Supervised Link Prediction Using Multiple Sources

Zhengdong Lu*, Berkant Savas†, Wei Tang‡, and Inderjit S. Dhillon†‡
*Microsoft Research Asia
†Institute for Computational Engineering and Sciences, The University of Texas at Austin
‡Department of Computer Science, The University of Texas at Austin
Email: zhengdol@microsoft.com, {berkant, wtang, inderjit}@cs.utexas.edu

***Abstract*—Link prediction is a fundamental problem in social network analysis and modern-day commercial applications such as Facebook and Myspace. Most existing research approaches this problem by exploring the topological structure of a social network using only one source of information. However, in many application domains, in addition to the social network of interest, there are a number of auxiliary social networks and/or derived proximity networks available. The contribution of the paper is twofold: (1) a supervised learning framework that can effectively and efficiently learn the dynamics of social networks in the presence of auxiliary networks; (2) a feature design scheme for constructing a rich variety of path-based features using multiple sources, and an effective feature selection strategy based on structured sparsity. Extensive experiments on three real-world collaboration networks show that our model can effectively learn to predict new links using multiple sources, yielding higher prediction accuracy than unsupervised and single-source supervised models.**

***Index Terms*—social network; link prediction; multiple sources; supervised learning;**

I. Introduction

Social networks are dynamic by nature. They change quickly over time when new relationships establish between people (called *actors*), and when old relationships dissolve. These relational changes, characteristics of the actors, characteristics of pairs of actors, and random unexplained influences are the joint contribution to the dynamics of a network topology. Understanding the mechanisms by which the social networks evolve is a fundamental question that is still not well understood, and it forms the motivation for our work here.

In addition to the links in the network, we may also have exogenous features with various level of uncertainty, most interestingly the auxiliary networks between the same group of vertices from heterogeneous sources. Take the Facebook network for example, besides the friendship relations between the users, there are other relations based on blog article citations and commenting, or online messaging. Another example is the so-called collaboration network among scientific researchers. A collaboration relation forms between two researchers if they have co-authored a paper, but there are other types of relations or proximity that are informative for telling whether they will have collaboration in the future, e.g., whether they have attended the same conference, whether they have cited the same papers, or whether they have published papers with similar keywords.

This work was done when Zhengdong Lu was a postdoctoral researcher in ICES, University of Texas at Austin

In this paper, we focus on exploiting the topological information for a basic computational problem underlying social network evolution—the link prediction problem. Given snapshots of an evolving social network from time 1 to t, we seek to accurately predict the edges that will be added to the network during the interval from time t to a future time $t+1$. This problem is also related to uncovering hidden links in a network, which can be considered as a missing value problem for entries of the graph's adjacency matrix. Various unsupervised [1], [11], [14] and supervised [7], [12], [20] models have been proposed to address these problems, assuming there is one network available. However, there is little work on incorporating auxiliary sources in link prediction. Kashima et al. [10] introduced a *link propagation* framework to exploit multiple types of links between vertices. However, this work is largely unsupervised, and only works for missing link recovery in static networks. In constrast, we aim to find a predictive model for evolving networks and learn the dynamics with a supervised framework.

Overview: In Section II we give some background on statistical network models for link prediction. In Section III we present a dynamic model for network evolution with multiple auxiliary networks. In Section IV we show how to fit the model and make predictions using historical and auxiliary network information. Then in Section V, we discuss regularization strategies to trim these features with supervision. In Section VI we present experimental results, which show that supervision with multiple sources outperform single source methods.

Contribution: Our contribution in this paper is twofold: (1) a supervised learning framework that can effectively and efficiently learn the dynamics of social networks in the presence of auxiliary networks; and (2) a feature design scheme for constructing a rich variety of path-based features using multiple sources, and an effective feature selection strategy based on structured sparsity. A longer and more detailed version of this paper is the technical report [15].

II. Background

Our work is motivated by the long thread of work in statistical modeling of static and dynamical social networks, as well as the work on heuristic but practically effective unsupervised link prediction models. We now give a brief introduction to the two contrasting threads of work, with emphasis on the parts that are directly related to our model.

A. Unsupervised Link Prediction

Various models have been proposed for link prediction, which, as summarized in [14], generally fall into three categories. The first category has methods based on vertex neighborhoods, including Common neighbors [17], Adamic/Adar [1], Preferential Attachment [3], [16]. The second category has methods based on the ensemble of all paths, including Katz [11] and Hitting Time, while the third category includes high level approaches, such as matrix factorization and clustering. All of these methods rely on a predictive score function for all entries to get a ranking of edges that are likely to occur.

