Graph Regularized Meta-path Based Transductive Regression in Heterogeneous Information Network

Mengting Wan* Yunbo Ouyang* Lance Kaplan† Jiawei Han*

Abstract

A number of real-world networks are heterogeneous information networks, which are composed of different types of nodes and links. Numerical prediction in heterogeneous information networks is a challenging but significant area because network based information for unlabeled objects is usually limited to make precise estimations. In this paper, we consider a graph regularized meta-path based transductive regression model (*Grempt*), which combines the principal philosophies of typical graph-based transductive classification methods and transductive regression models designed for homogeneous networks. The computation of our method is time and space efficient and the precision of our model can be verified by numerical experiments.

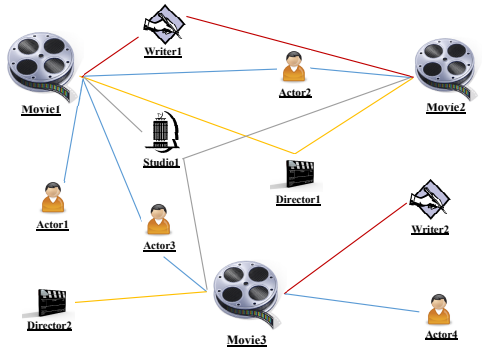


Figure 1: An example of heterogeneous information network: IMDb network composed of movies, directors, writers, actors, studios and relationships among them.

1 Introduction

The real world is full of information networks. For these networks, there are some quantities (or attributes) associated with objects (or nodes) that are usually of most interest. A number of information networks are heterogeneous information networks (HIN), for example, the IMDb network which contains movies, actors, directors, writers, and studios as different types of objects. Movies in this network cannot be linked directly while they could be linked by same actors, directors, studios or writers. Different links have different types just as different nodes have different types. Figure 1 is an example of the IMDb network, which uses different shapes and colors to indicate different object and link types.

Numeric prediction in HIN is important in real-world cases. For example, people may be interested in predicting box office sales of an upcoming movie based on the IMDb network, or predicting the total number of citations of an author based on the DBLP plus citation network. Moreover, we notice that some real-world inductive regression problems can be transformed into transductive learning problems by constructing a network structure. This network structure regularization offers us additional information to overcome the weakness of standard induction regression. In this paper, we will adhere to the transduction setting and develop a numerical prediction method based on HIN.

We notice that numerical prediction in heterogeneous networks has not been thoroughly studied before. However, classification in heterogeneous networks [1–4] and regression in homogeneous networks [5, 6] have been studied. Some homogeneous and heterogeneous graph-based classification methods do provide numerical 'soft' predictions before assigning the class labels [1, 2, 7]. However, there are two challenges if we apply these methods directly on the numeric prediction problem: 1) most classification methods arbitrarily set the labels of unlabeled objects to be zeros in the fitting constraint items, which will dramatically shrink the numeric prediction to zero; 2) if unlabeled objects are removed from the fitting constraint items, large variance of prediction could be a problem since numeric prediction is too sensitive to the global network structure. Cortes and Mohri [5] addressed this problem in homogeneous networks where pseudo-labels of unlabeled objects are estimated based on local information and an additional item controlling the distance between predictions and pseudo-labels is applied. Thus based on the philosophy of the regularization framework, we exploit the idea of local estimated labels for unlabeled object-

*University of Illinois at Urbana-Champaign.
†U.S. Army Research Laboratory.

s and meta-path based HIN modeling skills to develop a **g**raph **re**gularized **m**eta-**p**ath based **t**ransductive regression model (*Grempt*) in HIN. Compared with previous HIN models, we conclude the contributions of our model as following:

- Our study is the first one to address the numerical prediction problem in heterogeneous information networks;
- The response variable is narrowed to single type of objects, and meta-path and *PathSim* are used to perceive the similarity between objects;
- Local estimated label (pseudo-label) is used to regularize the numerical prediction precision;
- The contribution of each type of meta-path which corresponds to specific semantic meaning can be automatically obtained from our model.

We will briefly introduce some related work in Section 2. In Section 3, we will introduce the background of heterogeneous information networks. In section 4, we will introduce our *Grempt* model and the implementation algorithm. Details and results of the experiment will be described in Section 5. In Section 6, we will provide our conclusion and future directions.

2 Related Work

A straightforward idea to predict unknown attribute of an object in the network is exploiting its neighbors' information. *Relational Neighbor Classifier* [8] and *Nearest Neighbor Prediction* [9] are typical methods with this philosophy.

Another well established prediction method in a homogeneous setting is *Kernel Regression*, which restricts the search for an appropriate estimator of labeled and unlabeled objects $\hat{h}$ in Reproducing Kernel Hilbert Space $\mathcal{H}_k$ [10]. *Transductive Regression* in homogeneous networks can be regarded as a generalization of kernel regression, where the idea of exploiting neighborhood information is also included [5, 6].

For heterogeneous networks, some graph-based classification models [1–3] have been proposed. The general framework of these methods is based on the similar assumptions of kernel regression, which has a two-item objective function – the global structure smoothness item and the goodness-of-fit item. However, these classification methods either do not include unlabeled objects in the second item or arbitrarily set the labels of unlabeled objects to be zeros in the fitting constraint items, which may not be suitable for our numeric prediction problem.

3 Background

3.1 Problem Definition In this study, a heterogeneous information network (HIN) can be defined as a graph $G = (V, E)$, where $X_i = \{x_{i1}, ..., x_{in_i}\}, i = 1, 2, ..., t$ are t types of data objects, $V = \cup_i^t X_i$ and E =$\{$links between any two data objects in $V\}$. If weights of links are specified, $G = (V, E)$ can be extended to be $G = (V, E, R)$, where R =$\{$weights of links in $E\}$ and V, E are defined as before. We are interested in particular objects and their associated numerical variable.

