Combining Link and Content for Community Detection: A Discriminative Approach

Tianbao Yang, Rong Jin
Michigan State University
East Lansing, MI, USA
{yangtia1,rongjin}@msu.edu

Yun Chi, Shenghuo Zhu
NEC Laboratories America
Cupertino, CA, USA
{ychi,zsh}@sv.nec-labs.com

ABSTRACT

In this paper, we consider the problem of combining link and content analysis for community detection from networked data, such as paper citation networks and Word Wide Web. Most existing approaches combine link and content information by a *generative* model that generates both links and contents via a shared set of community memberships. These generative models have some shortcomings in that they failed to consider additional factors that could affect the community memberships and isolate the contents that are irrelevant to community memberships. To explicitly address these shortcomings, we propose a discriminative model for combining the link and content analysis for community detection. First, we propose a *conditional* model for link analysis and in the model, we introduce hidden variables to explicitly model the popularity of nodes. Second, to alleviate the impact of irrelevant content attributes, we develop a *discriminative* model for content analysis. These two models are unified seamlessly via the community memberships. We present efficient algorithms to solve the related optimization problems based on bound optimization and alternating projection. Extensive experiments with benchmark data sets show that the proposed framework significantly outperforms the state-of-the-art approaches for combining link and content analysis for community detection.

Categories and Subject Descriptors

H.2.8 [**Database Management**]: Database Applications—*Data mining*; H.3.3 [**Information Storage and Retrieval**]: Information Search and Retrieval—*Clustering*

General Terms

Algorithms, Experimentation, Measurement, Theory

Keywords

Discriminative Model, Link Analysis, EM Algorithm, Two-Stage Optimization

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, to republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.
KDD'09, June 28–July 1, 2009, Paris, France.
Copyright 2009 ACM 978-1-60558-495-9/09/06 ...$5.00.

1. INTRODUCTION

As online repositories such as digital libraries and user-generated media(e.g. blogs) become more popular, analyzing such networked data has become an increasingly important research issue. One major topic in analyzing such networked data is to detect salient communities among individuals. Community detection has many applications such as understanding the social structure of organizations and modeling large-scale networks in Internet services [32]. While there are different formulations for community detection, in this work, we focus on the unsupervised learning, or the clustering viewpoint, a commonly accepted and well studied perspective.

A networked data set is usually represented as a graph where individuals in the network are represented by the nodes in the graph. The nodes are tied with each other by either directed links or undirected links, which represent the relations among the individuals. In addition to the links that they are incident to, nodes are often described by certain attributes, which we refer to as *contents* of the nodes. For example, when it comes to the web pages, online blogs, or scientific papers, the contents are usually represented by histograms of keywords; in the network of co-authorship, the contents of nodes can be the demographic or affiliation information of researchers.

Many existing studies on community detection focus on either link analysis or content analysis. However, neither information alone is satisfactory in determining accurately the community memberships: the link information is usually sparse and noisy and often results in a poor partition of networks; the irrelevant content attributes could significantly mislead the process of community detection. It is therefore important to combine the link analysis and content analysis for community detection in networks. Recently, several approaches have been proposed to combine link and content information for community detection. However, as we will survey in the next section, most of these approaches adopted a *generative* framework where a generative link model and a generative content model are combined through a set of shared hidden variables of community memberships. We argue that such a generative framework suffers from two shortcomings. First, community membership by itself is insufficient to model links—link patterns are usually affected by factors other than communities such as the popularity of a node(i.e. how likely the node is cited by other nodes). Second, the content information often include irrelevant attributes and as a result, a generative model without feature selection usually leads to poor performance.

In this paper, we propose a *discriminative* model of combining link and content analysis for community detection that explicitly addresses the above shortcomings of existing approaches. Our main contributions are summarized as follows.

- We propose a conditional model for link analysis. In contrast to generative models, our approach does not attempt to generate the links; instead, the conditional probability for the destination of a given link is to be captured. To achieve this, in our model we introduce a hidden variable to capture the popularity of a node in terms of how likely the node is cited by other nodes.
- To alleviate the impact of irrelevant content attributes, we adopt a discriminative approach to make use of the node contents. We refer to this part as discriminative content model. As a consequence, the attributes are automatically weighed by their discriminative power in terms of telling apart salient communities.
- We combine the above two models into a unified framework and propose a novel two-stage optimization algorithm for the maximum likelihood inference. In addition, we show how the proposed link model and content model can be used to extend existing complementary approaches. Additional algorithms are presented to solve the extended models.

To the best of our knowledge, the model proposed in this paper is the first that combines conditional link models and discriminative content models for community detection. We conduct extensive experimental studies by using several benchmark data sets. The experimental results show significant improvement over the state-of-the-art approaches. Additional experiments are conducted to further verify the effectiveness of each of our link model and content model, respectively.

The rest of the paper is organized as follows. In section 2 we give an overview of the related work. In Section 3 we present and analyze the conditional link model. In Section 4, we extend the link model to include the content information. Also in Section 4, we describe the two-stage optimization algorithm. In Section 5, we show extensions by combining our link model and content model with other existing content and link models. In Section 6, we show extensive experimental results on benchmark data sets. Finally, we give conclusion in Section 7.