We will elaborate on the Katz measure [11], for its modeling simplicity and its wide success in practice. More importantly, as we will show later, Katz is closely related to the proposed framework and provides a justification for our work. The Katz measure directly sums over the collection of paths, exponentially damped by length to count short paths more heavily, leading to the β-parametrized measure:

$$\mathsf{score}_{\mathsf{Katz}}(i,j) = \sum_{l=1}^{\infty} \beta^l |\mathsf{path}_{i,j}^{\langle l \rangle}|, \tag{1}$$

where $\mathsf{path}_{i,j}^{\langle l \rangle}$ is the set of all length-l paths from vertex i to j. With the adjacency matrix $\mathbf{A}$ of the graph, one can verify that for $\beta < 1/\|\mathbf{A}\|_2$, the score matrix is given by

$$\mathsf{score}_{\mathsf{Katz}} = \sum_{l=1}^{\infty} \beta^l \mathbf{A}^l = (\mathbf{I} - \beta\mathbf{A})^{-1} - \mathbf{I}. \tag{2}$$

When inverting $(\mathbf{I} - \beta\mathbf{A})$ becomes too expensive, one can choose to stop after paths of length $l_{\max}$ in (2) to get the truncated Katz score:

$$\mathsf{score}_{\mathsf{tKatz}} = \sum_{l=1}^{l_{\max}} \beta^l \mathbf{A}^l. \tag{3}$$

It is easy to see that truncated Katz becomes a good approximation of Katz when β is small enough. In practice truncated Katz often outperforms Katz for link prediction.

B. Dynamical Random Graph Models

In contrast to the heuristic methods for link prediction, there is a group of models devoted to study the intrinsic mechanism governing the topological changes of networks over time. For a series of snapshots $\mathbf{A}^{(t)}$ of a network at different time steps, a statistical model for network evolution can be estimated. Usually it is assumed that the underlying states of the social network follow a stationary Markov process, and the statistical modeling therefore boils down to the modeling of a transition probability, $\mathcal{P}(\mathbf{A}^{(t+1)}|\mathbf{A}^{(t)})$. For example, Robins and Pattison [18] and Hanneke et al. [5] have studied a family of models of network dynamics over discrete time steps, with an exponential random graph model (ERGM) describing the transition probability

$$\mathcal{P}(\mathbf{A}^{(t+1)}|\mathbf{A}^{(t)}) \propto e^{\langle \boldsymbol{\theta}, \Phi(\mathbf{A}^{(t+1)}|\mathbf{A}^{(t)}) \rangle},$$

where $\Phi(\mathbf{A}^{(t+1)}|\mathbf{A}^{(t)})$ denotes the vector of sufficient statistics and $\boldsymbol{\theta}$ denotes the vector with natural parameters.

III. Learning The Dynamics

A. Dynamics of Social Network Evolutions

In this section we will describe in detail our model for the dynamics of social network evolution in the presence of multiple auxiliary networks. For simplicity, we only consider the undirected unweighted graph, which implies that all relationships are mutual and weighted equally. It is straightforward to extend these models to directed and/or weighted graphs.

Suppose we observe snapshots of an evolving social network from time $\tau = 1, \cdots, t$, with the corresponding adjacency matrices $\mathbf{A}^{(1)}, \cdots, \mathbf{A}^{(t)}$. The task is to find a prediction model for $\mathbf{A}^{(t+1)}$. We assume no vertices are added or removed during the evolution, but edges could form and/or disappear at each time step. In addition to $\mathbf{A}^{(1)}, \cdots, \mathbf{A}^{(t)}$, we have snapshots of a network from heterogeneous but related source denoted $\mathbf{B}^{(1)}, \cdots, \mathbf{B}^{(t)}$.

We start describing our model with the following two assumptions:

$$\mathcal{P}(\mathbf{A}^{(\tau)}|\mathbf{A}^{(1:\tau-1)}, \mathbf{B}^{(1:\tau-1)}) = \mathcal{P}(\mathbf{A}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) \quad \text{(A1)}$$

$$\mathcal{P}(\mathbf{A}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) = \prod_{i,j} \mathcal{P}(A_{ij}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) \quad \text{(A2)}$$

In (A1) we assume the evolution of $\mathbf{A}^{(\tau)}$ is a Markov process, where the probability of network state $\mathbf{A}^{(\tau)}$ is governed jointly by $\mathbf{A}^{(\tau-1)}$ and $\mathbf{B}^{(\tau-1)}$. In (A2) we assume that $\mathcal{P}(\mathbf{A}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ fully factorizes. Both assumptions are made for modeling tractability. We can loosen (A1) to include the dependence of current snapshot on m previous time steps yielding $\mathcal{P}(\mathbf{A}^{(\tau)}|\mathbf{A}^{(\tau-m:\tau-1)}, \mathbf{B}^{(\tau-m:\tau-1)})$ which makes more sense in a collaboration network, since the underlying relationship may not appear as observable events (e.g., co-authoring a paper) in the duration time of a certain snapshot. For notational simplicity, we will describe the case for $m = 1$, while discussing the case of multiple retrospective steps only when the extension is not trivial.

