2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Max-margin Latent Feature Relational Models for Entity-Attribute Networks

Fei Xia[†‡] Ning Chen[†] Jun Zhu[†] Aonan Zhang[†] Xiaoming Jin[‡]
[†]Dept. of CS & T, TNList Lab, State Key Lab of ITS., [‡] School of Software,
Tsinghua University, Beijing 100084, China.
{xia-f09@mails, ningchen@, dcszj@, zan12@mails, xmjin@}.tsinghua.edu.cn

***Abstract*—Link prediction is a fundamental task in statistical analysis of network data. Though much research has concentrated on predicting entity-entity relationships in homogeneous networks, it has attracted increasing attentions to predict relationships in heterogeneous networks, which consist of multiple types of nodes and relational links. Existing work on heterogeneous network link prediction mainly focuses on using input features that are explicitly extracted by humans. This paper presents an approach to automatically learn latent features from partially observed heterogeneous networks, with a particular focus on entity-attribute networks (EANs), and making predictions for unseen pairs. To make the latent features discriminative, we adopt the max-margin idea under the framework of maximum entropy discrimination (MED). Our maximum entropy discrimination joint relational model (MED-JRM) can jointly predict entity-entity relationships as well as the missing attributes of entities in EANs. Experimental results on several real networks demonstrate that our model has improved performance over state-of-the-art homogeneous and heterogeneous network link prediction algorithms.**

I. Introduction

As the availability and scope of network data increase, link prediction [20] as a fundamental task of statistical network analysis has attracted many research attentions. Link prediction methods can find applications in recommendation systems [13], information retrieval [1], marketing [25], bioinformatics [30], and so on.

Link prediction is typically formulated as a task of predicting unseen links between entities given partially observed link information [17]. Many methods have been developed, including random walk methods [2], [28], [16], probabilistic models [29], and the methods that formalize the prediction task as a classification problem [10], [19]. All these methods rely on human designed features. Though the explicit feature based methods can work effectively, they may suffer from some problems [2]. First, deciding which features to use and extracting good features can be notorious and may require extensive expert knowledge. Second, some explicit features are domain-specific and not generalizable, e.g., some social science knowledge for predicting links in social networks may not be suitable for document networks (e.g., paper citation networks). Finally, it is difficult for humans to design or perceive features accounting for the high-order interactions hiding in the complex networks. To deal with such issues and better capture the properties of networks, latent space models have been widely studied to automatically learn good features. Representative work includes latent feature relational models (LFRMs) [12], [11] and low-rank matrix factorization [23]. Recently, improvements on LFRMs have been obtained in various aspects. For example, to avoid the time consuming step of model selection, Miller et al. [24] introduced nonparametric LFRMs methods to automatically infer the latent feature dimension; Zhu [34] further integrated max-margin learning ideas into LFRMs to learn discriminative latent features.

However, one limitation of the above methods is that they do not use entities' attributes or only use attributes as extra input information. Thus, these methods do not take the interaction between entities and attributes into consideration; and they cannot infer the entities' attributes, an important task in many practical applications. For example, many people have rich attribute information (e.g., gender, age, interests and employers) in online social networks (e.g., Google+ or Facebook), and it is useful to incorporate such information to make entity-entity link prediction. Meanwhile, when we use attribute information, one important issue to be addressed is missing values, which are common due to two reasons: (1) users usually set some of their private attributes publicly invisible; and (2) users may not fill out all the information in their profiles. Inferring missing attributes based on network structure [33] is an attractive topic since with more information about users, we can customize searching results, improve recommendation systems, or make personalized software.

Therefore, it is essential and challenging to develop statistical models that seamlessly integrate attributes with entities' link structures and allow link prediction and attribute inference mutually influencing each other. Supervised random walk [2] provides some attempts in combining attribute information and network structure. However, it only leverages attributes of neighborhood nodes to learn edge weight and cannot be used to infer attributes. We need some models to get more insights of the mutual influence, so that we could jointly predict entities' links and infer their attributes. Heterogeneous networks, which could better portrait the real world in that they contain multiple kinds of nodes and links, give us an elegant way to do this [32], [8]. Some recent work of link prediction has been extended to more complicated heterogeneous networks [32], [8], [26], [31], [6]. However, those methods mainly focus on using explicitly designed features such as network topology, node activity, time stamps, etc.; and to the best of our knowledge, latent feature approaches have not been well studied.

We present a novel max-margin latent feature model for heterogeneous networks to avoid the potential limitations of the methods with explicitly designed features and to analyze the mutual influence between entities and attributes. We make the following contributions:

- We formalize Entity-Attribute Networks (EANs) as a

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

generic class of heterogeneous networks. EANs can be social networks [32], [8], document networks and protein interaction networks, as long as they can be abstracted to entities, attributes and their relationships

- We propose *maximum entropy discrimination joint relational model* (MED-JRM) with variational approximation methods. MED-JRM incorporates the max-margin principle to learn discriminative latent features and jointly predict entity-entity (E-E) relationships and entity-attribute (E-A) relationships on EAN networks
- Finally, Experiments on three real world EAN networks are done to demonstrate the advantages of joint MED-JRM model on link predictions. We further extensively analyze the sensitivity to some key parameters.

The rest paper is structured as follows. Section II reviews some related work. Section III formalizes EAN as a generic class of heterogeneous networks. Section IV presents the MED-JRM model with inference algorithms. Section V presents experimental results. Finally, Section VI concludes.

II. Related Work

Our work is closely related to heterogeneous network link prediction and latent feature relational models.

A. Heterogeneous network link prediction

There have been a few studies on link prediction in heterogeneous networks. Yin et al. [32] proposed a unified framework, which augments the commonly used social network graph with attribute nodes, and they focused on exploring random walk methods with restart to improve the performance. This framework was further used by Gong et al. [8], who extended several competitive algorithms. They showed that the framework with attribute nodes is superior to those without attribute nodes. They also found that inferring missing attributes could further improve the accuracy, which demonstrates the importance of attribute nodes.

Link prediction in more complicated heterogeneous networks was also studied [26], [31], [6], [18]. For example, Sun et al. [26] formally defined the heterogeneous information network and quantified meta path [27] based topology features. Yang et al. [31] proposed MRIP model and they also explored temporal model of link formation. MRIP is based on topology and the temporal model employed features like recency, node activeness, degree preferential, etc. But as can be seen, though these link prediction algorithms can analyze complex networks, they mainly focus on using human designed features, which could have some limitations as discussed above and motivate the developments of latent feature relational models.