2. RELATED WORK

In this section, we review the existing work for community-detection using link analysis, content analysis, and their combination.

Link Analysis for Community Detection.

Approaches in this area can be classified into two categories: measure-based approaches and probabilistic model based approaches. In the literature of measure-based approaches, a measure is proposed to quantify the quality of partition, and the partition is obtained either by optimizing the measure or by iteratively adding and removing edges or nodes from the existing partitions to improve the measure. Some of the most well-known measures include normalized cut [26, 31] and modularity [24], which have been examined in many previous studies [26, 31, 6, 22, 23, 12]. Other measures used for clustering can be found in [3, 29]. Besides measure-based algorithms, many probabilistic models are developed for community detection. One such model is the stochastic block model [27], which assumes that links are generated with the probabilities that only depend on the communities of nodes. Variants of the stochastic block model include mixed-membership stochastic block model [2] and Bayesian stochastic block model [14]. In addition, some other probabilistic models identify the optimal communities by soft-graph clustering [33, 30, 25].

Most of the existing approaches either assume links are nondirectional or treat directional links as nondirectional ones. Additional studies are devoted to address directional links in network analysis, including PageRank [28], HITS [17], and PHITS [7]. In PageRank, each web page is assigned a score based on the random walk model. HITS [17] derives an authoritative and a hub score for each web pages from the link structure. Cohn et al. [7] proposed the PHITS algorithm that extends the HITS algorithm for community detection by a probabilistic model that is similar to Probabilistic Latent Semantic Analysis(PLSA) [14]. LDA-Link model [10] extends Latent Dirichlet Allocation (LDA) for link analysis by assuming a link distribution for each community. Other link models proposed in the framework of LDA can be found in [9, 13]. R. Nallapti et al. [21] extend the mixed membership stochastic block (MMSB) model [2] to directional links.

Content Analysis for Community Detection.

One of the most well-known approaches for content analysis is the topic model, where each topic is naturally interpreted as a community in our framework. Two well-known topic models are PLSA [15] and LDA [5]. Most topic models are generative and are vulnerable to the words that are irrelevant to the target topics. To overcome this problem, S. Lacoste-Julien et al. [18] proposed a discriminative LDA. The main problem with discriminative LDA is that it is a supervised learning algorithm and cannot be applied directly to a unsupervised learning setup, which is the case of our problem. In contrast, the discriminative framework proposed in this paper does not require the labeling information. It automatically discovers an appropriate discriminative model that fits best with the link information.

Combined Link and Content Analysis.

As aforementioned, neither link information nor content information is sufficient to decide the community memberships. Combining link and content for community detection usually achieves better performance, as revealed in studies [8, 11]. PHITS-PLSA combines PHITS with PLSA for community detection [8]. E. Erosheva et al. [10] combine LDA with LDA-Link for network analysis, referred to as LDA-Link-Word model in this paper. R. Nallapti et al. [21] combine the mixed membership stochastic block model with LDA, and extend the LDA-Link-Word model by separating the citing documents and cited documents with LDA-Link-Word model on the citing documents and PLSA model on the cited documents. Other approaches that exploit LDA for combining link and content analysis include [9, 13]. One major problem with these approaches is that they apply a generative model for content analysis, which makes them vulnerable to the irrelevant keywords. In addition to probabilistic models, some other approaches that have been proposed to combine link and content information include matrix factorization[35] and kernel fusion[34] for spectral clustering.

3. CONDITIONAL LINK MODEL

In this section, we first present the proposed link model and followed by a maximum likelihood estimation method used to estimate the unknown parameters of the proposed model. In Section 4, we incorporate the content information into the proposed link model by a discriminative model.

3.1 Popularity-based Conditional Link Model (PCL)

Before going to the mathematical model, we first establish the assumptions and notations that are used in our model. All nodes in the network form a node space $\mathcal{V} = \{1, \cdots, n\}$, where the nodes could represent web pages, online blogs, etc. For each pair of ordered nodes (i, j), let s_{ij} record the information of the link from node i to node j. s_{ij} could either be $\{0, 1\}$, N^+, or any nonnegative values dependent on the type of the link. If $s_{ij} \neq 0$, we say there is a directional link from node i to node j, or node i cites j (equivalently, node j is cited by node i). Let $\mathcal{E} = \{(i \to j)|s_{ij} \neq 0\}$ denote all the directional links in the network. Each node i has an associated "link-in" space denoted by $\mathcal{LI}(i) \in \mathcal{V}$, which is the set of nodes that could possibly cite node i. Similarly, each node i is associated with a "link-out" space denoted by $\mathcal{LO}(i) \in \mathcal{V}$, which is the set of nodes that could possibly been cited by node i. Although in most cases we have $\mathcal{LI}(i) = \mathcal{LO}(i) = \mathcal{V}$, in some scenarios such as citation of publications, the link-out space of a paper is the set of all papers that are older than the paper itself, and the link-in space is the set of all papers that are newer than the paper itself. Let $\mathcal{I}(i) = \{j|s_{ji} \neq 0\}$ be the set of nodes that actually cite node i, $\mathcal{O}(i) = \{j|s_{ij} \neq 0\}$ be the set of nodes that are actually cited by node i, and $d_{in}(i) = |\mathcal{I}(i)|$ be the indegree of node i, $d_{out}(i) = |\mathcal{O}(i)|$ be the outdegree of node i. Finally, we denote by K the number of communities we aim to find.