B. Probabilistic Model

We generalize the ERGM [5], [18] to describe the transition probability

$$\mathcal{P}(\mathbf{A}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) \propto e^{\langle \boldsymbol{\theta}, \Phi(\mathbf{A}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) \rangle},$$

where $\Phi(\mathbf{A}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ is the "sufficient statistics" associated with $\mathbf{A}^{(\tau)}$ conditioned on the historical states $\mathbf{A}^{(\tau-1)}$ and $\mathbf{B}^{(\tau-1)}$, and $\boldsymbol{\theta} = [\theta_1, \cdots, \theta_K]^\top$ denotes the natural parameters to be learned. Using (A2) we can simplify and model the probability for each link $\mathcal{P}(A_{ij}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ as

$$\frac{1}{Z_{ij}(\boldsymbol{\theta}, \mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})} e^{\sum_{k=1}^{K} \theta_k \phi_k(A_{ij}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})},$$

where $\phi_k(A_{ij}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ is the k^{th} statistic associated with pair (i,j), and $Z_{ij}(\boldsymbol{\theta}, \mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ is the normalization constant. Since we are modeling the presence/absence of the link $A_{ij}^{(\tau)}$, one natural choice of the feature ϕ_k is

$$\phi_k(A_{ij}^{(\tau)}|\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) = A_{ij}^{(\tau)} \cdot g_{k,ij}(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) \tag{4}$$

where $A_{ij}^{(\tau)} \in \{0,1\}$, and $g_{k,ij}(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ is the k^{th} feature extracted from previous snapshot $\mathbf{A}^{(\tau-1)}$ and $\mathbf{B}^{(\tau-1)}$ for pair (i,j). Usually $g_{k,ij}(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ summarizes a certain local property from $\mathbf{A}^{(\tau-1)}$ and $\mathbf{B}^{(\tau-1)}$ of interest to the generation of link (i,j), e.g. $g_{k,ij}(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) = \sum_n B_{in}^{(\tau-1)} B_{nj}^{(\tau-1)}$ gives the number of common neighbors in $\mathbf{B}^{(\tau-1)}$. Note that we assume all the links are formed based on the same local dynamics. This implies the same parameter θ_k for all (i,j) in each $g_{k,ij}$.

Introducing the latent potential

$$p_{\boldsymbol{\theta}}(i,j) = \sum_{k=1}^{K} \theta_k g_{k,ij}(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}), \tag{5}$$

it follows from (4) that the model is a logistic regression

$$\mathcal{P}(A_{ij}^{(\tau)} = 1 | \mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}) = \frac{e^{p_{\boldsymbol{\theta}}(i,j)}}{1 + e^{p_{\boldsymbol{\theta}}(i,j)}}. \tag{6}$$

This implies that the probability of having a link formed beteween i and j at time τ is governed by (5), which is a linear combination of $g_{k,ij}(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)})$ from snapshot $\tau - 1$.

IV. Model Fitting

Suppose we want to predict the links in snapshot $\mathbf{A}^{(t+1)}$, and have as observations the historical snapshots of the main network $\mathbf{A}^{(1)}, \cdots, \mathbf{A}^{(t)}$ as well as auxiliary network $\mathbf{B}^{(1)}, \cdots, \mathbf{B}^{(t)}$. Extension to more than one auxiliary network is straightforward.

A. Model Fitting and Prediction

The task of model fitting is to determine $\boldsymbol{\theta}$ from observations $\mathbf{A}^{(1)}, \cdots, \mathbf{A}^{(t)}$ and $\mathbf{B}^{(1)}, \cdots, \mathbf{B}^{(t)}$, and predict the links in $\mathbf{A}^{(t+1)}$. The problem we focus on is the formation of *new* links in the main network, i.e. links that do not appear in the retrospective steps. Let $\mathcal{E}_\tau$ denote the set of links in snapshot τ, let $\mathcal{N}_\tau$ denote new links formed in time interval $[\tau, \tau+1]$, and $\mathcal{Z}_\tau$ denote the complement of $\mathcal{E}_\tau \cup \mathcal{N}_\tau$. Clearly $\mathcal{E}_{\tau+1} = \mathcal{E}_\tau \cup \mathcal{N}_\tau$ while $\mathcal{E}_\tau \cup \mathcal{N}_\tau \cup \mathcal{Z}_\tau$ is the set of all possible pairs. The task is to find $\boldsymbol{\theta} = [\theta_1, \cdots, \theta_n]^\top$ that maximize the likelihood of the observed new links from $\tau = 2, \cdots, t$;

$$\boldsymbol{\theta}^* = \arg\max_{\boldsymbol{\theta}} \prod_{\tau=2}^{t} \Big(\prod_{i,j \in \mathcal{N}_\tau \cup \mathcal{Z}_\tau} \mathcal{P}\big(A_{ij}^{(\tau)} | \mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}\big) \Big).$$

With the model in (6), we minimize the negative log likelihood

$$L(\boldsymbol{\theta}) = -\sum_{\tau=2}^{t} \Bigg(\sum_{i,j \in \mathcal{N}_\tau \cup \mathcal{Z}_\tau} \sum_{k=1}^{K} \theta_k g_{k,ij}\big(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}\big) - \sum_{i,j \in \mathcal{Z}_\tau \cup \mathcal{N}_\tau} \log\Big(1 + e^{\sum_{k=1}^{K} \theta_k g_{k,ij}\left(\mathbf{A}^{(\tau-1)}, \mathbf{B}^{(\tau-1)}\right)}\Big) \Bigg),$$

which is convex in $\boldsymbol{\theta}$ and various optimization routines can be used to get a global minimum.