Suppose $\mathcal{X}$ =$\{X_1, X_2, ..., X_t\}$. Given some labels of a numerical variable Y associated with a particular type of objects $X_* \in \mathcal{X}$, the problem is to predict this variable for unlabeled objects of this type. Different from standard inductive regression which requires a fully labeled training set to derive a specific function, we consider the transductive setting where the unlabeled objects are involved in the learning procedure and the specific function is not of interest. This problem can be formally defined as:

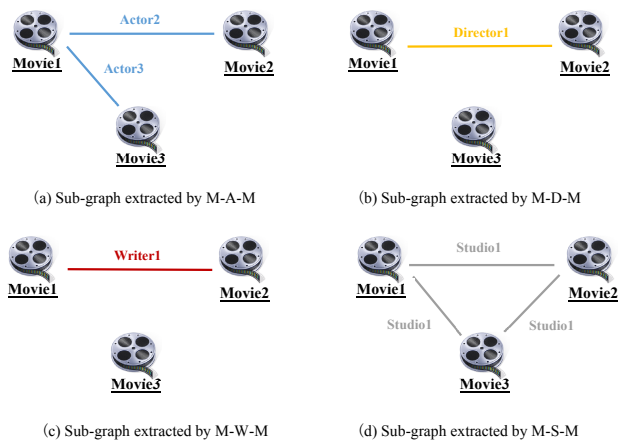
DEFINITION 1. **(Transductive Regression on HIN)** *For a given HIN $G = (V, E)$, suppose variable Y is associated with $X_* \in \mathcal{X}$. Suppose the number of labeled objects is n and the number of unlabeled objects is m. Given the full space of X_* which is composed of $n + m$ objects $x_1, x_2, ..., x_n, x_{n+1}, ..., x_{n+m}$, the labeled subspace with Y can be defined as*

$$(x_1, y_1), (x_2, y_2), ..., (x_n, y_n) \in X_* \times \mathbb{R},$$

and remaining objects $x_{n+1}, ..., x_{n+m}$ are regarded as unlabeled objects. If the purpose of the learning procedure is to infer $y_{n+1}, ..., y_{n+m}$ of unlabeled objects, we call it transductive regression.

3.2 Meta-path and Meta-path Based Similarity In most cases, it may not be suitable to force the target variable Y to represent the characteristics of all types of objects. For example, among movie, actor, actress, studio, genre, writer and other object types in the IMDb network, box office sales is only suitable to be associated with movie. In addition, because of the diversity of links, HINs usually include a large number of objects and edges. Thus the computational cost is high if all types of objects are considered in the whole learning procedure. Therefore, we need to pre-compute some measures which could represent the type of links and then only focus on our target type of objects in the subsequent procedure.

Meta-path and meta-path based similarity have been studied and applied in several HIN related problems [3, 4, 11, 12]. Our model is to shrink the topology of $G = (V, E)$ based on different types of meta-paths and only keep the objects of interest. Thus we define network schema and topology-shrinking sub-networks in the following paragraphs. Sun et al. defined the *network schema* as a meta template for a heterogeneous network,



(a) Sub-graph extracted by M-A-M (b) Sub-graph extracted by M-D-M

(c) Sub-graph extracted by M-W-M (d) Sub-graph extracted by M-S-M

Figure 2: Four sub-graphs extracted from the IMDb example showed in Figure 1 based on four different meta-paths: a) movie-actor-movie; b) movie-director-movie; c) movie-writer-movie; d) movie-studio-movie.

and they provided the definition of *Meta-path* based on this network schema [11]. If $\mathcal{A}$ denotes object types and $\mathcal{R}$ denotes relation types, then a meta-path P can be denoted as $A_1 \rightarrow R_1 \rightarrow A_2 \rightarrow R_2 \rightarrow ... \rightarrow R_l \rightarrow A_{l+1}$, where $A_i \in \mathcal{A}$ and $R_j \in \mathcal{R}$. This meta-path P indicates a composite relation $R = R_1 \circ R_2 \circ ... \circ R_l$ between types A_1 and A_{l+1}, where $\circ$ denotes the composition operator on relations [11].

For this transductive regression problem, we only consider meta-paths where $A_1 = A_{l+1}$. This is because we are only interested in one certain type T of objects, such as movies in IMDb network and papers in DBLP network. Suppose T is the type of X_*. Given one or more meta-paths $P_1, ..., P_K$ in which $A_1 = A_{l+1} = T$, we can define the topology shrinking sub-network composed of a particular type of objects as the following.

DEFINITION 2. **(Topology shrinking Sub-network)** *Given a heterogeneous information network $G = (V, E)$ and a type of meta-paths P, the topology shrinking sub-network of the certain object type T can be denoted as $G_T = (V_T, E_T, R_T)$, $V_T = X_*$, $E_T = \{e_{uv} | p_{x_u \rightsquigarrow x_v} \in P, x_u, x_v \in V_T\}$ and $R_T = \{R_{uv} | R_{uv}$ is the weight of $e_{uv} \in E_T\}$, where $p_{x_u \rightsquigarrow x_v}$ denotes a path instance between x_u and x_v.*

Given a set of meta-paths $P_1, P_2, \cdots, P_K$, our analysis is based on the corresponding set of associated topology-shrinking sub-networks $G_T^{(k)} = (V_T, E_T^{(k)}, R_T^{(k)})$, $k = 1, 2, ..., K$. For the particular IMDb example in Figure 1, Figure 2 shows four sub-graphs extracted based on four different meta-paths: a) movie-actor-movie; b) movie-director-movie; c) movie-writer-movie; d) movie-studio-movie.

When we obtain the structure of $G_T^{(k)}$, what we need to do is to decide the weight of each link. Thus we introduce a meta-path based similarity measure *PathSim* [11], which can favor objects with strong connectivity and similar visibility, i.e. "peers", under the given meta-path. Given a symmetric meta-path P_k, *PathSim* between two objects x_u and x_v of the same type can be defined as

$$s_k(x_u, x_v) = \frac{2 \times |\{p_{x_u \rightsquigarrow x_v} : p_{x_u \rightsquigarrow x_v} \in P_k\}|}{|\{p_{x_u \rightsquigarrow x_u} : p_{x_u \rightsquigarrow x_u} \in P_k\}| + |\{p_{x_v \rightsquigarrow x_v} : p_{x_v \rightsquigarrow x_v} \in P_k\}|}$$

where $p_{x_u \rightsquigarrow x_v}$, $p_{x_u \rightsquigarrow x_u}$ and $p_{x_v \rightsquigarrow x_v}$ are path instances. Then for a homogeneous sub-graph $G_T^{(k)}$, the weight $R_{uv}^{(k)}$ of the link $e_{uv}^{(k)}$ can be defined as the *PathSim* measure $s_k(x_u, x_v)$ between x_u and x_v based on the meta-path P_k. If there is no link between an object x_u and itself, the weight $R_{uu}^{(k)}$ will be zero. In this study, we use a relation matrix $\mathbf{R}^{(k)} = \{R_{uv}^{(k)}\}_{(n+m)\times(n+m)}$ to denote $R_T^{(k)}$. To simplify computation, we only consider the undirected $G_T^{(k)}$ in subsequent sections and thus $\mathbf{R}^{(k)}$ will be symmetric. However, the same procedure can be used to do numerical prediction on directed graph as well.