In our link model, we focus on modeling $\Pr(j|i)$, i.e., the probability of linking node i to node j among all the other candidates in $\mathcal{LO}(i)$. In other words, we model which node j among $\mathcal{LO}(i)$ is more likely to be cited by node i. This is in contrast to many existing approaches that explicitly model the presence or absence of link $i \to j$, i.e., $\Pr(i \to j)$. This modeling choice allows us to avoid modeling the absence of links, which was observed in [2, 19] as a major problem for link analysis. We introduce a set of hidden variables $z_i \in \{1, \cdots, K\}$ for each node $i \in \{1, \cdots, n\}$ to denote the community of node i. On the other hand, to model how likely a node will receive a citation in general, in our model for $\Pr(j|i)$, we introduce a popularity variable $b_i \geq 0$ for each node i: the higher popularity of one node, the higher chance the node will be cited by other nodes. Given the popularity and community memberships of all nodes, the link probability $\Pr(j|i)$ conditioned on the community variable z_i of node i associated with this link is given as follows

$$\Pr(j|i; z_i, b) = \frac{\gamma_{jz_i} b_j}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'z_i} b_{j'}} \tag{1}$$

where γ_{ik} gives the community membership of node i in community k. As indicated by the above expression, the conditional link probability $\Pr(j|i)$ is proportional to b_j, the popularity of the ending node of the link. By assuming a multinomial distribution for z_i, i.e., $z_i \sim Mult(\gamma_{i1}, \cdots, \gamma_{iK})$, we have $\Pr(j|i)$ written as

$$\Pr(j|i; \gamma, b) = \sum_k \gamma_{ik} \frac{\gamma_{jk} b_j}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k} b_{j'}} \tag{2}$$

where $\gamma_{ik} = \Pr(z_i = k)$.

In Eq. (2), we assume that b_i is independently from the community variable. As a result, each node will only have one copy of the popularity. An alternative approach is to have the popularity variable b_i conditioned on the community variable. In other words, we have a different popularity variable b_{ik} for each node i when it is in a different community $z_i = k$. Using the community dependent popularity b_{ik}, $\Pr(j|i)$ is computed as

$$\Pr(j|i; z_i, b) = \frac{\gamma_{jz_i} b_{jz_i}}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'z_i} b_{j'z_i}}$$

or by integrating out z_i

$$\Pr(j|i; b) = \sum_k \gamma_{ik} \frac{\gamma_{jk} b_{jk}}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k} b_{j'k}} \tag{3}$$

Comparing Eq. (3) to (2), we see that Eq. (3) introduces the freedom of modeling the community dependent popularity at the price of increasing number of variables. As will be shown in our empirical study, Eq. (2) achieves better performance because of the reduced number of variables.

3.2 Analysis of the PCL Model

In this section, we analyze our link model by establishing the relation and comparing to PHITS model [7]. For the purpose of consistency, we assume $\mathcal{LO}(i) = \mathcal{V}$ for all i.

In PHITS, each community is assumed to have a multinomial distribution that specifies the probability for each node to be cited by the other nodes in the same community. We denote by β_{jk} the probability for node j to be cited by any nodes in the k^{th} community. $\Pr(j|i)$ conditioned on community variable z_i of node i for this link, and β is then expressed as

$$\Pr(j|i; z_i, \beta) = \beta_{jz_i}$$

Note that unlike our model in Eq. (1), the conditional link probability in PHITS model has nothing to do with the community membership of node j. This leads to the problem of undetermined community membership for nodes that do not cite any other nodes for PHITS, as discussed in the next section. By integrating out z_i, we have $\Pr(j|i)$ written as

$$\Pr(j|i; \gamma, \beta) = \sum_k \gamma_{ik} \beta_{jk} \tag{4}$$

where γ_{ik} is the probability that node i is in the kth community.

The following proposition allows us to establish the relationship between the PHITS model and the popularity-based conditional link model.

PROPOSITION 1. *The PHITS model specified in Eq. (4) is equivalent to the link model with* $\Pr(j|i)$ *specified in Eq. (3).*

The above proposition is proved by observing the link between β_{jk} and the quantity $\gamma_{jk} b_{jk} / \left(\sum_{j'} \gamma_{j'k} b_{j'k}\right)$. As revealed by the above proposition, PHITS is in fact a relaxed version of the proposed PCL model by assuming that the popularity of each node depends on the community of the node.