Once the optimal parameter $\boldsymbol{\theta}^*$ is obtained, the prediction of $\mathbf{A}^{(t+1)}$ can be carried out using the potential in (5)

$$\mathsf{score}_{\boldsymbol{\theta}^*}(i,j) = p_{\boldsymbol{\theta}^*}(i,j) = \sum_k \theta_k^* g_{k,ij}(\mathbf{A}^{(t)}, \mathbf{B}^{(t)}), \tag{7}$$

which can also be justified since $\mathsf{score}_{\boldsymbol{\theta}^*}(i,j)$ is also the log odds ratio $\log \frac{\mathcal{P}(A_{ij}^{(t+1)}=1|\mathbf{A}^{(t)}, \mathbf{B}^{(t)})}{\mathcal{P}(A_{ij}^{(t+1)}=0|\mathbf{A}^{(t)}, \mathbf{B}^{(t)})}$. We use the score function (instead of the actual probability) in link prediction if only the ranking of the predicted links are needed.

B. Square Loss Surrogate

The logistic regression model is still computationally expensive for many real-world applications. Here we show that the simple square loss can be used as a cheap and effective surrogate for the logistic regression objective.

It is easy to see that the potential $p_{\boldsymbol{\theta}}(i,j)$ in (5) is positive when the probability of $A_{ij}^{(t)} = 1$ ("link") is greater than $A_{ij}^{(t)} = 0$ ("no link"), and vice versa. A simple heuristic of fitting the scores of "linked" pairs to $+1$ and "not-linked" pairs to -1 leads to the surrogate objective function for $\boldsymbol{\theta}$,

$$L_{\text{lsq}}(\boldsymbol{\theta}) = \sum_{i,j \in \mathcal{N}_t \cup \mathcal{Z}_t} \|p_{\boldsymbol{\theta}}(i,j) - \text{sign}(A_{ij}^{(t)} - 0.5)\|^2 \tag{8}$$

where $\text{sign}(\cdot)$ returns the sign ($+1$ or -1) of the input argument. Equation (8) can be rearranged into the following matrix form $L_{\text{lsq}}(\boldsymbol{\theta}) = \|\mathbf{S}\boldsymbol{\theta} - \mathbf{y}\|_2^2$, where the number of rows in $\mathbf{S}$ is $|\mathcal{N}_t \cup \mathcal{Z}_t|$ and $\mathbf{y}$ is a target vector with $+1$ or -1.

C. Path-counting Features

The features $g_{k,ij}(\mathbf{A}^{(\tau)}, \mathbf{B}^{(\tau)})$ in (4) could take a great variety of knowledge about the possible links between vertices i and j. These could be features based on the topological properties of the network, e.g. Preferential Attachment [3], [16] and Adamic/Adar [1], clustering coefficients [9], [17], or based on demographical and other kind of informations about the vertices. In this paper, we are particularly interested in path-counting features, since it has been shown to be a simple but informative measure of proximity between vertices. Also, our supervised model with path-counting features are natural extension to popular unsupervised models such as Katz measure (and hence nearest neighbors).

The path-counting features for a single graph/network source are simply the number of length-l paths with $l = 1, \cdots, l_{\max}$. With any unweighted graph the l^{th} pairwise feature on any snapshot τ can be computed using

$$g_{l,ij}(\mathbf{A}^{(\tau)}) = |\mathsf{path}_{i,j}^{\langle l \rangle}| = [(\mathbf{A}^{(\tau)})^l]_{ij}, \tag{9}$$

or simply $\mathbf{G}_l = (\mathbf{A}^{(\tau)})^l$ for all pairs (i,j). In this paper we use (9) for weighted graphs also. It is easy to verify that the features corresponding to paths with length $0, 1, \cdots, l_{\max}$, i.e. $\mathbf{G}_0, \mathbf{G}_1, \cdots, \mathbf{G}_{l_{\max}}$, are given by terms in the polynomial $(\mathbf{A}^{(\tau)} + \mathbf{I})^{l_{\max}}$, where the identity matrix $\mathbf{I}$ is for the "length-0" paths, which also servers as an offset in the logistic regression.

With multiple sources, we will have a much richer set of paths if we allow cross routes between networks from different sources. The best way to understand this is through the concept of a *multigraph* [6], which allows more than one edge between two vertices. Suppose we have a multigraph $\mathcal{M}$, between any two vertices there could be an edge from $\mathbf{A}^{(\tau)}$ and an edge from $\mathbf{B}^{(\tau)}$. For description convenience, we can have the two

kind of edges color-coded, "A" colored versus "B" colored. This results in three types of paths in $\mathcal{M}$:

1) Pure color paths with only edges of $\mathbf{A}$, e.g., $i \xrightarrow{\mathbf{A}} j \xrightarrow{\mathbf{A}} k$;
2) Pure color paths with only edges of $\mathbf{B}$, e.g., $i \xrightarrow{\mathbf{B}} j \xrightarrow{\mathbf{B}} k$;
3) Hybrid color paths with edges of both $\mathbf{A}$ and $\mathbf{B}$, e.g., $i \xrightarrow{\mathbf{B}} j \xrightarrow{\mathbf{A}} k$, as illustrated in Figure 1.