B. Latent feature relational models

Latent feature relational models are effective approaches to capture latent properties of a network. Suppose the number of entities is N, we can construct a binary $N \times N$ matrix Y, with $Y_{ij} = 1$ if there are observed positive links between entities i, j and $Y_{ij} = -1$, otherwise. The goal of link prediction is to learn a model that could predict whether unobserved links

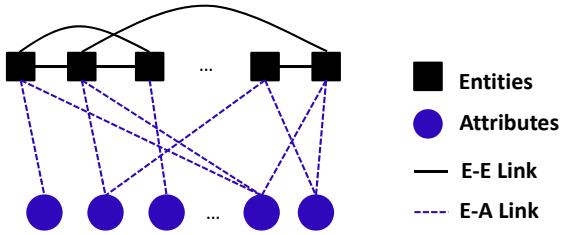


Fig. 1. An illustration of an EAN Network

are positive or negative. In the learning and predicting process, whether the link between entities i and j exists may be affected by some extra information, which is denoted by X_{ij}.

Suppose each entity i is associated with a K-dimension real valued feature vector $u_i \in \mathbb{R}^K$. Then the likelihood of two entities having positive links can be formulated as

$$p(Y_{ij} = 1 | X_{ij}, u_i, u_j) = \Phi(\psi(u_i, u_j) + \beta^\top X_{ij} + b)$$

where b is an offset parameter and $\Phi(x) = \frac{1}{1+e^{-x}}$ is the logistic function. For function $\psi(u_i, u_j)$, Hoff et al. [12] proposed latent distance model $\psi(u_i, u_j) = -d(u_i, u_j)$, where $d(\cdot)$ is a distance function, and [11] generalized it for modeling symmetric relational data $\psi(u_i, u_j) = u_i^\top D u_j$, where D is a diagonal matrix. Other latent feature relational models include nonparametric model [9], [34], relational topic model [4] and its generalization [5]. While these models are powerful, they are constrained to homogeneous networks. We will build a model for one kind of heterogeneous networks, as explained below.

III. Entity-Attribute Network

To make our joint latent feature models generally applicable, we first formalize a generic class of heterogeneous networks as *Entity-Attribute Networks*, which include social networks [32] and many others, as long as they could be abstracted to entities, attributes and relationships among them. Formally, we define:

Definition 1: An Entity-Attribute Network (EAN) is a heterogeneous network that can be characterized by a graph $G = \langle \mathcal{V}, \mathcal{E} \rangle$, where $\mathcal{V} = \mathcal{V}_N \cup \mathcal{V}_M$ is the union of entity nodes $\mathcal{V}_N$ and attribute nodes $\mathcal{V}_M$; $\mathcal{E} = \mathcal{E}_N \cup \mathcal{E}_M$ is the union of E-E links $\mathcal{E}_N$ and E-A links $\mathcal{E}_M$.

Fig. 1 shows an illustration of an EAN network. For concrete examples, in social networks, entity nodes set $\mathcal{V}_N$ represents people, attribute nodes set $\mathcal{V}_M$ represents people's profiles such as gender, working company and interests, E-E links set $\mathcal{E}_N$ represents friendship links among people and E-A links set $\mathcal{E}_M$ represents if a person in $\mathcal{V}_N$ has one profile in $\mathcal{V}_M$. Similarly, in document networks, $\mathcal{V}_N$ represents documents, $\mathcal{V}_M$ represents words, $\mathcal{E}_N$ represents citation links among documents and $\mathcal{E}_M$ represents if a document has the word.

Definition 2: Given a snapshot of EAN ($G = \langle \mathcal{V}_N \cup \mathcal{V}_M, \mathcal{E}_N \cup \mathcal{E}_M \rangle$), joint link prediction is to jointly analyze interactions of entities and attributes, then predict whether there is or will be a link $e_{ij} \in \mathcal{E}_N$ between nodes u_i, u_j or a link $e_{ij} \in \mathcal{E}_M$ between nodes u_i, v_j, where $u_i, u_j \in \mathcal{V}_N$ and $v_j \in \mathcal{V}_\mathcal{M}$.

IV. The MED-JRM Models

We formally present maximum entropy discrimination joint relational models (MED-JRM) to characterize the influence between entity-entity (E-E) relations and entity-attribute (E-A) relations in EAN networks.

A. MED

We first briefly review the maximum entropy discrimination (MED) framework [14], [15], in the context of binary classification, where binary labels $y_d \in \{-1, 1\}$ are assigned to examples x_d. Given a discriminant function $f(x; \eta)$ parameterized by η and a prior $p_0(\eta)$, MED tries to find an optimal posterior distribution $q(\eta)$, rather than a single optimal value η in standard SVM, by solving the entropic regularized risk minimization problem

$$\min_{q(\eta)\in\mathcal{P}} \mathrm{KL}(q(\eta)||p_0(\eta)) + C\mathcal{R}(q(\eta)),$$

where $\mathrm{KL}(q||p_0)$ is the K-L divergence, C is a positive regularization constant, $\mathcal{R}(q(\eta)) = \sum_d \max(0, \ell - y_d \mathbb{E}_{q(\eta)}[f(x_d; \eta)])$ (ℓ is a positive parameter that measures the cost of making wrong predictions) is the generalized hinge loss that captures the large-margin principle underlying the MED prediction rule $\hat{y} = \mathrm{sign}\, \mathbb{E}_{q(\eta)}[f(x; \eta)]$, and $\mathcal{P}$ denotes the probability simplex with an appropriate dimension.

MED subsumes SVM and provides an elegant way to integrate the discriminative max-margin learning with Bayesian generative models. Recently, many extensions have been done including those on incorporating latent variables [15], [35] [36]; those on performing structure output prediction [39]; as well as those on integrating Bayesian nonparametrics and max-margin learning [38], [34], two important subfields that have been largely treated as isolated.