We can also derive the proposed model in Eq. (2) from the PHITS model in Eq. (4) by considering the relationship between γ_{jk} and β_{jk}, as revealed by the following proposition.

PROPOSITION 2. *The popularity-based conditional link model specified in Eq. (2) is equivalent to the PHITS model specified in Eq. (4) if β_{jk} is interpreted as $\Pr(j|C_k)$, i.e., the probability of selecting node j from the k^{th} community.*

The above proposition follows the Bayes's rule, i.e.,

$$\Pr(j|C_k) = \frac{\Pr(C_k|j)\Pr(j)}{\sum_{j'}\Pr(C_k|j')\Pr(j')} = \frac{\gamma_{jk}b_j}{\sum_{j'}\gamma_{j'k}b_{j'}}$$

The above proposition once again reveals that the proposed conditional link model is a restricted version of the PHITS model. We believe that it is the constraints introduced in the proposed conditional link model that lead to more reliable performance.

3.3 Maximum Likelihood Estimation

In this section, we present the method of maximum likelihood for the PCL model specified in Eq. (2). Observing the directional links $\mathcal{E} = \{(i \to j)|s_{ij} \neq 0\}$, we write the log-likelihood as

$$\log \mathcal{L} = \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij} \log \sum_k \gamma_{ik} \frac{\gamma_{jk}b_j}{\sum_{j'\in\mathcal{LO}(i)}\gamma_{j'k}b_{j'}} \tag{5}$$

where $\hat{s}_{ij}$ is normalized s_{ij} such that $\sum_{j\in\mathcal{LO}(i)}\hat{s}_{ij} = 1$. We find optimal γ and b by maximizing the log-likelihood

$$\begin{aligned}\max_{\gamma,b} \quad & \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij} \log \sum_k \gamma_{ik} \frac{\gamma_{jk}b_j}{\sum_{j'\in\mathcal{LO}(i)}\gamma_{j'k}b_{j'}} \\ s.t. \quad & \sum_k \gamma_{ik} = 1, \gamma_{ik} \geq 0, b_i \geq 0\end{aligned}$$

To derive the EM algorithm, we first have the following lemma for a low bound for $\log \mathcal{L}$.

LEMMA 3. *The log-likelihood $\log \mathcal{L}$ in Eq. (5) at the t^{th} iteration is lower bounded as follows*

$$\begin{aligned}&\log \mathcal{L} \geq Q(b,\gamma;b^{t-1},\gamma^{t-1}) \\ &= \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk}\left(\log\gamma_{ik} + \log\gamma_{jk} + \log\frac{b_j}{\tau_{ik}}\right. \\ &\quad \left. +1 - \sum_{j'\in\mathcal{LO}(i)}\frac{\gamma_{j'k}b_{j'}}{\tau_{ik}}\right) - \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij}\sum_k q_{ijk}\log q_{ijk}\end{aligned}$$

where the parameters τ_{ik} and q_{ijk} are computed as

$$\tau_{ik} = \sum_{j'\in\mathcal{LO}(i)} \gamma_{j'k}^{t-1} b_{j'}^{t-1} \tag{6}$$

$$q_{ijk} \propto \gamma_{ik}^{t-1}\frac{\gamma_{jk}^{t-1}b_j^{t-1}}{\sum_{j'\in\mathcal{LO}(i)}\gamma_{j'k}^{t-1}b_{j'}^{t-1}} \quad s.t. \sum_k q_{ijk} = 1 \tag{7}$$

and b^{t-1}, γ^{t-1} are the corresponding solutions in the $t - 1^{th}$ iteration.

The above lemma follows from the Jensen's inequality and the inequality of $-\log x \geq 1 - x$. Using the result in the above lemma, we search for b and γ at the t^{th} iteration that maximize the lower bound of $\log \mathcal{L}$, i.e.,

$$\max_{\gamma,b} \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij}\sum_k q_{ijk}\left(\log\gamma_{ik}\gamma_{jk}b_j - \sum_{j'\in\mathcal{LO}(i)}\frac{\gamma_{j'k}b_{j'}}{\tau_{ik}}\right) \tag{8}$$

$$s.t. \sum_k \gamma_{ik} = 1, \gamma_{ik} \geq 0, b_i \geq 0$$

For this maximization problem, we have the following theorem. Before stating the theorem, we first establish the notations for the purpose of representation:

$$\begin{aligned}n_{in}(i,k) &= \sum_{j\in\mathcal{I}(i)}\hat{s}_{ji}q_{jik} & n_{out}(i,k) &= \sum_{j\in\mathcal{O}(i)}\hat{s}_{ij}q_{ijk} \\ n_{in}(i) &= \sum_k n_{in}(i,k) & n_{out}(i) &= \sum_k n_{out}(i,k) \\ n(i,k) &= n_{in}(i,k) + n_{out}(i,k) & m(i,k) &= \sum_{j\in\mathcal{LI}(i)}\frac{n_{out}(j,k)}{\tau_{jk}}\end{aligned}$$