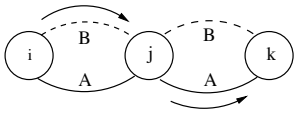

Fig. 1. Example of a hybrid color path, $i \xrightarrow{\mathbf{B}} j \xrightarrow{\mathbf{A}} k$

The counting of the type-1 and type-2 paths with length-l are simply given as $(\mathbf{A}^{(\tau)})^l$ and $(\mathbf{B}^{(\tau)})^l$. A simple extension to the path counting features in the single source case would be to use pure color paths only, i.e. type-1 and type-2. Counting the type-3 paths is more complicated since we want to distinguish paths between two vertices not only by their lengths, but also by color of edges in the path. For example, we may want

$$\text{path-1: } i \xrightarrow{\mathbf{B}} i' \xrightarrow{\mathbf{A}} j' \xrightarrow{\mathbf{A}} j \quad \text{and} \quad \text{path-2: } i \xrightarrow{\mathbf{B}} i' \xrightarrow{\mathbf{B}} j' \xrightarrow{\mathbf{A}} j$$

to have different weights, because edges of $\mathbf{A}$ could be more informative than edges of $\mathbf{B}$ in predicting the links in $\mathbf{A}^{(\tau+1)}$. In the supervised learning framework, we wish to have a feature for each particular color combination. Considering only undirected graphs, we require $g_{k,ij}(\cdot) = g_{k,ji}(\cdot)$, for any pair (i, j) and any k, and therefore count paths with reverse color patterns as the same. One can verify that the number of paths up to length $l_{\max}$ from all combination types are given by terms in the following matrix polynomial

$$(\mathbf{I} + \mathbf{A}^{(\tau)} + \mathbf{B}^{(\tau)})^{l_{\max}}, \tag{10}$$

and, say, paths with pattern " $\circ \xrightarrow{\mathbf{B}} \circ \xrightarrow{\mathbf{B}} \circ \xrightarrow{\mathbf{A}} \circ$ " can be counted efficiently using the matrix $\mathbf{B}^{(\tau)}\mathbf{B}^{(\tau)}\mathbf{A}^{(\tau)}$.

With multiple auxiliary sources, denoted $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$, $\cdots$, the features in matrix form are given by terms in the polynomial

$$(\mathbf{I} + \mathbf{A}^{(\tau)} + \mathbf{B}^{(\tau)} + \mathbf{C}^{(\tau)} + \mathbf{D}^{(\tau)} + \cdots)^{l_{\max}}. \tag{11}$$

In practice, we may consider more than one retrospective step, and hence several separate multigraphs, each corresponding to a time step. To control the number of features, we do not allow any path combination between different time steps. Therefore in the case of $k+1$ retrospective steps, the features set for predicting $\mathbf{A}^{(\tau+1)}$ are terms from $(\mathbf{I} + \mathbf{A}^{(\tau-k)} + \mathbf{B}^{(\tau-k)})^{l_{\max}}, \cdots, (\mathbf{I} + \mathbf{A}^{(\tau)} + \mathbf{B}^{(\tau)})^{l_{\max}}$.

D. Generalization to the Katz Measure

We now show that the score function (7) generalizes popular unsupervised models in several ways when using the path-counting features. From Section IV-C, when only considering the feature from the main network, the feature associated with length-l paths is $\mathbf{G}_l(\mathbf{A}^{(t)}) = (\mathbf{A}^{(t)})^l$ in matrix form. The score function therefore becomes

$$\text{score} = \sum_{l=1}^{l_{\max}} \theta_l^* \mathbf{G}_l(\mathbf{A}^{(t)}) = \sum_{l=1}^{l_{\max}} \theta_l^* (\mathbf{A}^{(t)})^l. \tag{12}$$

Clearly (12) generalizes the truncated Katz measure by replacing the exponential damping factor β^l in (3) with a more general parameter θ_l, and hence introduces more modeling flexibility. With auxiliary sources like $\mathbf{B}^{(t)}$, the feature set will get much richer and the score function will have additional power terms $(\mathbf{B}^{(t)})^l$, corresponding to the pure color paths, and mixed terms, e.g. $\mathbf{B}^{(t)}\mathbf{A}^{(t)}\mathbf{B}^{(t)}$, corresponding to hybrid color paths. Both types of terms could lend substantial prediction capability to the prediction model. Moreover, we may consider more than one retrospective step, which generalizes the truncated Katz measure even more.

V. Regularization

The path-counting features for multiple sources yield a rich set of features. In fact, with c different sources, the number of features associated with length-l paths is exponential in l. Thus the model fitting is prone to over-fitting as the observations are extremely noisy, and the dimensionality of the feature space is high due to the multiple sources with potentially irrelevant features. When predicting with multiple sources, it could be the case that some auxiliary sources do not contain information valuable for prediction. In addition one particular path pattern or feature may not be useful even though the component source is informative. These characteristics of our problem call for a sensible feature selection strategy, and in particular sparsity-promoting regularization schemes. As is shown in [15, Section 6], the least squares objective is a rather effective surrogate of the logistic regression with much lower complexity. So, in this section we will focus only on the least squares objective, and consider the ℓ_1 and *hierarchical sparsity* (HS) regularization on $\boldsymbol{\theta}$ to filter out irrelevant features.