4 Model

Our graph regularized meta-path based transductive regression model (*Grempt*) is based on the consistency among network data. In the context of meta-path and similarity measure *PathSim*, our model follows three principles: 1) predictions of the target variable of two linked objects are likely to be similar, and the tighter the link is, the more similar the two predictions are; 2) predictions of the target variable of labeled objects should be similar to their labels; 3) predictions of the target variable of unlabeled objects should be similar to their local estimated labels (pseudo-labels).

Particularly, pseudo-label is significant in our model since local regularization could be introduced to improve the prediction, which is also the key difference between *Grempt* and previous HIN models. If only global information is included, the prediction would shrink to the global mean, which might influence the performance on some sparse HINs. However, only using local estimates is not sufficiently robust in network prediction problems. Combining global information and local information can be regarded as a kind of model averaging method which takes advantage of both two types of model, so that it can improve the prediction power based on both global consistency and local consistency.

In this section, we will first introduce the general framework of our model based on these intuitions. Then we will describe the details of estimating pseudo-labels

and the algorithm for optimizing the objective function which controls both global and local consistency.

4.1 Global and Local Graph Regularized Framework. This *Grempt* model is a constraint optimization framework based on the three consistency principles we discussed above. Given K meta-paths $P_1, P_2, ..., P_K$, we can obtain a set of topology-shrinking homogeneous sub-networks of type T, denoted by $G_T^{(k)} = (V_T, E_T^{(k)}, R_T^{(k)})$, $k = 1, 2, ..., K$. We first introduce some notations as following:

$\mathbf{y}_L = (y_1, ..., y_n)^T$ denotes a vector of true labels of labeled objects $x_1, ..., x_n$;
$\mathbf{y}_U = (y_{n+1}, ..., y_{n+m})^T$ denotes a vector of true labels of unlabeled objects $x_{n+1}, ..., x_{n+m}$;
$\mathbf{y} = (\mathbf{y}_L^T, \mathbf{y}_U^T)^T$.
$\tilde{\mathbf{y}}_U = (\tilde{y}_{n+1}, ..., \tilde{y}_{n+m})^T$ denotes a vector of pseudo-labels of unlabeled objects $x_{n+1}, ..., x_{n+m}$;
$\mathbf{w} = (w_1, ..., w_K)$ denotes a vector of weights of sub-networks $G_T^{(k)} = (V_T, E_T^{(k)}, R_T^{(k)})$, $k = 1, 2, ..., K$.

Suppose the estimation of y_u from our model is denoted by f_u, $u = 1, ..., n, n+1, ..., n+m$, then we have following notations:

$\mathbf{f}_L = (f_1, ..., f_n)^T$ denotes estimations of $\mathbf{y}_L$;
$\mathbf{f}_U = (f_{n+1}, ..., f_{n+m})^T$ denotes predictions of $\mathbf{y}_U$;
$\mathbf{f} = (\mathbf{f}_L^T, \mathbf{f}_U^T)^T$.

Then the objective function in our optimization framework can be defined as

$$\mathbf{J}(\mathbf{w}; \mathbf{f}) = \mathbf{\Omega}(\mathbf{w}; \mathbf{f}) + \alpha_1 \mathbf{C}_1(\mathbf{f}_L; \mathbf{y}_L) + \alpha_2 \mathbf{C}_2(\mathbf{f}_U; \tilde{\mathbf{y}}_U). \tag{4.1}$$

In this function, α_1 and α_2 are two given parameters, and $\mathbf{\Omega}(\mathbf{w}; \mathbf{f})$, $\mathbf{C}_1(\mathbf{f}_L; \mathbf{y}_L)$ and $\mathbf{C}_2(\mathbf{f}_U; \tilde{\mathbf{y}}_U)$ are three different loss functions to guarantee the previous three principles respectively.

- The first item $\mathbf{\Omega}(\mathbf{w}; \mathbf{f})$ in the objective function (4.1) is a composite graph regularization item controlling the global consistency among all the topology-shrinking sub-graphs $G_T^{(k)}$, $k = 1, 2, ..., K$. It can be defined as
(4.2)
$$\Omega(\mathbf{w}; \mathbf{f}) = \sum_{k=1}^{K} w_k \sum_{u,v=1, u \neq v}^{m+n} R_{uv}^{(k)} \left(\frac{f_u}{\sqrt{D_u^{(k)}}} - \frac{f_v}{\sqrt{D_v^{(k)}}} \right)^2,$$
where $D_u^{(k)}$ is the summation of u-th row in $\mathbf{R}^{(k)}$. This item controls not only the global consistency of each graph $G_T^{(k)}$ but also the consistency of different sub-graphs, where $\mathbf{w}$ and $\mathbf{f}$ are two sets of unknown variables. With the constraint (4.5), the weight vector $\mathbf{w}$ reflects the contribution of each sub-graph $G_T^{(k)}$'s structure to the consistency of target variable Y.

- $\mathbf{C}_1(\mathbf{f}_L; \mathbf{y}_L)$ in (4.1) is a loss function controlling the difference between predicted label values $\mathbf{f}_L$ and given labels $\mathbf{y}_L$ of labeled objects. In *Grempt* model, we use a quadratic loss function which can be simply defined as
(4.3)
$$\mathbf{C}_1(\mathbf{f}_L; \mathbf{y}_L) = \sum_{u=1}^{n} (f_u - y_u)^2 = (\mathbf{f}_L - \mathbf{y}_L)^T (\mathbf{f}_L - \mathbf{y}_L).$$

- Similarly, $\mathbf{C}_2(\mathbf{f}_U; \tilde{\mathbf{y}}_U)$ in (4.1) is a loss function controlling the difference between predicted values $\mathbf{f}_U$ and pseudo-labels $\tilde{\mathbf{y}}_U$ of unlabeled objects. This pseudo-label estimation usually involves location information and thus can be treated as a local consistency constraint. Since errors can be introduced by estimating the pseudo-label as well, not only the raw difference but also the reliability of the pseudo-label which is represented by variance need to be taken into account. Therefore a Mahalanobis-distance-type loss function is used in our *Grempt* model. Specifically, it can be defined as
$$\begin{aligned} \mathbf{C}_2(\mathbf{f}_U; \tilde{\mathbf{y}}_U) &= \sum_{v=1}^{m} \frac{(f_{n+v} - \tilde{y}_{n+v})^2}{\sigma_{\tilde{y}_{n+v}}^2} \\ &= (\mathbf{f}_U - \tilde{\mathbf{y}}_U)^T \Sigma^{-1} (\mathbf{f}_U - \tilde{\mathbf{y}}_U), \end{aligned} \tag{4.4}$$
where $\sigma_{\tilde{y}_{n+v}}^2$ is the variance of x_{n+v}'s pseudo-label estimation, Σ is a $m \times m$ diagonal matrix, the (v, v)-th element of which is $\sigma_{\tilde{y}_{n+v}}^2$. Specific pseudo-label estimation procedure will be introduced later on.