THEOREM 4. *The optimal solution to Eq. (8) satisfies the following conditions*
$\forall i, d_{out}(i) \neq 0, d_{in}(i) \neq 0,$

$$\gamma_{ik} = \frac{n(i,k)}{m(i,k)b_i + n_{out}(i)}, \quad b_i = \frac{n_{in}(i)}{\sum_k m(i,k)\gamma_{ik}} \tag{9}$$

$\forall i, d_{out}(i) = 0, d_{in}(i) \neq 0,$

$$\gamma_{ik} \propto \frac{n_{in}(i,k)}{m(i,k)}, \quad b_i = \frac{n_{in}(i)}{\sum_k m(i,k)\gamma_{ik}}$$

$\forall i, d_{out}(i) \neq 0, d_{in}(i) = 0,$

$$\gamma_{ik} = \frac{n_{out}(i,k)}{\sum_k n_{out}(i,k)}, \quad b_i = 0$$

$\forall i, d_{out}(i) = 0, d_{in}(i) = 0,$

γ_{ik} *is any non-negative value such that* $\sum_k \gamma_{ik} = 1, \quad b_i = 0$

Due to the limit of space, we skip the detailed proof of the theorem.

Remark: As revealed in Eq. (9), b_i is proportional to the number of nodes that cites node i, i.e., $n_{in}(i)$, which is consistent with interpreting b_i as "popularity" or "authoritative" for node i. Advantage of PCL over PHITS can also be seen in the solution of γ_{ik}. It can be shown that the membership of node i in PHITS model only depends on the membership of the nodes that are cited by node i, i.e., $\gamma_{ik} \propto n_{out}(i,k)$, and not affected by the nodes that cite node i. When $n_{out}(i) = 0$, i.e., node i has no outgoing links, the membership γ_{ik} is not determined. In contrast, in PCL model, community membership of node i depends on the membership of all the nodes connected to node i.

4. INCORPORATING CONTENT VIA A DISCRIMINATIVE MODEL

In this section, we extend our link model to incorporate the content information of nodes. As we discussed in Sections 1 and 2, most existing approaches combine link and content by a generative model that generates both links and content attributes via a shared set of hidden variables related to community memberships. In this work, we propose a discriminative model, referred to as Discriminative Content(DC) model, to incorporate the content into the proposed link model. Let $x_i \in \mathbb{R}^d$ denote the content vector of node i. The content information is used to model the memberships of nodes by a discriminative model, given by

$$\Pr(z_i = k) = y_{ik} = \frac{\exp(w_k^T x_i)}{\sum_l \exp(w_l^T x_i)} \tag{10}$$

where $w_k \in \mathbb{R}^d$ is a d-dimensional weight vector for community k with each element corresponding to each attribute. We can see that by incorporating the content model, the community membership is no longer specified by parameters γ_{ik}, but rather conditioned on the content through y_{ik} by a softmax transformation. Then, the conditional link probability $\Pr(j|i)$ expressed in Eq. (2) is modified as follows

$$\Pr(j|i; b, w) = \sum_k y_{ik} \frac{y_{jk} b_j}{\sum_{j' \in \mathcal{LO}(i)} y_{j'k} b_{j'}}$$

where y_{ik} depends on w as given in Eq. (10). As revealed in the above expression, we do not generate the content attributes as most topic models do. Instead, by using the discriminative model, with an appropriately chosen weight vector w_k that assign large weights to important attributes and small weights or zero weights to irrelevant attributes, we avoid the shortcoming of the generative models, i.e., being misled by irrelevant attributes. Another benefit from the discriminative model is that we can use a non-linear transformation $\phi(x) : \mathbb{R}^d \rightarrow \mathbb{R}^m$ on the content vector as the new attribute to obtain a non-linear model. In the sequel, we use $\phi(x)$ rather than x for presentation.

The log-likelihood of the combined model is written as

$$\log \mathcal{L} = \sum_{(i \rightarrow j) \in \mathcal{E}} \hat{s}_{ij} \log \sum_k y_{ik} \frac{y_{jk} b_j}{\sum_{j' \in \mathcal{LO}(i)} y_{j'k} b_{j'}} \tag{11}$$

We maximize the log-likelihood over the free parameters w and b. Although we can use any gradient-based algorithm to optimize with w_k and b_i, we propose an efficient two-stage method as discussed in the next section, which helps us better understand the relation of link model and content model.

A Two-Stage Method for Optimization

In this section, we describe the method to maximize the log-likelihood in Eq. (11). We still use the EM algorithm to maximize the log-likelihood. In the E-step, we compute τ_{ik} and q_{ijk} from y and b. In the M-step, we maximize the following problem:

$$\max_{w,b} \sum_{(i \rightarrow j) \in \mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk} \left(\log y_{ik} y_{jk} b_j - \sum_{j' \in \mathcal{LO}(i)} \frac{y_{j'k} b_{j'}}{\tau_{ik}} \right) \tag{12}$$

subject to non-negative constraints on b.