B. MED-JRM for Joint Link Prediction

Suppose we have N entity nodes and M attribute nodes in EAN. As a latent space model, MED-JRM assumes that each entity is associated with a K_N dimension latent feature vector $u_i \in \{0, 1\}^{K_N}$ and each attribute is associated with a K_M dimension latent feature vector $v_j \in \{0, 1\}^{K_M}$. The latent feature matrices of entities and attributes are denoted by $U = [u_1^\top; u_2^\top; ...; u_N^\top]$ and $V = [v_1^\top; v_2^\top; ...; v_M^\top]$, respectively, where $U_{ik} = 1$ means entity i has feature k; likewise for V_{jk}. We further denote Y^N and Y^M as observed matrices, with $y_{ij}^N = 1$ when there is a link from entity i to entity j, $y_{ij}^N = -1$ otherwise; and $y_{ij}^M = 1$ when there is a link from entity i to attribute j, $y_{ij}^M = -1$ otherwise.

Given latent features u_i and u_j, MED-JRM defines the discriminant function for the link between entities i and j as

$$f(u_i, u_j; W^N) = u_i^\top W^N u_j = \mathrm{Tr}(W^N u_j u_i^\top), \qquad (1)$$

where W^N is a weight matrix associated with E-E pairs (i.e., $W_{kk'}^N$ is the weight that affects a link from entity i to entity j if i has the feature k and j has the feature k') and $\mathrm{Tr}(\cdot)$ is the trace of a matrix. Similarly, the discriminant function for the link between entity i and attribute j is

$$f(u_i, v_j; W^M) = u_i^\top W^M v_j = \mathrm{Tr}(W^M v_j u_i^\top), \qquad (2)$$

where W^M is another weight matrix which is associated with E-A pairs. As illustrated in Fig. 2, Eq. (1) models the interaction among entities, while Eq. (2) models the interaction between entities and attributes. To perform Bayesian inference, we treat W^N, W^M as random variables, and to get rid of the uncertainties, the effective discriminant functions are further defined as

$$f^N(i, j) = \mathbb{E}_{q(U,W^N)}[f(u_i, u_j; W^N)]$$
$$f^M(i, j) = \mathbb{E}_{q(U,V,W^M)}[f(u_i, v_j; W^M)],$$

which give us the link prediction rules

$$\hat{y}_{ij}^N = \mathrm{sign}\ f^N(i, j), \text{ and } \hat{y}_{ij}^M = \mathrm{sign}\ f^M(i, j).$$

If the predicted label $\hat{y}_{ij} = 1$, there exists link between the

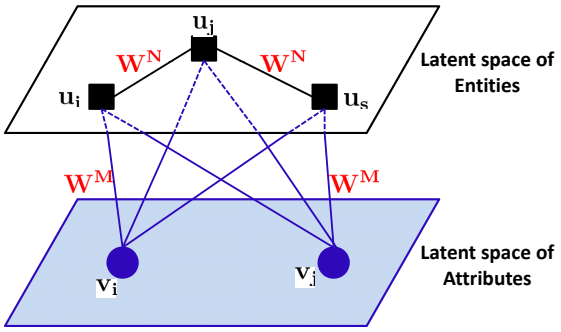


Fig. 2. E-E interaction and E-A interaction in two latent spaces

Let I^N be the set of E-E training pairs and I^M be the set of E-A training pairs, and let $\Theta = \{U, V, W^N, W^M\}$ denote all the latent variables. We define MED-JRM as solving the optimization problem

$$\min_{q(\Theta)\in\mathcal{P}} \mathcal{L}(q(\Theta)) + C_1\mathcal{R}_1(q(\Theta)) + C_2\mathcal{R}_2(q(\Theta)), \qquad (3)$$

where $\mathcal{L}(q(\Theta)) = \mathrm{KL}(q(\Theta)\|p_0(\Theta))$, $\mathcal{R}_1(q(\Theta)) = \sum_{(i,j)\in I^N} \max(0, \ell_1 - y_{ij}^N f^N(i, j))$ and $\mathcal{R}_2(q(\Theta)) = \sum_{(i,j)\in I^M} \max(0, \ell_2 - y_{ij}^M f^M(i, j))$ are hinge losses; C_1 and C_2 are positive regularization constants balancing the relative importance of various terms.

An important issue of problem (3) is to choose appropriate prior p_0. For simplicity, we assume that U, V, W^N and W^M are mutually independent a priori and choose the factorized prior $p_0(\Theta) = p_0(U)p_0(V)p_0(W^N)p_0(W^M)$. For priors $p_0(W^N)$ and $p_0(W^M)$, we use standard normal distribution, i.e. $W_{ij}^N \sim \mathcal{N}(0, I)$, and $W_{ij}^M \sim \mathcal{N}(0, I)$. For priors $p_0(U)$ and $p_0(V)$, we use the Beta-Bernoulli process [22] for finite feature matrices[1]. Specifically, we introduce auxiliary variables $\boldsymbol{\pi} = \{\pi^N, \pi^M\}$, and the priors $p_0(U)$ and $p_0(V)$ could be generated as follows

$$\pi_k^N|\alpha_N, K_N \sim \mathrm{Beta}(\frac{\alpha_N}{K_N}, 1) \quad U_{ik}|\pi_k^N \sim \mathrm{Bernoulli}(\pi_k^N)$$
$$\pi_k^M|\alpha_M, K_M \sim \mathrm{Beta}(\frac{\alpha_M}{K_M}, 1) \quad V_{ik}|\pi_k^M \sim \mathrm{Bernoulli}(\pi_k^M).$$

[1]Though an infinite MED-JRM model can be formulated by using Indian buffet process [9], this paper considers the finite MED-JRM model for simplicity.

Then the augmented learning problem is to solve

$$\min_{q(\boldsymbol{\pi},\Theta)\in\mathcal{P}} \quad \mathrm{KL}(q(\boldsymbol{\pi},\Theta)||p_0(\boldsymbol{\pi},\Theta)) + C_1\mathcal{R}_1(q(\Theta)) + C_2\mathcal{R}_2(q(\Theta)) \tag{4}$$

where $p_0(\boldsymbol{\pi},\Theta) = p_0(\pi^N)p(U|\pi^N)p_0(\pi^M)p(V|\pi^M)$ $p_0(W^N)p_0(W^M)$.