- The parameters α_1 and α_2 control the trade-off among all three items. These two parameters can be assigned based on prior knowledge or determined by cross-validation.

Our target is seeking $\mathbf{f}$ and $\mathbf{w}$ to minimize this objective function subject to a constraint $\delta(\mathbf{w}) = 0$. Specifically in our model, we use the constraint function $\delta(\mathbf{w}) = \sum_{k=1}^{K} \exp(-w_k) - 1$. This constraint ensures the problem can be converted into a convex optimization problem and closed-form global optimization solution of w_k can be obtained. Then this problem can be written as

$$\begin{aligned} \min_{\mathbf{f}, \mathbf{w}} \ \mathbf{J}(\mathbf{w}; \mathbf{f}) =& \mathbf{\Omega}(\mathbf{w}; \mathbf{f}) + \alpha_1 \mathbf{C}_1(\mathbf{f}_L; \mathbf{y}_L) + \alpha_2 \mathbf{C}_2(\mathbf{f}_U; \tilde{\mathbf{y}}_U) \\ =& \sum_{k=1}^{K} w_k \left[\sum_{u,v=1, u \neq v}^{m+n} R_{uv}^{(k)} \left(\frac{f_u}{\sqrt{D_u^{(k)}}} - \frac{f_v}{\sqrt{D_v^{(k)}}} \right)^2 \right] \\ &+ \alpha_1 \sum_{u=1}^{n} (f_u - y_u)^2 + \alpha_2 \sum_{v=1}^{m} \frac{(f_{n+v} - \tilde{y}_{n+v})^2}{\sigma_{\tilde{y}_{n+v}}^2} \end{aligned}$$

subject to

$$\sum_{k=1}^{K} \exp(-w_k) = 1. \tag{4.5}$$

We thus conclude that the optimization algorithm can be implemented in two stages:

1. Estimating pseudo-labels $\tilde{\mathbf{y}}_U$ of unlabeled objects and their associated variance using local information;
2. Given pseudo-labels, optimizing the objective function (4.1) subject to constraint (4.5).

We will introduce details of each stage in next two subsections.

4.2 Pseudo-label Estimation. There are several approaches to determine pseudo-labels. In this study, pseudo-labels are estimated based on the position of unlabeled objects. Specifically, we only consider neighborhood information based on the equal combination of all homogeneous sub-networks $G_T^{(k)}$[1]. The combined relation matrix can be defined as $\mathbf{R} = \sum_{k=1}^{K} \mathbf{R}^{(k)}$, where the (u, v)-th element is $R_{uv} = \sum_{k=1}^{K} R_{uv}^{(k)}$. Then the labeled q-nearest neighborhood based on the combination of sub-networks $G_T^{(k)}$ of an unlabeled object x_{n+v} can be defined as

$$\mathcal{N}_q(x_{n+v}) = \{x_u | R_{n+v,u} > 0, 1 \le u \le n,$$
$$R_{n+v,u} \in \{\text{largest } q \text{ elements of } R_{n+v,1}, ..., R_{n+v,n}\}\}.$$

We use a simple relational model to describe this local information and obtain the pseudo-label of x_{n+v} and the variance of this distribution. Given the labeled q-nearest neighborhood of x_{n+v}, suppose y_{n+v} follows a discrete distribution where for $x_u \in \mathcal{N}_q(x_{n+v})$,

$$p_{n+v,u} = P(y_{n+v} = y_u | \mathcal{N}_q(x_{n+v})) \propto R_{n+v,u}.$$

Then $\tilde{y}_{n+v}$ can be assigned to the mean of this distribution, i.e.

$$\tilde{y}_{n+v} = \sum_{u \in \mathcal{N}_q(x_{n+v})} p_{n+v,u} y_u = \frac{\sum_{u \in \mathcal{N}_q(x_{n+v})} R_{n+v,u} y_u}{\sum_{u \in \mathcal{N}_q(x_{n+v})} R_{n+v,u}}. \tag{4.6}$$

[1] Weight of each shrinking homogeneous sub-network can be given by prior knowledge or determined by some validation methods. However, it is usually tricky to tune these parameters. In this study, we use equal combination as a straightforward example and the experiments indicate that it is enough to show the priority of using pseudo-label.

The variance of this distribution can be calculated as

$$\sigma^2_{\tilde{y}_{n+v}} = \sum_{u \in \mathcal{N}_q(x_{n+v})} p_{n+v,u} (y_u - \tilde{y}_{n+v})^2. \tag{4.7}$$

From (4.7) we notice that if x_{n+v}'s neighbors' labels tend to be similar, $\sigma^2_{\tilde{y}_{n+v}}$ tends to be small. The pseudo-label of x_{n+v} thus tend to be reliable and vice versa.

Based on above description, pseudo-labels of unlabeled objects $\tilde{\mathbf{y}}_U$ and their associating variances $\Sigma = diag\{\sigma^2_{\tilde{y}_{n+v}}\}_{v=1,...,m}$ can be directly computed.

4.3 Optimization Procedure. In this section, we will discuss the algorithm used in the optimization procedure.

In the objective function (4.1), the first item can be explained by matrix transformation. For each relation matrix $\mathbf{R}^{(k)}$,

- $\mathbf{D}^{(k)}$ is a diagonal matrix where the (u, u)-th element is the summation of u-th row in $\mathbf{R}^{(k)}$;
- $\mathbf{S}^{(k)} = [\mathbf{D}^{(k)}]^{-1/2} \mathbf{R}^{(k)} [\mathbf{D}^{(k)}]^{-1/2}$;
- $\mathcal{L}^{(k)} = \mathbf{I} - \mathbf{S}^{(k)} = \mathbf{I} - [\mathbf{D}^{(k)}]^{-1/2} \mathbf{R}^{(k)} [\mathbf{D}^{(k)}]^{-1/2}$ is the normalized Laplacian matrix for the topology-shrinking sub-graphs $G_T^{(k)}$.