Lasso: Using the ℓ_1 regularization we get the Lasso regression model [8] with the objective function

$$L_{\text{Lasso}}(\boldsymbol{\theta}) = \|\mathbf{S}\boldsymbol{\theta} - \mathbf{y}\|_2^2 + \lambda\|\boldsymbol{\theta}\|_1, \tag{13}$$

where λ controls the sparsity in $\boldsymbol{\theta}$. This regularization can be applied to both pure color paths and hybrid paths.

Hierarchical Sparsity: The ℓ_1 regularization in Lasso is flat in the sense that it puts uniform regularization on all features. More sophisticated group-Lasso based regularization designs consider the hierarchical structure inherent to the task [2], [21]. The structure in our problem lies in the way the composite paths are constructed. We wish that if a path pattern (or feature) ω is knocked out, all the path patterns containing ω as sub-pattern should receive zero weight too. This relation between features can be fully expressed as a *directed acyclic graph* (DAG). In Figure 2 we illustrate how two sources are mixed together to form a DAG up to power $l = 3$. For example, if the path pattern $\circ \xrightarrow{\mathbf{B}} \circ \xrightarrow{\mathbf{B}} \circ$ (or equivalently the feature in matrix form $\mathbf{BB}$) has zero weight, we wish that all the features containing $\mathbf{BB}$, e.g., $\mathbf{BBA}$, $\mathbf{BBB}$ and $\mathbf{ABB}$, to be excluded from the variable set. Consider the dashed circles in Figure 2. The remaining set of features becomes $\mathbf{A}$, $\mathbf{B}$, $\mathbf{AA}$, $\mathbf{AB}$, $\mathbf{BA}$, $\mathbf{AAA}$, $\mathbf{AAB}$, $\mathbf{ABA}$, $\mathbf{BAB}$.

To enforce this kind of feature preference, we can use the composite absolute norm introduced by Zhao et al. [21].

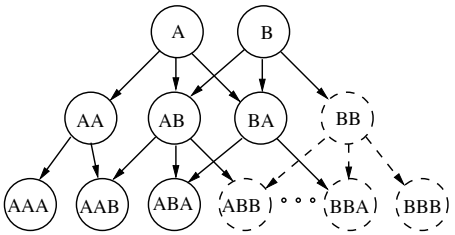

Fig. 2. An illustration of the hierarchical sparsity from two sources.

This is implemented through group Lasso with overlapping groups. With a DAG $(\mathcal{V}, \mathcal{E})$, a group $\mathcal{G}_v \subset \mathcal{V}$ with "root" node v contains v and all of its offsprings, and we denote the set of all such groups $\mathcal{G} = \{v \cup \text{all offsprings of } v | v \in \mathcal{V}\}$. The ℓ_∞ norm for each group $g \in \mathcal{G}$ is defined as $\|\boldsymbol{\theta}_g\|_\infty = \max_{v \in g} |\boldsymbol{\theta}_v|$ and the composite norm is simply $\|\boldsymbol{\theta}_\mathcal{G}\|_c = \sum_{g \in \mathcal{G}} \|\boldsymbol{\theta}_g\|_\infty$. The cost function with this hierarchically structured penalty is defined as

$$L_{\text{HS}}(\boldsymbol{\theta}) = \|\mathbf{S}\boldsymbol{\theta} - \mathbf{y}\|_2^2 + \lambda \|\boldsymbol{\theta}_\mathcal{G}\|_c. \tag{14}$$

Note that all features comply with the sparsity structure as well as the hierarchy of the DAG. Note also that this sparsity structure is promoted but not enforced. In practice undesired feature combinations could still appear, especially when the regularization parameter λ is small.

VI. Experimental Results

A. The Data Sets

We adopted three real-world data sets from the scientific publication domain: (1) arXiv publications from 1992 to 2003 in high energy physics (theory). We formed four snapshots from non-overlapping intervals of three years; (2) CiteSeer publications from 1995 to 2003. We divided this data set into three subsets: CS-1 with publications from 1995 to 1997; CS-2 with publications from 1998 to 2000; and CS-3 with publications from 2001 to 2003. For each subset, we formed three snapshots based on the publications of one year. (3) SIAM (Society of Industrial and Applied Mathematics) publications from 1999 to 2004 covering 11 journals and proceedings.

Several networks are formed with the authors being the vertices: co-authorship $\mathbf{A}$: $A_{ij} = 1$ if author i and author j co-authored a paper; co-citation $\mathbf{B}$: $B_{ij} = 1$ if author i and author j have cited same papers; co-reference $\mathbf{C}$: $C_{ij} = 1$ if papers by author i and author j are cited by the same paper; text similarity $\mathbf{D}$: $D_{ij} = 1$ if the cosine similarity between papers (represented with the "bag-of-words" model) published by author i and author j is over a threshold. We will predict on co-authorship $\mathbf{A}$ with other networks $\mathbf{B}$, $\mathbf{C}$, $\mathbf{D}$ as auxiliary data. Results using networks $\mathbf{B}$ and $\mathbf{C}$ as targets are similar in spirit to the results presented here. Some statistics of each data set are given in Table I.

