liu. Relational learning with social status analysis. In *WSDM*, 2016.
[35] Z. Xu, V. Tresp, S. Yu, and K. Yu. Nonparametric relational learning for social network analysis. In *SNA-KDD*, 2008.
[36] M. Yamada, K. Takeuchi, T. Iwata, J. Shawe-Taylor, and S. Kaski. Sparse network lasso for local high-dimensional regression. *arXiv preprint arXiv:1603.06743*, 2016.
[37] D. Zhou, J. Huang, and B. Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *ICML*, 2005.
[38] Y. Zhou and J. He. Crowdsourcing via tensor augmentation and completion. In *IJCAI*, 2016.
[39] Y. Zhou, R. Jin, and S. C. Hoi. Exclusive lasso for multi-task feature selection. In *AISTATS*, 2010.
[40] X. Zhu, Z. Ghahramani, J. Lafferty, et al. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, 2003.