Thus we have

$$\Omega(\mathbf{w}; \mathbf{f}) = \sum_{k=1}^{K} w_k (2\mathbf{f}^T \mathbf{f} - 2\mathbf{f}^T \mathbf{S}^{(k)} \mathbf{f}) = 2 \sum_{k=1}^{K} w_k \mathbf{f}^T \mathcal{L}^{(k)} \mathbf{f}, \tag{4.8}$$

which indicates that $\Omega(\mathbf{w}; \mathbf{f})$ can be regarded as a linear combination of normalized Laplacian regularizers based on different sub-graphs $G_T^{(k)}$.

From (4.8) (4.3) and (4.4), the objective function (4.1) can be re-written as

$$\begin{aligned} \mathbf{J}(\mathbf{w}; \mathbf{f}) =& 2 \sum_{k=1}^{K} w_k \mathbf{f}^T \mathcal{L}^{(k)} \mathbf{f} + \alpha_1 (\mathbf{f}_L - \mathbf{y}_L)^T (\mathbf{f}_L - \mathbf{y}_L) \\ &+ \alpha_2 (\mathbf{f}_U - \tilde{\mathbf{y}}_U)^T \Sigma^{-1} (\mathbf{f}_U - \tilde{\mathbf{y}}_U). \end{aligned}$$

Then we can use the block coordinate descent approach [13], which will keep reducing the value of the objective function, to iteratively update $\mathbf{f}$ and $\mathbf{w}$. The first step is to fix $\mathbf{f}$ and update $\mathbf{w}$ to minimize $\mathbf{J}(\mathbf{w}; \mathbf{f})$; the second step is to fix $\mathbf{w}$ and update $\mathbf{f}$ to minimize $\mathbf{J}(\mathbf{w}; \mathbf{f})$. We could iteratively do these two steps until convergence. Here we provide two theorems to help us deduce the algorithm, which can be proved using similar techniques in previous graph regularized regression studies [1, 5].

THEOREM 4.1. *Suppose* $\mathbf{f}$ *is fixed, the objective problem* $\mathbf{J}(\mathbf{w};\mathbf{f})$ *with constraint function* $\delta(\mathbf{w}) = 0$ *is a convex optimization problem. The global optimal solution is given by*

$$w_k = -\log\left(\frac{\mathbf{f}^T\mathcal{L}^{(k)}\mathbf{f}}{\sum_{k=1}^{K}\mathbf{f}^T\mathcal{L}^{(k)}\mathbf{f}}\right). \tag{4.9}$$

THEOREM 4.2. *Suppose* $\mathbf{w}$ *is fixed, the objective problem* $\mathbf{J}(\mathbf{w};\mathbf{f})$ *is a convex optimization problem. The global optimal solution is given by solving the following linear system:*

(4.10)

$$(2\sum_{k=1}^{K} w_k + \alpha_1)\mathbf{f}_L = 2\sum_{k=1}^{K} w_k(\mathbf{S}_{11}^{(k)}\mathbf{f}_L + \mathbf{S}_{12}^{(k)}\mathbf{f}_U) + \alpha_1\mathbf{y}_L;$$

$$(2\sum_{k=1}^{K} w_k + \alpha_2\Sigma^{-1})\mathbf{f}_U = 2\sum_{k=1}^{K} w_k(\mathbf{S}_{21}^{(k)}\mathbf{f}_L + \mathbf{S}_{22}^{(k)}\mathbf{f}_U) + \alpha_2\Sigma^{-1}\tilde{\mathbf{y}}_U.$$

where

$$\mathbf{S}^{(k)} = \begin{bmatrix}\mathbf{S}_{11}^{(k)} & \mathbf{S}_{12}^{(k)}\\ \mathbf{S}_{21}^{(k)} & \mathbf{S}_{22}^{(k)}\end{bmatrix},$$

is partitioned according labeled and unlabeled objects.

We notice that it is nontrivial to obtain a closed form solution by jointly solving equations (4.9) and (4.10). Although given $\mathbf{w}$, (4.10) can be solved directly, for time and space efficiency, we can provide an iterative algorithm as well, which can be described as

1. Determine pseudo-label $\tilde{y}_{n+v}$ and corresponding variance $\sigma^2_{\tilde{y}_{n+v}}$ based on (4.6) and (4.7);
2. Initialize $t = 0$, $w_1(0) = w_2(0) = ... = w_K(0) = log(K)$, and $\mathbf{f}(0) = (\mathbf{y}_L^T, \tilde{\mathbf{y}}_U^T)^T$;
3. Suppose we have $\mathbf{w}(t)$ and $\mathbf{f}(t)$, then use $\mathbf{f}(t)$ to calculate $\mathbf{w}(t+1)$ based on (4.9), and use $\mathbf{w}(t+1)$ to update $\mathbf{f}(t+1)$ based on the following rules:

$$f_u(t+1) = \frac{2\sum_{k=1}^{K} w_k(\mathbf{S}_{11}^{(k)}\mathbf{f}_L(t) + \mathbf{S}_{12}^{(k)}\mathbf{f}_U(t))_u + \alpha_1 y_u}{2\sum_{k=1}^{K} w_k + \alpha_1},$$

$$u = 1, 2, ..., n;$$

$$f_{n+v}(t+1) = \frac{2\sum_{k=1}^{K} w_k(\mathbf{S}_{21}^{(k)}\mathbf{f}_L(t) + \mathbf{S}_{22}^{(k)}\mathbf{f}_U(t))_v + \frac{\alpha_2\tilde{y}_{n+v}}{\sigma^2_{\tilde{y}_{n+v}}}}{2\sum_{k=1}^{K} w_k + \frac{\alpha_2}{\sigma^2_{\tilde{y}_{n+v}}}},$$

$$v = 1, 2, ...m.$$

where $(\cdot)_u$ indicates the u-th element of a vector.