C. Approximate Mean-Field Inference Algorithms

Though problem (4) is convex and we can derive the optimal solution using convex analysis tools, it is generally intractable to make inference with the optimal solution. Therefore, we resort to approximation inference by making additional mean-field assumptions. Specifically, we impose the following additional mean-field constraint to problem (4):

$$q(\boldsymbol{\pi},\Theta) = q(W^N)q(W^M)\Big(\prod_{k=1}^{K_N} q(\pi_k^N|\gamma_k^N) \prod_{i=1}^{N} q(U_{ik}|\sigma_{ik}^N)\Big) \times \Big(\prod_{k=1}^{K_M} q(\pi_k^M|\gamma_k^M)\prod_{i=1}^{M} q(V_{ik}|\sigma_{ik}^M)\Big),$$

where $q(\pi_k^N|\gamma_k^N) = \mathrm{Beta}(\gamma_{k_1}^N, \gamma_{k_2}^N)$, $q(\pi_k^M|\gamma_k^M) = \mathrm{Beta}(\gamma_{k_1}^M, \gamma_{k_2}^M)$, $q(U_{ik}|\sigma_{ik}^N) = \mathrm{Bernoulli}(\sigma_{ik}^N)$ and $q(V_{ik}|\sigma_{ik}^M) = \mathrm{Bernoulli}(\sigma_{ik}^M)$. Then problem (4) can be solved to find a local optimum by executing the four steps below iteratively (to save space we only provide the outline).

Step I: Solve for $q(W^N)$. By fixing other model parameters and latent features, we can show that the posterior is also a normal distribution $q(W^N) = \mathcal{N}(\Lambda^N, I)$, and Λ^N can be solved by optimizing the subproblem:

$$\min_{\Lambda^N,\boldsymbol{\xi}} \quad \frac{1}{2}||\Lambda^N||_2^2 + C_1\sum_{(i,j)\in I^N}\xi_{ij}$$
$$\forall(i,j)\in I^N, \text{s.t.}: y_{ij}^N(\mathrm{Tr}(\Lambda^N\mathbb{E}[u_j u_i^\top])) \geq \ell_1 - \xi_{ij}.$$

where $\boldsymbol{\xi} = \{\xi_{ij} : (i,j)\in I^N\}$ are slack variables. This problem is the same as a standard SVM, thus it could be solved with existing efficient SVM tools such as LIBSVM or SVMLight. We use SVMLight in experiments.

Step II: Solve for $q(W^M)$. Similar as in Step I, $q(W^M) = \mathcal{N}(\Lambda^M, I)$, we can get the optimization problem for Λ^M by solving:

$$\min_{\Lambda^M,\boldsymbol{\xi}} \quad \frac{1}{2}||\Lambda^M||_2^2 + C_2\sum_{(i,j)\in I^M}\xi_{ij}$$
$$\forall(i,j)\in I^M, \text{s.t.}: y_{ij}^M(\mathrm{Tr}(\Lambda^M\mathbb{E}[v_j u_i^\top])) \geq \ell_2 - \xi_{ij},$$

This again can be efficiently solved by SVMLight.

Step III: Solve for $q(\pi^N, U)$. By fixing others, this step involves solving the subproblem:

$$\min_{q(\pi^N,U)} \quad \mathrm{KL}(q(\pi^N,U)||p_0(\pi^N,U)) + C_1\mathcal{R}_1(q(\Theta)) + C_2\mathcal{R}_2(q(\Theta)).$$

For $q(\pi^N)$, by setting the gradients at zero, we can derive the update equations:

$$\begin{cases} \gamma_{k_1}^N &= \frac{\alpha_N}{K_N} + \sum_{i=1}^N \sigma_{ik}^N \\ \gamma_{k_2}^N &= N + 1 - \sum_{i=1}^N \sigma_{ik}^N. \end{cases} \tag{5}$$

For $q(U)$, we can still derive a closed-form update equation by using sub-gradient methods. Namely, by setting the sub-gradients at zero, we have:

$$\sigma_{ik}^N = \Phi\Big(\mathbb{E}[\ln\pi_k^N] - \mathbb{E}[\ln(1-\pi_k^N)] - C_1\frac{\partial\mathcal{R}_1}{\partial\sigma_{ik}^N} - C_2\frac{\partial\mathcal{R}_2}{\partial\sigma_{ik}^N}\Big), \tag{6}$$

where $\Phi(\cdot)$ is the logistic function.

Step IV: Solve for $q(\pi^M, V)$. Similar as in Step III, we get:

$$\begin{cases} \gamma_{k_1}^M &= \frac{\alpha_M}{K_M} + \sum_{i=1}^M \sigma_{ik}^M \\ \gamma_{k_2}^M &= M + 1 - \sum_{i=1}^M \sigma_{ik}^M, \end{cases} \tag{7}$$

$$\sigma_{jk}^M = \Phi\Big(\mathbb{E}[\ln\pi_k^M] - \mathbb{E}[\ln(1-\pi_k^M)] - C_2\frac{\partial\mathcal{R}_2}{\partial\sigma_{jk}^M}\Big). \tag{8}$$

With these update equations above, we summarize the procedure in Algorithm 1. For the convergence condition, we can monitor the value q of Problem (4) or set a maximum iteration number i_m. Convergence is met if q changes little from the previous iteration or the number of iterations is greater than i_m.

Algorithm 1 for Learning MED-JRM

```
initialize γ^N_{k_1} = α_N/K_N, γ^N_{k_2} = 1 and γ^M_{k_1} = α_M/K_M, γ^M_{k_2} = 1
initialize σ^N_{ik} and σ^M_{jk} randomly
initialize Λ^N_{kk'} and Λ^M_{kk'} randomly
repeat
    repeat
        for each k in [0, K_N) do
            update γ^N_{k_1}, γ^N_{k_2} using Eq. (5)
        for each k in [0, K_M) do
            update γ^M_{k_1}, γ^M_{k_2} using Eq. (7)
        for each i in [0, N) do
            for each k in [0, K_N) do
                update σ^N_{ik} using Eq. (6)
        for each j in [0, M) do
            for each k in [0, K_M) do
                update σ^M_{jk} using Eq. (8)
    until convergence
    update Λ^N_{kk'}, Λ^M_{kk'} using SVMLight, respectively
until convergence
```

D. Prediction

As in many existing network analysis models, we restrict ourselves to the case of *warm-start prediction*, that is, all the entities and attributes are observed at least once in the training phase. Extension to the more subtle cold-start prediction [3] is interesting and comprise our future work. Under the warm-start condition, MED-JRM can learn a latent feature for every attribute and every entity, as well as the weight matrices W^N and W^M. Specifically, through the training algorithm, we can get the variational parameters $\sigma_{ik}^N, \sigma_{jk}^M$ and model parameters $\Lambda_{kk'}^N, \Lambda_{kk'}^M$. Then, the expectations are $\mathbb{E}[W_{kk'}^N] = \Lambda_{kk'}^N$, $\mathbb{E}[U_{ik}] = \sigma_{ik}^N$, $\mathbb{E}[W_{kk'}^M] = \Lambda_{kk'}^M$, and $\mathbb{E}[V_{jk}] = \sigma_{jk}^M$.