B. Models

We use the following models in our experiments.

TABLE I
Some statistics of the arXiv, CiteSeer (CS-1, CS-2, CS-3) and SIAM data sets. Here "core" denotes the number of authors with at least one publication in any given snapshot during training. "train" and "test"denote the number of links.

data set		HepTh	CS-1	CS-2	CS-3	SIAM
core		1381	2321	2448	1182	6891
train	**A**	14507	9683	13634	8703	5528
	B	61674	35509	41956	21781	5431
	C	22075	11269	16294	11988	5124
	D	39596	19776	30931	20283	6504
test	**A**	6788	6809	9609	6262	2764
	B	20523	17067	19464	12764	2715
	C	15459	6591	10100	6297	2562
	D	11576	12643	18799	9860	2586

Unsupervised Models: Katz-s Katz measure based on a single source ($\mathbf{A}$); Katz-c Katz measure based on all sources, ($\mathbf{A}+\mathbf{B}+\mathbf{C}+\mathbf{D}$); and similarly tKatz-s and tKatz-c with the truncated Katz measure. We used an optimal β in all cases.

Supervised Models: The supervised learning models are SL-s with single source; SL-p with only pure color paths; and SL-h with hybrid color paths. We may also have ℓ_1 or hierarchical structured regularization, which will for example be denoted with SL-h(l1) or SL-h(hs), respectively.

We have conducted extensive tests of our prediction model on a variety of collaboration networks. We use the Katz and truncated Katz as the representatives of unsupervised models, because of their overall good performance in link prediction [14] and their close relation to our path-counting features. Within the proposed supervised framework, we also intend to compare ones with single source and multiple sources, as well as models with different feature designs and regularization.

C. Experimental Settings

For any data set with $t+1$ snapshots, we use $\mathbf{A}^{(1)}$ to $\mathbf{A}^{(t)}$ for training, and $\mathbf{A}^{(t+1)}$ for testing. Once a model parameter $\boldsymbol{\theta}$ is learned, we use a corresponding score function to predict new links in $\mathbf{A}^{(t+1)}$. We select $|\mathcal{N}_t|$ "feasible" pairs with highest scores as our predictions of new links for time $t+1$ and calculated the proportion of true links in terms of percentage.

D. Results and Analysis

The performance of the above mentioned models are reported in Table II. Basically we achieve improved performance with supervised models and multiple sources.

1) The Role of Supervision: The clear message from Table II is "supervision helps in link prediction". In particular, The single-source supervised model SL-s, which has more parameters, is overall better than unsupervised counterparts, e.g. tKatz-s. Supervision also helps in learning a proper way to synthesize information from multiple auxiliary sources, as in SL-p and SL-h and their regularized versions. This turns out to be much more effective than the naive way to combine different sources, as in Katz-c and tKatz-c.

2) The Role of Multiple Sources: Multiple auxiliary sources greatly help the prediction on the target network in the

TABLE II
LINK PREDICTION RESULTS IN TERMS OF PRECISION ON THREE CITESEER SUBSETS, ARXIV-HEPTH DATA SET AND SIAM DATA SET.

	$l_{\max}$	CS-1	CS-2	CS-3	arXiv	SIAM
KATZ-S		19.3	13.2	20.4	2.26	32.6
KATZ-C		21.1	14.8	15.3	0.38	34.6
TKATZ-S	2	19.5	16.4	21.6	1.41	41.1
	4	19.3	13.2	20.4	2.26	32.2
TKATZ-C	2	21.1	14.8	15.0	0.38	32.1
	4	21.1	14.8	15.0	0.38	32.0
SL-S	2	19.6	16.8	19.2	2.30	41.5
	4	24.1	21.2	24.6	**2.83**	41.5
SL-P	2	19.3	16.1	15.1	1.95	49.2
	4	24.1	23.4	22.5	2.12	50.7
SL-P(L1)	2	19.3	16.1	15.1	2.08	49.2
	4	24.1	23.4	22.5	2.21	50.6
SL-H	2	32.5	27.3	34.2	2.08	50.9
	4	32.6	27.3	34.3	2.48	52.3
SL-H(L1)	2	32.5	27.3	34.0	2.48	50.8
	4	32.5	27.3	34.0	2.12	52.3
SL-H(HS)	2	**33.9**	27.7	**34.9**	1.95	51.1
	4	33.4	**27.9**	34.2	2.48	**52.6**

supervised framework. On all data sets, except arXiv, multiple-source supervised models (SL-P, SL-H, and their regularized versions) are clearly better than the single-source supervised models SL-S, especially when the information from auxiliary sources are encoded in a richer feature set. From Table II, it is not rare that a naive combination of network sources in KATZ-C and TKATZ-C yields inferior performance than the single source unsupervised model. On arXiv we observe over 80% decrease of accuracy when using multiple sources in an unsupervised way. We also argue that the supervised framework can effectively discriminate useful auxiliary sources and features from the irrelevant and distractive ones, as is seen on arXiv with multiple-source supervised models.