4. Repeat previous procedure until $\mathbf{w}(t)$ and $\mathbf{f}(t)$ converge.

4.4 Time Complexity Analysis. We only consider the computational complexity of above iterative method for the objective function optimization in this section. Suppose $m + n$ is the number of objects, K is the number of meta-paths we selected, and $|E_T^{(1)}|, .., |E_T^{(K)}|$ are the numbers of edges of topology-shrinking sub-networks given meta-path $P_1, ..., P_K$. Suppose the relation matrix $\mathbf{R}^{(k)}$ and its corresponding normalized matrix $\mathbf{S}^{(k)}$ are pre-computed before the learning procedure. Then for initialization, we need $O(m+n+K)$ time. For each iteration, we need to scan each edge in $G_T^{(1)}, ..., G_T^{(K)}$ to do matrix multiplication, scan each object and each type of meta-path to do other arithmetics. Therefore the objective function optimization costs $O((n+m+K)+N(n+m+K+\sum_{k=1}^{K}|E_T^{(k)}|))$ time, where N is the number of iteration. Since the number of objects of topology shrinking sub-networks are much smaller than the number of objects in the original HIN and the iterative procedure converges rapidly ($N < 20$ in our experiments), our algorithm is time and space scalable.

5 Experiment

5.1 Dataset. We applied our model to two sets of data – data from the Internet Movie Database (IMDb) and data from the DBLP Computer Science Bibliography.

- The IMDb data used in this study are extracted from the IMDb interface [2] and Box Office Mojo[3] affiliated with IMDb. We keep the data related to movies whose names can be exactly matched based on these two sources. For the combined dataset, we only keep the movies released in 2000-2013 with at least 1000 user votes on the IMDb website and related actors, actresses, directors, writers, genres and studios. In this dataset, the target variable is log(*box office sales*), which is associated with movie. The meta-path we used in the IMDb network are movie-actor-movie (M-A_1-M), movie-actress-movie (M-A_2-M), movie-director-movie (M-D-M), movie-genre-movie (M-G-M), and movie-writer-movie (M-W-M). Notice that in the experiment, we only keep the actors, actresses, directors, genres and writers, each of which appears in at least two movies, and the movies which are related to these objects. The final IMDb network used in the experiment contains 3300 movies, 18845 actors, 9065 actresses, 746 directors, 20 genres, 197 studios and 1623 writers. To address the temporal nature of movies, we labeled four different sets of movies based their released years. The summary of these four datasets is showed in Table 1.
- The DBLP data used in this study are collected by ArnetMiner[4] [14], which contains all papers from DBLP and a fraction of citation relationships between papers. The latest version was updated in Sep. 2013. We keep papers published in 2009-2013, data mining and

[2] http://www.imdb.com/
[3] http://www.boxofficemojo.com/
[4] http://arnetminer.org/DBLP_Citation

IMDb	Number of labeled objects	Number of unlabeled objects	Percentage of labeled objects
dataset1	3067 (2000–2012)	233 (2013–2013)	92.94%
dataset2	2820 (2000–2011)	480 (2012–2013)	85.45%
dataset3	2578 (2000–2010)	722 (2011–2013)	78.12%
dataset4	2345 (2000–2009)	955 (2010–2013)	71.06%
DBLP	**Number of labeled objects**	**Number of unlabeled objects**	**Percentage of labeled objects**
dataset1	3017	315	90.55%
dataset2	1666	1666	50.00%
dataset3	334	2998	10.02%
dataset4	167	3165	5.01%

Table 1: Summary of IMDb datasets (numbers in parentheses indicate released year) and DBLP datasets.

machine learning related venues[5], related authors, venues and citation relationship. In this dataset, the target variable is $\log(\#citation + 1)$ where for a particular author, $\#citation$ is the total citation number of papers he/she published in 2009-2013. We only consider authors who have published papers in 2009 and have published at least two papers in 2009-2013. The meta-path used in the DBLP network are author-paper-author (A-P-A), author-venue-author (A-V-A), and author-paper-(cited by)-paper-(cite)-paper-author (A-P←P→P-A). The final DBLP network used in the experiment contains 3332 authors, 1289 papers, 1046 terms and 27 venues. For DBLP data, we randomly labeled four different sets of authors according to different label proportions. To address the cases where labels are limited, we labeled a small portion of data (10% and 5%) in the last two datasets. The summary of these four datsets is showed in Table 1.

5.2 Preprocessing. For both IMDb data and DBLP data, we are more interested in the log transformation of original response variable box office sales or the number of citation because of their wide ranges. Besides, to easily compare the parameters α_1 and α_2 in two datasets, we normalize the original label values $y_u, u = 1, ..., n$ of $\log(box\ office\ sales)$ and $\log(\#citation + 1)$ to occupy the unit interval $[0, 1]$ by

$$(y_u - \min_{u=1,..,n} y_u)/(\max_{u=1,..,n} y_u - \min_{u=1,...,n} y_u),$$

and use these values as inputs y_u, $u = 1, 2, ..., n$. When the outputs $f_{n+v}, v = 1, ..., m$ are obtained, we use the inverse transformation

$$f_{n+v} \times (\max_{u=1,..,n} y_u - \min_{u=1,..,n} y_u) + \min_{u=1,..,n} y_u$$

to transform them back. These transformed predicted values are used in the model evaluation procedure.

[5] AAAI, CIKM, CVPR, ECIR, ECML, EDBT, ICDE, ICDM, ICML, IJCAI, KDD, PAKDD, PKDD, PODS, SDM, SIGIR, SIGMOD, VLDB, WWW, WSDM, SIGMOD record, ACM trans. database syst., data knowl. eng., data min. knowl. discov., IEEE data eng. bull., IEEE trans. knowl. data eng., j. database manag., journal of machine learning research, machine learning, knowl. inf. syst., SIGKDD explorations, VLDB j.

5.3 Models For Comparison. We compare our graph regularized meta-path based transductive regression model (*Grempt*) with six different models – *Lasso*, *RN_ntp*, *RN*, *TRnloc_ntp*, *TRnloc* and *Grempt_ntp*.