Thus, for E-E link prediction, the effective discriminant function is:

$$f^N(i,j) = \mathrm{Tr}(\mathbb{E}[W^N]\mathbb{E}[u_j u_i^\top]) = \begin{cases} (\sigma_i^N)^\top \Lambda^N \sigma_j^N & i \neq j \\ (\sigma_i^N)^\top \Lambda^N \sigma_j^N + \sum_k \Lambda_{kk}^N \sigma_{ik}^N (1-\sigma_{ik}^N) & i = j \end{cases}$$

and for E-A link prediction, the effective discriminant function is:

$$f^M(i,j) = \mathrm{Tr}(\mathbb{E}[W^M]\mathbb{E}[v_j u_i^\top]) = (\sigma_i^N)^\top \Lambda^M \sigma_j^M.$$

V. Experiments

A. Datasets and Experiment Setup

We apply the proposed model on three real world network datasets – Google+ social networks, Cora document networks and CiteSeer document networks. The datasets we use are randomly subsampled from corresponding original ones [8], [21], [7]. *Google+ dataset*: Our Google+ dataset is a subset of a snapshot of Google+ in September, 2011. It consists of 203 entities, 124 attributes, 468 E-E links and 443 E-A links. *Cora dataset*: As an internet portal, Cora places computer science research papers into a topic hierarchy and maps the citation between papers. The E-E links can be formed by papers' citation and entities' attributes are represented by words in papers' abstracts, forming E-A links. In total, we have 209 entities, 582 attributes, 700 E-E links and 3717 E-A links. *Citeseer dataset*: CiteSeer is an automatic citation indexing system and it indexes academic literature including its abstract and citations. In total, we have 102 entities, 587 attributes, 234 E-E links and 2617 E-A links.

Baselines: To better illustrate the advantages of MED-JRM, we compare with two types of baselines – one type works on homogeneous networks and the other works under the heterogenous EAN networks.

Type 1: Homogeneous network baselines. We use a state-of-the-art model MED-NRM [34] [2](maximum entropy discrimination nonparametric relational model), which is a kind of nonparametric latent feature relational models and has shown excellent performance in link prediction tasks. But it doesn't take E-A interaction into consideration and could only be used to predict E-E relationships.

Type 2: EAN baselines. As [8] claims they tested several leading algorithms under similar framework, we choose some of the well-performed algorithms from it and apply them to EAN. Specifically, we use: *Common Neighbors* (CN-EAN), *Adamic/Adar* (AA-EAN), *Low Rank Approximation* (LRA-EAN) and *Random Walk with Restart* (RWR-EAN). The suffix "EAN" is used to indicate that the algorithms are applied to EAN networks. Please see [8] for more details.

Experiment setup: we randomly take 80% of each dataset for training and the remaining 20% for testing, under the constraint that every entity and attribute appear at least once in the training set. Since all the three network datasets are extremely sparse (i.e., the number of negative links is much larger than positive ones), we randomly sample 2% negative links[3] for training in the experiments. Except LRA-EAN and RWR-EAN, which are implemented in Matlab, all the algorithms are programmed in C++. MED-NRM and MED-JRM are run in Ubuntu 12.04 with 2.4 GHz Intel Xeon CPU and 24 GB main memory, and the other algorithms are executed in Windows 8 with 2.5 GHz Intel Core P8700 and 4GB main memory. We use AUC (i.e., Area Under the ROC curve) value as performance measurement, same as in [8], [34].

TABLE I. AUC of E-E and E-A link prediction on the Google+ dataset

Methods	E-E AUC	E-A AUC
MED-NRM	0.771 ± 0.020	—
CN-EAN	0.724	0.697
AA-EAN	0.734	0.700
LRA-EAN	0.590	0.612
RWR-EAN	0.807	0.829
MED-JRM	$\mathbf{0.850} \pm 0.006$	$\mathbf{0.841} \pm 0.003$

TABLE II. AUC of E-E and E-A link prediction on the Cora dataset

Methods	E-E AUC	E-A AUC
MED-NRM	0.797 ± 0.017	—
CN-EAN	0.708	0.605
AA-EAN	0.734	0.604
LRA-EAN	0.802	0.703
RWR-EAN	0.823	0.742
MED-JRM	$\mathbf{0.845} \pm 0.005$	$\mathbf{0.821} \pm 0.019$

Note that in the training phase, we jointly analyze E-E interaction and E-A interaction for learning good latent feature representations by MED-JRM. In the testing phase, though we are able to perform both E-E link prediction and E-A link prediction simultaneously, we found that the practical performance could be improved if we focus on predicting one type of links at one time. Similar observations apply to other EAN baselines. Therefore, we adopt this testing strategy for all the EAN models for fair comparison. This simplified testing strategy can be further improved by an iterative model learning method, as discussed later.

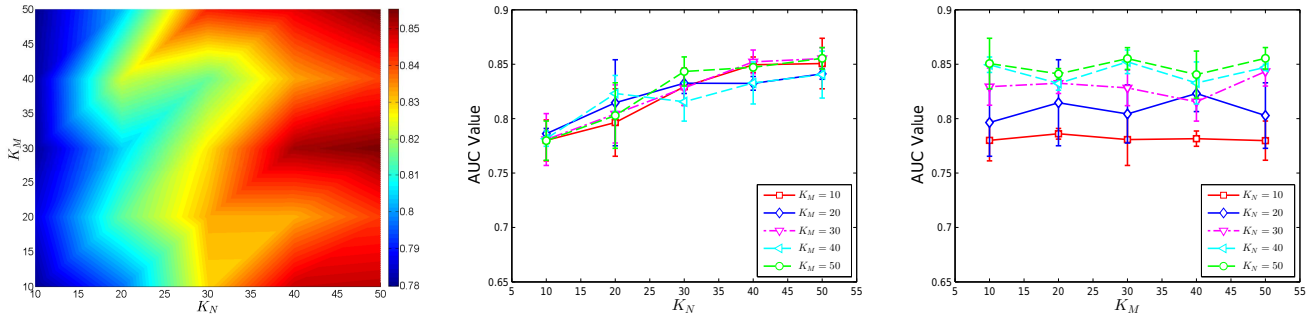
B. Quantitative Results

We first report link prediction performance for both E-E and E-A relationship in terms of AUC values. For models with random initialization (i.e., MED-NRM and MED-JRM), we show the mean and standard deviation of AUC scores with five random restarts.