Instead of maximizing over w, we convert Eq. (12) into a constraint optimization problem over y and b by

$$\max_{y \in \Delta, b} \sum_{(i \rightarrow j) \in \mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk} \left(\log y_{ik} y_{jk} b_j - \sum_{j' \in \mathcal{LO}(i)} \frac{y_{j'k} b_{j'}}{\tau_{ik}} \right) \tag{13}$$

where the domain Δ is defined as

$$\Delta = \left\{ y | \exists w, y_{ik} = \frac{\exp(w_k^T \phi(x_i))}{\sum_l \exp(w_l^T \phi(x_i))} \right\} \tag{14}$$

By having the domain of y given in Eq. (14) as a convex set, we can take a projection method to maximize the problem of Eq. (13), which leads to the two-stage method. In the first stage, we simply ignore the complex constraint for y_{ik} imposed by the domain Δ and solve the optimization problem in Eq. (13) with only sum-to-one constraint on y_{ik} and non-negative constraints on b using the result in Theorem 4. In the second stage, we project the y_{ik} into the domain Δ. Let $\tilde{y}_{ik}$ denote the optimal solution obtained from the first stage. The projection of $\tilde{y}_{ik}$, denoted by y_{ik}, is obtained by minimizing the KL divergence between $\tilde{y}_{ik}$ and $y_{ik} \in \Delta$, which is equal to the following optimization problem

$$\max_w \sum_i \sum_k \tilde{y}_{ik} \log y_{ik} = \sum_i \sum_k \tilde{y}_{ik} \log \frac{\exp(w_k^T \phi(x_i))}{\sum_l \exp(w_l^T \phi(x_i))}$$

This problem is similar to the log-likelihood in multi-class logistic regression problem except that the class membership $\tilde{y}_{ik}$ is not just binary but between 0 and 1 . As in logistic regression, we can add regularization term on w_k to make the solution more robust, which leads to the following optimization problem

$$\max_w \sum_i \sum_k \tilde{y}_{ik} \log \frac{\exp(w_k^T \phi(x_i))}{\sum_l \exp(w_l^T \phi(x_i))} - \frac{\lambda}{2} \sum_k w_k^T w_k \tag{15}$$

where λ is the regularization coefficient. This problem is a convex problem [4] and has a unique optimal solution, and can be maximized efficiently by the Newton-Raphson method.

Algorithm 1 Algorithm for maximizing the log-likelihood

```
1. Input the number of iterations or convergence rate
2. Initialize $w_k$ to zeros, $b_i$ randomly, $\lambda$ to a fixed value
3. in the E-step, compute $\tau_{ik}$ and $q_{ijk}$ as in Eq. (6) and (7) using $y_{ik}$ rather than $\gamma_{ik}$
4. in the M-step,
     • compute $\gamma_{ik}$, and $b_i$ as in Theorem 4
     • update $w_k$ by maximizing the objective in Eq. (15) with $\gamma_{ik}$ in place of $\tilde{y}_{ik}$, and then compute $y_{ik}$
5. repeat Step 3 and  4 until the input number of iterations is exceeded or convergence rate is satisfied.
6. Output $\gamma_{ik}$ or $y_{ik}$ as the final membership
```

By converting the optimization problem over w into the problem over y and taking the two-stage method, we are able to have a better understanding of our combined model—the link structure will first give us a noisy estimation of community memberships $\tilde{y}$, and the noisy memberships are then used as supervised information for our discriminative content model to derive high-quality memberships y. These estimated memberships are further used in our EM iterations. Algorithm 1 summarizes the overall algorithms for combined link and content analysis for community detection. The algorithm has a time complexity of $\mathcal{O}(M(eKC_1 + nKC_2 + T_3))$, where M is the number of iterations, e is the number of links in the network, n is the number of nodes in the network, C_1 is a constant factor in computing q_{ijk} and τ_{ik}, C_2 is a constant factor in computing γ_{ik} and b_i, and T_3 is the time for maximizing problem in Eq. (15) by the Newton-Raphson method.

5. EXTENSIONS

In this section, we discuss two variants of the proposed framework for combining link information with content information. In the first variant, referred to as **PCL+PLSA**, we present an approach that combines the proposed conditional link model with the PLSA model for content analysis. In the second variant, referred to as **PHITS+DC**, we present an approach that combines the PHITS model for link analysis with the proposed discriminative approach for content analysis. These two combined models will serve as baselines in our experimental study.