- **Lasso [15].** In order to show the necessity of transduction setting in network data, we compare our model to a state-of-the-art inductive regression model – *Lasso*, which is also regarded as the baseline method in this study. When applying *Lasso* regression on IMDb data, objects except movies are treated as categorical variables associated with movies. Similarly, for DBLP data, objects except authors and citation relationships are treated as categorical variables.
- **k-nearest Relational Neighbor Estimation.** This relational neighbor prediction model which only involves local estimated labels shares the similar idea to *Relational Neighbor Classifier* (RN) [8]. However, we only use the k-nearest neighbors to to estimate labels of unlabeled objects, which is the same as the previous pseudo-label estimation method. We consider two different k-nearest RN models – the RN model regardless of different types of meta-path (*RN_ntp*) and the one in which types of meta-path are considered (*RN*). For *RN_ntp*, we treated all types of meta-path as a same type so that the input HIN could be a homogeneous one. Then we calculated relation matrix based on this unified meta-path, and the same as other non-type models used for comparison.
- **Transductive Regression Without Local Estimation.** This transductive regression model without using local estimated labels is equivalent to our *Grempt* model without the third item i.e. $\alpha_2 \equiv 0$. This two-term objective function is similar to two state-of-the-art HIN classification methods *GNetMine* [1] and *RankClass* [2]. Similar with previous method, we also consider two scenarios – all meta-paths are regarded as in the same type (*TRnloc_ntp*) and different types of meta-paths are involved (*TRnloc*).
- **Homogeneous Grempt Model.** To validate the different contributions of different meta-paths, we compare our standard *Grempt* model with a *Grempt_ntp* model where meta-paths are regarded as in the same type.

5.4 Evaluation Measure. All of the models are evaluated by mean absolute prediction error (MAE), which has the same scale of data and is relative insensitive to outliers. For the unlabeled objects x_{n+v}, $v = 1, ..., m$, we have

$$MAE = \frac{1}{m} \sum_{v=1}^{m} |f_{n+v} - y_{n+v}|.$$

5.5 Performance. In *RN_ntp*, *RN* and pseudo-label estimation in *Grempt_ntp* and *Grempt*, we use 5-nearest neighbors on both IMDb data and DBLP data and e-

-	Dataset1	%label=92.94%	Dataset2	%label=85.45%
Method	**MAE**	**Improvement**	**MAE**	**Improvment**
Lasso	2.824	Baseline	2.878	Baseline
RN_ntp	2.104	25.49%	2.163	24.85%
RN	2.031	28.08%	2.064	28.31%
TRnloc_ntp	2.213	21.63%	2.196	23.70%
TRnloc	2.858	-1.20%	3.059	-6.26%
Grempt_ntp	2.095	25.82%	2.144	25.52%
Grempt	**1.912**	**32.28%**	**1.941**	**32.57%**
-	**Dataset3**	%label=78.12%	**Dataset4**	%label=71.06%
Method	**MAE**	**Improvement**	**MAE**	**Improvment**
Lasso	2.929	Baseline	2.761	Baseline
RN_ntp	2.131	27.24%	2.096	24.10%
RN	2.079	29.02%	2.025	26.65%
TRnloc_ntp	2.230	23.85%	2.232	19.19%
TRnloc	3.272	-11.72%	3.362	-21.75%
Grempt_ntp	2.115	27.77%	2.084	24.55%
Grempt	**1.969**	**32.77%**	**1.916**	**30.61%**

Table 2: Results of Prediction Error on IMDb Datasets.

-	Dataset1	%label=90.55%	Dataset2	%label=50.00%
Method	**MAE**	**Improvement**	**MAE**	**Improvment**
Lasso	0.7410	Baseline	0.8152	Baseline
RN_ntp	0.6733	9.14%	0.7886	3.27%
RN	0.6689	9.73%	0.8196	-0.54%
TRnloc_ntp	0.8551	-15.40%	0.94	-15.31%
TRnloc	0.6359	14.18%	0.7754	4.89%
Grempt_ntp	0.8213	-10.85%	0.9139	-12.11%
Grempt	**0.6352**	**14.28%**	**0.7745**	**5.00%**
-	**Dataset3**	%label=10.02%	**Dataset4**	%label=5.01%
Method	**MAE**	**Improvement**	**MAE**	**Improvment**
Lasso	1.1935	Baseline	0.9673	Baseline
RN_ntp	0.9217	22.78%	0.958	0.96%
RN	0.9631	19.31%	0.9687	-0.15%
TRnloc_ntp	1.0143	15.02%	1.0788	-11.53%
TRnloc	0.9533	20.13%	1.0735	-10.98%
Grempt_ntp	0.9212	22.82%	0.9531	1.47%
Grempt	**0.9023**	**24.40%**	**0.9342**	**3.42%**

Table 3: Results of Prediction Error on DBLP Datasets.

IMDb	**Dataset 1**	**Dataset 2**	**Dataset 3**	**Dataset 4**
Running time(s)	8.876	10.956	9.084	10.799
DBLP	**Dataset 1**	**Dataset 2**	**Dataset 3**	**Dataset 4**
Running time(s)	5.5640	6.345	6.095	6.392

Table 4: Running Time of Single Experiment of *Grempt* Model on IMDb and DBLP Datasets

qual weighted combination of relation matrix of different homogeneous sub-networks. We notice that label accuracy is very important in our model since MAE decreases as α_1 increases on all the datasets. However, the relative importance of the local estimates α_2 varies for IMDb data and DBLP data. We notice that for the DBLP network, the importance of pseudo-label varies for different percentages of labeled objects. α_2 could be determined based on cross-validation in different datasets. For sparse network like IMDb, local estimates are more important so that we suggest relatively large values as candidates for α_2. We also notice that dense network like DBLP with a small percentage of labeled objects has a similar property. For dense network like DBLP with sufficient labeled objects, however, global consistencies are more significant and thus relatively small values for α_2 are suggested.

Experimental results on IMDb datasets and DBLP datasets are showed in Table 2 and Table 3 respectively. Here, in *TRnloc_ntp*, *TRnloc*, *Grempt_ntp* and our model *Grempt*, we set $\alpha_1 = 2000$. For *Grempt_ntp* and *Grempt*, we set $\alpha_2 = 3$ for all four IMDb datasets, $\alpha_2 = 0.005$ for DBLP dataset1 and dataset2, and $\alpha_2 = 1$ for DBLP dataset3 and dataset4. These parameters for *Grempt* model are not optimal setting, but are enough

IMDb datset 4 Name	**log(box office sales) Groundtruth**	**- Prediction**
The Hobbit: An Unexpected Journey	19.53	18.68
The Hobbit: The Desolation of Smaug	19.37	18.80
The Hunger Games	19.83	17.44
The Hunger Games: Catching Fire	19.87	17.67
Kung Fu Panda 2	18.92	18.55
Nebraska	16.69	16.27
Before Midnight	15.91	14.71
Shahid	9.41	13.05
Udaan	8.92	13.15

Table 5: Prediction Examples of log($box\ office\ sales$) from *Grempt* model applied on IMDb dataset4.

to show the superiority of our model. From these two tables, we can conclude that our *Grempt* model has the best performance on both IMDb datasets and DBLP datasets. Running time of *Grempt* model are showed in Table 4, which indicates that each single experiment of our method can be executed within seconds.