Tables I, II, III show the results on the three datasets, respectively. As can be seen, MED-NRM as an excellent homogeneous network model has better performance than some of the algorithms under EAN framework on Google+ and Cora, though it doesn't use attribute information. However, it is outperformed by Random walk with Restart, another EAN method that considers both entity and attribute information. Overall, our MED-JRM has the best performance for both E-E and E-A link prediction on all datasets, which demonstrates that max-margin supervised MED-JRM is not only good at capturing interactions of E-E and E-A, but also superior in prediction.

[2]Datasets in this paper are not used since they don't contain attribute information

[3]Other subsample ratios can be used without significantly affecting the performance by using appropriate regularization parameters.



(a) Case I (b) Case II (c) Case III

Fig. 3. E-E link prediction performance w.r.t. latent feature dimensions K_N and K_M on the Google+ dataset. Fig. 3a is heat map and the color represents AUC value. To better illustrate the map, we provide Fig. 3b and Fig. 3c. In Fig. 3b we keep K_M fixed and change K_N, while in Fig. 3c we keep K_N fixed and change K_M.

TABLE III. AUC OF E-E AND E-A LINK PREDICTION ON THE CITESEER DATASET

Methods	E-E AUC	E-A AUC
MED-NRM	0.543 ± 0.025	—
CN-EAN	0.790	0.656
AA-EAN	0.806	0.656
LRA-EAN	0.802	0.738
RWR-EAN	0.798	0.716
MED-JRM	$\mathbf{0.894} \pm 0.016$	$\mathbf{0.768} \pm 0.023$

On the CiteSeer dataset (Table III), MED-NRM preforms surprisingly bad. It is a good example to show that sometimes, only homogeneous network structure may not be sufficient to make good prediction on links; thus extra information (e.g., attributes) is needed to obtain good performance. By adding attribute nodes, even a simple method like "CN-EAN" can do better than the powerful MED-NRM. This demonstrates the advantages of EAN methods on incorporating E-A information.

We also note that on the Google+ dataset (Table I), the result of MED-JRM in E-A link prediction is only a little better than the best baseline. The reason might be that the Google+ dataset was crawled when the company just set up Google+ services, so there are too many missing attributes and the network is not very stable, which brings some difficulties in attribute prediction for our model. While in the more stable Cora network, MED-JRM obtains significant improvements in attribute prediction over baselines.

C. Sensitivity Analysis

We present sensitivity analysis to some key parameters in MED-JRM.

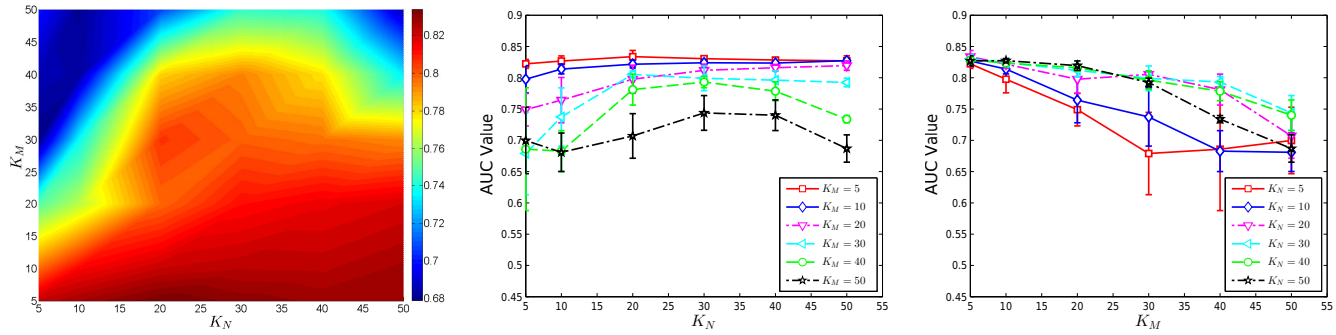
1) Sensitivity to Latent Feature Dimensions: Latent feature dimension is a very important factor for our model and we analyze its effect in this section.

Google+: Fig. 3 and Fig. 4 show how the performance varies along with the change of latent feature dimensions K_N and K_M on the Google+ dataset, where Fig. 3a and Fig. 4a are heat maps and the color represents AUC values. In Fig. 3, we can see that the results of E-E link prediction are mainly influenced by K_N, while K_M plays a less important role. In addition, with the increase of K_N, AUC tends to increase; when $K_N = 50$, AUC becomes stable at around 0.85. While in E-A link prediction (see Fig. 4), the influence of K_M is much more important than that of K_N; and with the increase of K_M, AUC tends to decrease. The reason might be that the number of attributes and entity-attribute links in Google+ is small, so the model may face overfitting problem for a large K_M. When $K_M = 5$, AUC becomes stable at around 0.83 or 0.84.

The observations of the performance on Google+ are reasonable – the prediction task is more sensitive to its corresponding latent feature dimension, that is, when predicting E-E relationship, the latent feature dimension of entities K_N dominates the performance; when predicting E-A relationship, the latent feature dimension of attribute K_M is more important. Therefore, if we want to perform the two tasks simultaneously, we can set K_M relatively small and K_N relatively large.

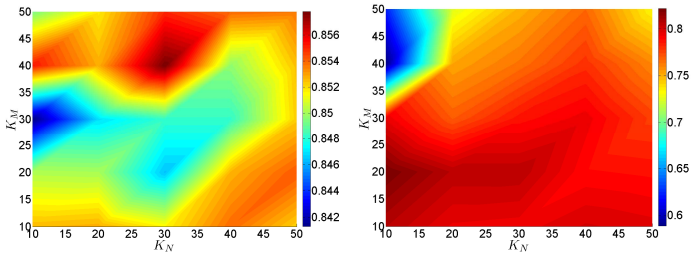
Cora: Fig. 5a shows how latent feature dimensions affect the AUC of E-E link prediction on the Cora dataset. It seems there are no patterns at first glance. However, note that the legend ranges approximately from 0.841 to 0.857 – the interval is less than 0.02, meaning that MED-JRM performs steadily well with respect to latent feature dimensions. Fig. 5b is the heat map for E-A link prediction. From this figure we can see that similar as E-A link prediction on Google+, K_N doesn't have much influence on the performance, while K_M does. Generally, MED-JRM works well if $K_M \leq 30$ and AUC decreases fast as K_M becomes larger if K_N is small. In conclusion, we don't need to care much about K_N and should keep K_M not too large if we want to do both E-E and E-A link prediction simultaneously.