3) Feature Design and Regularization: We are clearly benefiting from the rich set of features. It can be seen by comparing SL-P with SL-H. For all prediction tasks, SL-H performs better than SL-P, showing the predictive power of cross-source paths in feature design. Moreover, the regularization promoting structured sparsity helps to further improve the accuracy. For all the tasks SL-H(HS) is better than SL-H and SL-H(L1).

VII. CONCLUSION AND DISCUSSION

In this paper, we have proposed a novel and general framework for supervised link prediction. Our model can effectively and efficiently learn the network dynamics from a time series of network snapshots, and therefore improve the link prediction accuracy. In addition, multiple graphs over the same set of vertices but from different sources can be naturally incorporated into the framework. We have performed extensive set of experiments on real-world data sets. The experimental results confirm that prediction accuracy is improved using supervision and multiple sources of information.

Despite the empirical success of the proposed model, a few directions remain to be explored. (1) We have not exploited the ability of our models to take features other than path counts. As suggested in [9], [13], many features and other network characteristics can be informative for link formation, most of which can be readily used in our framework. (2) It is still unclear what probabilistic model is most appropriate for the predictive modeling of links. For example, we could adopt the Prackett-Luce ranking model [4] to describe the latent mechanism of link generation, and view all the new links as observed to be top-ranked. (3) Many social networks are massive in size and therefore pose a scalability issue [19]. We plan to address and conduct research on all these issues.

ACKNOWLEDGEMENTS

This research was supported by NSF grants CCF-0916309 and IIS-0713142. Zhengdong Lu is supported by the ICES postdoctoral fellowship from UT Austin. Berkant Savas is supported by the Swedish Research Council. We would like to thank SIAM for providing the SIAM data set and Sandia National Lab for providing the CiteSeer data set.

REFERENCES

[1] L. Adamic and E. Adar. Friends and neighbors on the web. *Social networks*, 25:211–230, 2003.
[2] F. Bach. Exploring large feature spaces with hierarchical multiple kernel learning. In *NIPS*, 2008.
[3] A. Barabasi, H. Jeong, Z. Neda, E. Ravasz, A. Schubert, and T. Vicsek. Evolution of the social network of scientific collaborations. *Physica A: Statistical Mechanics and its Applications*, 311:590–614, 2002.
[4] J. Guiver and E. Snelson. Bayesian inference for Plackett-Luce ranking models. In *ICML*, 2009.
[5] S. Hanneke, W. Fu, and E. Xing. Discrete temporal models of social networks. *Electronic Journal of Statistics*, 4(2010) 585–605, 2010.
[6] F. Harary. *Graph Theory*. Westview Press, 1994.
[7] M. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *Workshop on Link Analysis, Counterterrorism and Security (SDM)*, 2006.
[8] T. Hastie, R. Tibshirani, and J. Friedman. *The Elements of Statistical Learning*. Springer-Verlag, New York, 2001.
[9] Z. Huang. Link prediction based on graph topology: The predictive value of the generalized clustering coefficient. In *Workshop on Link Analysis (KDD)*, 2006.
[10] H. Kashima, T. Kato, Y. Yamanishi, M. Sugiyama, and K. Tsuda. Link propagation: A fast semi-supervised learning algorithm for link prediction. In *SDM*, 2009.
[11] L. Katz. A new status index derived from sociometric analysis. *Psychometrika*, 18:39–43, 1953.
[12] J. Kunegis and A. Lommatzsch. Learning spectral graph transformations for link prediction. In *ICML*, 2009.
[13] J. Leskovec, L. Backstrom, R. Kumar, and A. Tomkins. Microscopic evolution of social networks. In *KDD*, 2008.
[14] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.
[15] Z. Lu, B. Savas, W. Tang, and I. S. Dhillon. Supervised link prediction using multiple sources. Technical Report TR–10–35, Department of Computer Science, UT Austin, 2010.
[16] M. Mitzenmacher. A brief history of generative models for power law and lognormal distributions. *Internet Mathematics*, 1(2):226–251, 2004.
[17] M. Newman. Clustering and preferential attachment in growing networks. *Physical Review E*, 64(025102), 2001.
[18] G. Robins and P. Pattison. Random graph models for temporal processes in social networks. *Journal of Mathematical Sociology*, 25:5–41, 2001.
[19] H. H. Song, T. W. Cho, V. Dave, Y. Zhang, and L. Qiu. Scalable proximity estimation and link prediction in online social networks. In *IMC*, 2009.
[20] B. Taskar, M.-F. Wong, P. Abbeel, and D. Koller. Link prediction in relational data. In *NIPS*, 2003.
[21] P. Zhao, G. Rocha, and B. Yu. Grouped and hierarchical model selection through composite absolute penalties. *Annals of Statistics*, 37:3468–3497, 2009.