Some representative examples from IMDb dataset4 are selected to show the predictions obtained from our model, which are displayed in Table 5. We thus conclude that the *Grempt* model has the potential to predict the numeric variable in heterogeneous information networks Objects whose predicted values are much different from true values may need to be analyzed case-by-case.

We notice that traditional regression methods such as *Lasso* cannot predict the value of target variable precisely because it lacks the ability to capture the structure information of the network. Methods only using local information (*RN_ntp*, *RN*) and methods only using global consistency (*TRnloc_ntp*, *TRnloc*) have different performance on IMDb and DBLP datasets because of their different structure characteristics. Since the IMDb network is much sparser than the DBLP network, local information in the IMDb network could be more reliable than global consistency which is reversed in the DBLP network. However, our model can balance these two kinds of consistency so that it can yield a better overall result. In addition, poor performances of *RN_ntp*, *TRnloc_ntp* and *Grempt_ntp* indicate that heterogeneous structures cannot be ignored in graph-based numerical prediction problems.

The vector of weights of different meta-paths $\mathbf{w}$ obtained from the iterative algorithm on IMDb data and DBLP data are plotted in Figure 3. It can be concluded that for the IMDb network, movie-actor-movie and movie-actress-movie have more significant influence on the box office sales of a movie than other meta-paths, and movie-genre-movie is the least important among all selected meta-paths. For the DBLP network, author-paper-author and author-venue-author are more significant than author-paper-(cited by)-paper-(cite)-paper-author with respect to the total citation number of an author. Moreover, from Figure 3 we notice that contributions of those important meta-paths will increase as the number of labeled objects decreases, while con-

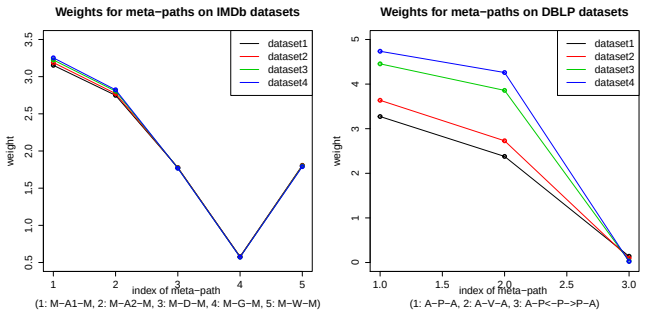


Figure 3: Weights for meta-paths of IMDb datasets and DBLP datasets from *Grempt* Model.

tributions of relatively unimportant meta-paths will decrease.

6 Conclusion and Future Work

In this paper, we proposed a meta-path based transductive regression model in HIN which incorporates the ideas of global graph-based consistency and local estimation. We obtained the best performance among all candidate frameworks for box office sales prediction in IMDb network and total citation number prediction in DBLP network.

There are some potential improvements of this initial research in numerical prediction in HIN. In many real-world cases, people may need more accurate results for important objects, such as blockbuster movies and highly-cited authors. Thus ranking information and preference could be introduced in the transductive regression models. We also notice that some variables may correlate with each other (e.g. box office and rating score). Therefore, another problem could be generalizing this model from univariate case (e.g. predicting box office only) to multivariate case (e.g. predicting box office and rating score jointly) based on correlation between variables.

Acknowledgment

Research was sponsored in part by the U.S. Army Research Lab. under Cooperative Agreement No. W911NF-09-2-0053 (NSCTA), the Army Research Office under Cooperative Agreement No. W911NF-13-1-0193, National Science Foundation IIS-1017362, IIS-1320617, and IIS-1354329, HDTRA1-10-1-0120, NIH Big Data to Knowledge (BD2K) (U54), and MIAS, a DHS-IDS Center for Multimodal Information Access and Synthesis at UIUC.

Information courtesy of IMDb[6] and Box Office Mojo[7], used with permission.

[6] http://www.imdb.com
[7] http://www.boxofficemojo.com

References

[1] M. Ji, Y. Sun, M. Danilevsky, J. Han, and J. Gao, "Graph regularized transductive classification on heterogeneous information networks," in *Machine Learning and Knowledge Discovery in Databases*. Springer, 2010, pp. 570–586.

[2] M. Ji, J. Han, and M. Danilevsky, "Ranking-based classification of heterogeneous information networks," in *Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2011, pp. 1298–1306.

[3] C. Luo, R. Guan, Z. Wang, and C. Lin, "Hetpathmine: A novel transductive classification algorithm on heterogeneous information networks," *Advances in Information Retrieval*, pp. 210–221, 2014.

[4] X. Kong, P. S. Yu, Y. Ding, and D. J. Wild, "Meta path-based collective classification in heterogeneous information networks," in *Proceedings of the 21st ACM international conference on Information and knowledge management*. ACM, 2012, pp. 1567–1571.

[5] C. Cortes, M. Mohri, and M. Mohri, "On transductive regression," in *NIPS*, 2006, pp. 305–312.

[6] C. Cortes, M. Mohri, D. Pechyony, and A. Rastogi, "Stability of transductive regression algorithms," in *Proceedings of the 25th international conference on Machine learning*. ACM, 2008, pp. 176–183.

[7] X. Zhu, "Semi-supervised learning literature survey," *Computer Science, University of Wisconsin-Madison*, vol. 2, p. 3, 2006.

[8] S. A. Macskassy and F. Provost, "A simple relational classifier," DTIC Document, Tech. Rep., 2003.

[9] S. A. Dudani, "The distance-weighted k-nearest-neighbor rule," *Systems, Man and Cybernetics, IEEE Transactions on*, no. 4, pp. 325–327, 1976.

[10] A. Berlinet and C. Thomas-Agnan, *Reproducing kernel Hilbert spaces in probability and statistics*. Springer, 2004, vol. 3.

[11] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *VLDB11*, 2011.

[12] Y. Sun, J. Han, C. C. Aggarwal, and N. V. Chawla, "When will it happen?: relationship prediction in heterogeneous information networks," in *Proceedings of the fifth ACM international conference on Web search and data mining*. ACM, 2012, pp. 663–672.

[13] D. P. Bertsekas, "Nonlinear programming," 1999.

[14] 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.

[15] R. Tibshirani, "Regression shrinkage and selection via the Lasso," *Journal of the Royal Statistical Society. Series B (Methodological)*, pp. 267–288, 1996.