2) Sensitivity to Other Parameters: To save space we only analyze how the performance is influenced by parameters $C = \{C_1, C_2\}$ on Cora. These parameters are used to balance the relative importance of KL divergence and hinge losses. Fig. 6 describes AUC sensitivity w.r.t C. In general, the trends in Fig. 6a and Fig. 6b are similar. When we keep one parameter small and tune another, the performance is good. When C_1 and C_2 are both large, we get low AUC values. This suggests that we shouldn't give too much weights for hinge loss. It is worth to note that the sensitivity of E-A link prediction over C is more apparent than that of E-E link prediction over C. Thus in our experiments, the range of C we choose in Fig. 6b is



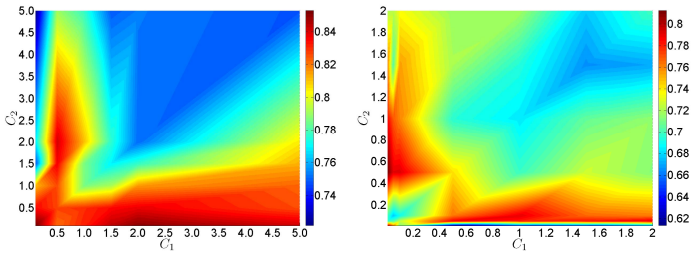
(a) Case I (b) Case II (c) Case III

Fig. 4. E-A link prediction performance w.r.t latent feature dimensions K_N and K_M on the Google+ dataset. Similar as Fig. 3, the color in Fig. 4a represents AUC value. In Fig. 4b, we keep K_M fixed and change K_N; in Fig. 4c, we keep K_N fixed and change K_M.



(a) entity-entity (b) entity-attribute

Fig. 5. Performance w.r.t dimensions K_N and K_M on the Cora dataset.



(a) entity-entity (b) entity-attribute

Fig. 6. Performance w.r.t parameters C_1 and C_2 on the Cora dataset.

relatively smaller than that in Fig. 6a.

D. More Results

As we have stated, the experiments in the paper adopt a practically well-performed testing strategy that separately makes the E-E link prediction and E-A link prediction. In order to obtain the mutual enhancements between the two prediction tasks, an iterative testing strategy can be developed (inspired by semi-supervised methodology and [8]). Specifically, for E-E link prediction, we iteratively perform two steps: 1) learning MED-JRM on the current training set; 2) predicting entities' attributes and adding some entity-attribute link data to the training set. The procedure is illustrated in Algorithm 2 below, where the step 7 is performed by first sorting the prediction scores s_{ij} decreasingly with respect to index j; and then adding the first i_a examples of s_i as positive examples and the last i_a examples of s_i as negative examples. A similar iterative procedure can be developed for enhancing the task of entity-attribute link prediction. Note that in Algorithm 2, i_n is defined

TABLE IV. E-E LINK PREDICTION AUC WITH ITERATIVE MODEL LEARNING ON THE GOOGLE+ DATASET

i_n \ i_a	1	2	3
0	0.817 ± 0.005	0.817 ± 0.005	0.817 ± 0.005
1	0.823 ± 0.006	0.830 ± 0.014	0.826 ± 0.016
2	0.830 ± 0.015	0.825 ± 0.014	0.836 ± 0.024
3	0.827 ± 0.008	0.818 ± 0.012	0.824 ± 0.005
4	0.828 ± 0.005	0.838 ± 0.017	0.824 ± 0.008

TABLE V. E-E LINK PREDICTION AUC WITH ITERATIVE MODEL LEARNING ON THE CORA DATASET

i_n \ i_a	1	2	3
0	0.852 ± 0.007	0.852 ± 0.007	0.852 ± 0.007
1	0.854 ± 0.008	0.848 ± 0.008	0.853 ± 0.008
2	0.856 ± 0.006	0.853 ± 0.006	0.860 ± 0.005
3	0.861 ± 0.008	0.849 ± 0.005	0.852 ± 0.006
4	0.853 ± 0.003	0.851 ± 0.008	0.852 ± 0.010

as the maximum iteration number and i_a is the number of positive examples and negative examples added to the training set at every iteration. Furthermore, when we update the training set, if an example is already in the training set, it won't be added again to avoid duplicates.

Algorithm 2 Iterative Model Learning

1: initialize $n = 0$
2: learn MED-JRM with Algorithm 1
3: **while** $n < i_n$ **do**
4: **for** each i in $[0, N)$ **do**
5: **for** each j in $[0, M)$ **do**
6: predict link score s_{ij}
7: Update the training set with i_a positive examples and i_a negative examples.
8: learn MED-JRM
9: $n \leftarrow n + 1$
10: perform final E-E link prediction

Table IV and V show the results on E-E link prediction, where the row with $i_n = 0$ corresponds to the MED-JRM model without iterative model learning. On the Google+ dataset, we can see that the performance is indeed improved, however on the Cora dataset, AUC changes little. This is

consistent with previous observations that there may be many missing attributes on the Google+ dataset and Cora is a relative stable dataset. The results also suggest that retrieving missing attributes could help to learn a better model.

VI. Conclusions and Discussions

In this paper, we propose maximum entropy discrimination joint latent feature relational models (MED-JRM) that learn discriminative latent features for both link prediction and attribute inference on entity-attribute networks (EANs), a generic class of heterogeneous networks that could be used in different kinds of networks as long as they can be represented by entities, attributes, and relationships. In order to learn discriminative latent features, MED-JRM adopts max-margin learning ideas under the MED framework. Experimental results on several real networks demonstrate superior performance of MED-JRM over existing competitors.

We have mainly focused on developing flexible latent feature models to characterize the mutual interactions of entity-entity and entity-attribute. Though the joint model gets superior prediction performance, the flexibility comes with computational cost when K_N and K_M are large, since the time complexity of solving multiple latent SVMs is increased by $O(K^2)$ with respect to latent feature dimension K. Inspired by the very recent work [37], one of our future work is to improve the efficiency by leveraging the data augmentation ideas to avoid solving SVM problems. Furthermore, we are interested in incorporating Bayesian nonparametric techniques [34] to automatically infer the latent feature dimension.