5.1 PCL + PLSA

Similar to [8] where the PHITS link model is combined with PLSA content model, we combine our PCL link model with PLSA. The combined log-likelihood is given by

$$\log \mathcal{L} = \alpha \sum_i \sum_{j \in \mathcal{W}(i)} \hat{s}^w_{ij} \log \sum_k \beta^w_{jk} \gamma_{ik}$$
$$+ (1-\alpha) \sum_i \sum_{j \in \mathcal{O}(i)} \hat{s}^l_{ij} \log \sum_k \gamma_{ik} \frac{\gamma_{jk} b_j}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k} b_{j'}}$$

where α is combination coefficient, $\hat{s}^w_{ij}$ is the normalized number of times that word j occurs in the content of node i, $\mathcal{W}(i)$ denotes the set of unique words that occur in the content of node i, and $\beta^w_{jk} = \Pr(\text{word } j | C_k)$. To maximize the log-likelihood, we derive the EM-algorithm as follows. In the E-step, we compute q^w_{ijk}, q^l_{ijk} and τ_{ik} as

$$q^w_{ijk} \propto \gamma_{ik} \beta^w_{jk}, \quad s.t. \sum_k q^w_{ijk} = 1$$
$$\tau_{ik} = \sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k} b_{j'}$$
$$q^l_{ijk} \propto \gamma_{ik} \frac{\gamma_{jk} b_j}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k} b_{j'}}, \quad s.t. \sum_k q^l_{ijk} = 1$$

In the M-step, we compute β^w_{jk}, γ_{ik} and b_i as

$$\beta^w_{jk} = \frac{\sum_{i \in \mathcal{N}(j)} \hat{s}^w_{ij} q^w_{ijk}}{\sum_j \sum_{i \in \mathcal{N}(j)} \hat{s}^w_{jk} q^w_{ijk}} = \frac{n^w_{in}(j,k)}{\sum_j n^w_{in}(j,k)}$$
$$\gamma_{ik} = \frac{\alpha n^w_{out}(i,k) + (1-\alpha) n^l(i,k)}{\alpha n^w_{out}(i) + (1-\alpha)\left(n^l_{out}(i) + b_i m^l(i,k)\right)}$$
$$b_i = \frac{n^l_{in}(i)}{\sum_k m^l(i,k) \gamma_{ik}}$$

where $\mathcal{N}(j)$ denotes the set of nodes whose content have the word j, and n^w_{in}, n^w_{out}, n^l_{in}, n^l_{out}, n^l, and m^l are defined similar as before.

5.2 PHITS + DC

In this variant, we combine the PHITS link model with our DC content model. The log-likelihood is given by

$$\log \mathcal{L} = \sum_{(i \to j) \in \mathcal{E}} \hat{s}_{ij} \log \sum_k y_{ik} \beta_{jk}$$

where $y_{ik} = \exp(w_k^T \phi(x_i)) / \sum_l \exp(w_l^T \phi(x_i))$. In the E-step, we compute q_{ijk} as

$$q_{ijk} \propto y_{ik} \beta_{jk}, \quad \sum_k q_{ijk} = 1$$

In the M-step, we first compute β_{jk} and the free form membership γ_{ik} by

$$\beta_{jk} = \frac{\sum_{i \in \mathcal{I}(i)} \hat{s}_{ij} q_{ijk}}{\sum_j \sum_{i \in \mathcal{I}(i)} \hat{s}_{ij} q_{ijk}} = \frac{n_{in}(j,k)}{\sum_j n_{in}(j,k)}$$
$$\gamma_{ik} = \frac{\sum_{j \in \mathcal{O}(j)} \hat{s}_{ij} q_{ijk}}{\sum_k \sum_{j \in \mathcal{O}(j)} \hat{s}_{ij} q_{ijk}} = \frac{n_{out}(i,k)}{n_{out}(i)}$$

Then we maximize the following objective to get w_k and y_{ik},

$$\max \sum_k \sum_i \gamma_{ik} \log y_{ik} - \frac{\lambda}{2} \sum_k w_k^T w_k$$

6. EXPERIMENT

In this section, we conduct several experimental studies. We first compare the PCL model with the PHITS model for the task of link prediction. Then we compare the performance of the PCL model with that of several state-of-the-art methods on the task of community detection by using two citation data sets. Before going into the details, we first describe the data sets and the metrics used in the experiment and evaluation.

6.1 Data Sets

We used four data sets in our experiments. They are described in the following:

Political Blog Data Set is a social blog network, which is a directed network of hyperlinks between webblogs about the US political issues, recorded in 2005 by Adamic and Glance [1]. There are totally 1490 blogs, and each blog is labeled as either conservative or liberal. In the data set, we only have the link information and have no content information. So this data set is only used in the link prediction task to compare the PCL model with the PHITS model. The number of communities for this data set is set to $K = 2$.

Wikipedia Data Set is a web page network which was crawled from Wikipedia web site by Gruber et al. [13]. This data set has 105 nodes and 799 links. This data set contains no explicit community label for each page. So we only use this data set in the link prediction task, with K set to 20 as suggested in [13].

Cora Data Set is a subset of the larger Cora citation data set [20]. This data set includes publications from the machine learning area, each of which is classified into 7 subcategories as: Case-based reasoning, Genetic Algorithms, Neural Networks, Probabilistic Methods, Reinforcement Learning, Rule Learning and Theory. There are totally 2708 nodes, and 5429 links. Each node corresponds to one paper and is described by a 0/1-valued word vector indicating the absence/presence of the corresponding word from the dictionary of 1433 unique words. We use this data set in both the link prediction task and the community detection task. The number of communities is set to be $K = 7$.