Acknowledgment

This work is supported by National Key Project for Basic Research of China (Grant Nos: 2013CB329403, 2012CB316301), National Natural Science Foundation of China (Nos: 61305066, 61322308, 61332007), Tsinghua Self-innovation Project (Grant Nos: 20121088071, 201110811) and China Postdoctoral Science Foundation Grant (Grant Nos: 2013T60117, 2012M520281).

References

[1] S. F. Adafre and M. d. Rijke. Discovering missing links in wikipedia. In *International Workshop on Link Discovery*, 2005.

[2] L. Backstrom and J. Leskovec. Supervised random walks: Predicting and recommending links in social networks. In *WSDM*, 2011.

[3] J. Bobadilla, F. Ortega, A. Hernando, and A. Gutirrez. Recommender Systems Survey. *Knowledge-Based Systems*, 46:109–132, 2013.

[4] J. Chang and D. Blei. Relational topic models for document networks. In *AISTATS*, 2009.

[5] N. Chen, J. Zhu, F. Xia, and B. Zhang. Generalized relational topic models with data augmentation. In *IJCAI*, 2013.

[6] D. Davis, R. Lichtenwalter, and N. V. Chawla. Multi-relational link prediction in heterogeneous information networks. In *ASONAM*, 2011.

[7] C. . Giles, K. D. Bollacker, and S. Lawrence. Citeseer: an automatic citation indexing system. In *ACM conference on Digital libraries*, pages 89–98, 1998.

[8] N. Z. Gong, A. Talwalkar, L. Mackey, L. Huang, E. C. Shin, E. Stefanov, E. Shi, and D. Song. Jointly predicting links and inferring attributes using a social-attribute network (san). In *SIGKDD Workshop*, 2012.

[9] T. Griffiths and Z. Ghahramani. Infinite latent feature models and the indian buffet process. In *NIPS*. 2006.

[10] M. A. Hasan, V. Chaoji, S. Salem, and M. Zaki. Link prediction using supervised learning. In *SDM workshop*, 2006.

[11] P. Hoff. Modeling homophily and stochastic equivalence in symmetric relational data. In *NIPS*. 2008.

[12] P. D. Hoff, A. E. Raftery, and M. S. Handcock. Latent space approaches to social network analysis. *JASA*, 97:1090–1098, 2002.

[13] Z. Huang, X. Li, and H. Chen. Link prediction approach to collaborative filtering. In *JCDL*, 2005.

[14] T. Jaakkola, M. Meila, and T. Jebara. Maximum Entropy Discrimination. In *NIPS*, 1999.

[15] T. Jebara. *Discriminative, Generative and Imitative Learning*. PhD thesis, MIT, 2002.

[16] R. Li, J. X. Yu, and J. Liu. Link prediction: the power of maximal entropy random walk. In *CIKM*, 2011.

[17] D. Liben-Nowell and J. Kleinberg. The link prediction problem for social networks. In *CIKM*, 2003.

[18] R. N. Lichtenwalter and N. V. Chawla. Vertex Collocation Profiles: Subgraph Counting for Link Analysis and Prediction. In *WWW*, 2012.

[19] R. N. Lichtenwalter, J. T. Lussier, and N. V. Chawla. New perspectives and methods in link prediction. In *SIGKDD*, 2010.

[20] L. Lv and T. Zhou. Link prediction in complex networks: A survey. *Physica A: Statistical Mechanics and its Applications*, 390(6):1150–1170, 2011.

[21] A. McCallum, K. Nigam, J. Rennie, and K. Seymore. Automating the construction of internet portals with machine learning. *Information Retrieval*, 2000.

[22] E. Meeds, Z. Ghahramani, R. M. Neal, and S. T. Roweis. Modeling dyadic data with binary latent factors. In *NIPS*. 2007.

[23] A. K. Menon and C. Elkan. Link prediction via matrix factorization. In *ECML*, 2011.

[24] K. Miller, T. Griffiths, and M. Jordan. Nonparametric latent feature models for link prediction. In *NIPS*. 2009.

[25] C. Perlich, G. Swirszcz, and R. Lawrence. Content-based link prediction for patent marketing. Technical report, IBM, 2009.

[26] Y. Sun, J. Han, C. C. Aggarwal, and N. V. Chawla. When will it happen? - relationship prediction in heterogeneous information networks. In *WSDM*, 2012.

[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. *VLDB Journal*, 4(11):992–1003, 2011.

[28] H. Tong, C. Faloutsos, and J. Pan. Fast random walk with restart and its applications. In *ICDM*, 2006.

[29] C. Wang, V. Satuluri, and S. Parthasarathy. Local probabilistic models for link prediction. In *ICDM*, 2007.

[30] Y. Yamanishi, J. Vert, and M. Kanehisa. Supervised enzyme network inference from the integration of genomic data and chemical information. *Bioinformatics*, 21(1):468–477, 2005.

[31] Y. Yang, N. V. Chawla, Y. Sun, and J. Han. Predicting links in multi-relational and heterogeneous networks. In *ICDM*, 2012.

[32] Z. Yin, M. Gupta, T. Weninger, and J. Han. A unified framework for link recommendation using random walks. In *ASONAM*, 2010.

[33] E. Zheleva and L. Getoor. To join or not to join: the illusion of privacy in social networks with mixed public and private user profiles. In *WWW*, 2009.

[34] J. Zhu. Max-margin nonparametric latent feature models for link prediction. In *ICML*, 2012.

[35] J. Zhu, A. Ahmed, and E. P. Xing. MedLDA: maximum margin supervised topic models for regression and classification. In *ICML*, 2009.

[36] J. Zhu, A. Ahmed, and E. P. Xing. MedLDA: Maximum margin supervised topic models. *JMLR*, 13:2237–2278, 2012.

[37] J. Zhu, N. Chen, H. Perkins, and B. Zhang. Gibbs max-margin topic models with data augmentation. *JMLR*, 15:949–986, 2014.

[38] J. Zhu, N. Chen, and E. P. Xing. Infinite latent svm for classification and multi-task learning. In *NIPS*. 2011.

[39] J. Zhu and E. P. Xing. Maximum entropy discrimination Markov networks. *JMLR*, 10:2531–2569, 2009.