Citeseer Data Set is a subset of the larger Citeseer data set[1]. The Citeseer data set consists of 3312 scientific publications labeled as one of 6 classes and 4732 links. Each

[1] `http://citeseer.ist.psu.edu/`

publication is described by a 0/1 valued word vector. The dictionary of word consists of 3703 unique words. This data set is used in both link prediction and community detection tasks. The number of communities is set to be $K = 6$.

6.2 Performance Metrics

In the comparison of the PCL model and the PHITS model on the task of link prediction, we hide some links from the network, and run the two models on the remaining links. The performance is measured by the metric of *Recall*.

Recall is an Information Retrieval measure. For each node, we compute the probabilities for the node to generate links to the other nodes and then sort these probabilities in the decreasing order. The recall is computed at each position in the rank and defined as the fraction of target nodes that correspond to the hidden links. The recall is reported from positions 1 to 20 in the rank.

To measure the performance of community detection, we used four metrics among which two are supervised and the other two are unsupervised. The two supervised metrics are *normalized mutual information (NMI)*, and *pairwise F-measure (PWF)*. These two metrics use the supervised label information. The other two unsupervised metrics are *modularity (Modu)* and *normalized cut (NCut)*. These two metrics measure the partition performance in terms of the link structure.

With the supervised label information, we can form the true community structure $\mathcal{C} = \{C_1, \ldots, C_K\}$, where C_k contains the set of nodes that are in the kth community. The community structure given by the algorithms is represented by $\mathcal{C}' = \{C'_1, \ldots, C'_K\}$. Then the *mutual information* between the two is defined as

$$\widehat{MI}(\mathcal{C}, \mathcal{C}') = \sum_{C_i, C'_j} p(C_i, C'_j) \log \frac{p(C_i, C'_j)}{p(C_i)p(C'_j)}$$

and the *normalized mutual information* is defined by

$$NMI(\mathcal{C}, \mathcal{C}') = \frac{\widehat{MI}(\mathcal{C}, \mathcal{C}')}{\max(H(\mathcal{C}), H(\mathcal{C}'))}$$

where $H(\mathcal{C})$ and $H(\mathcal{C}')$ are the entropies of the partitions $\mathcal{C}$ and $\mathcal{C}'$. The higher the normalized mutual information, the closer the partition is to the ground truth.

Let T denote the set of node pairs that have the same label, S denote the set of node pairs that are assigned to the same community, $|T|$ denote the cardinality of set T. The *pairwise F-measure* is computed from the pairwise precision and recall, as the following

$$precision = |S \bigcap T|/|S| \quad recall = |S \bigcap T|/|T|$$

$$PWF = \frac{2 \times precision \times recall}{precision + recall}$$

The higher the PWF, the better is the partition.

Modularity is proposed by Newman et al. [24] for measuring community partitions. For a given community partition $\mathcal{C} = \{C_1, \ldots, C_K\}$, the modularity is defined as

$$Modu(\mathcal{C}) = \sum_k \left[\frac{Cut(C_k, C_k)}{Cut(\mathcal{C}, \mathcal{C})} - \left(\frac{Cut(C_k, \mathcal{C})}{Cut(\mathcal{C}, \mathcal{C})} \right)^2 \right]$$

where $Cut(C_i, C_j) = \sum_{p \in C_i, q \in C_j} w_{pq}$. As stated in [24], modularity measures how likely a network is generated due to the proposed community structure versus generated by a random process. Therefore, a higher modularity value indicates a community structure that better explains the observed network.

Normalized cut is the objective of the normalized cut algorithm ([31], which we refer to as NCUT). Given a community partition $\mathcal{C} = \{C_1, \ldots, C_K\}$, the normalized cut is defined as

$$NCut(C_1, \cdots, C_k) = \sum_{i=1}^{K} \frac{Cut(C_i, \bar{C}_i)}{vol(C_i)}$$

where $\bar{C}_i$ denotes the set of nodes that are not in C_i and $vol(C_i) = \sum_{p \in C_i} \sum_q w_{pq}$.

6.3 PCL vs. PHITS

To validate the advantage of the PCL link model over the PHITS link model, we experiment them on the four data sets described in Section 6.1. The performance is reported in Figure 1 in terms of recall at positions 1 to 20. Each number in the figure is averaged over 5 runs. The PCL outperforms the PHITS in all the cases. To investigate the effects of the popularity parameter, b, we also perform the same experiments on PCL by setting $b_i = 1$ for all i. The results are labeled as "PCL-b=1" in the figure. The performance given $b_i = 1$ is worse than PCL and PHITS. It further confirms the importance of the popularity parameter. Overall, this result validates our conjecture that the conditional link model outperforms the generative link model, at least for the task of link predication.

6.4 Partition Performance

In this section, we investigate the performance of our model on the task of community detection. We perform experiments on the two scientific publication date sets, which have both link and content information.

To validate the advantage of our proposed model, we compare it with several baselines. Based on what information is used, the algorithms are categorized into 3 classes:

Based on Link, we compare the following models: PHITS, PCL, LDA-Link, and Spectral Clustering (NCUT).

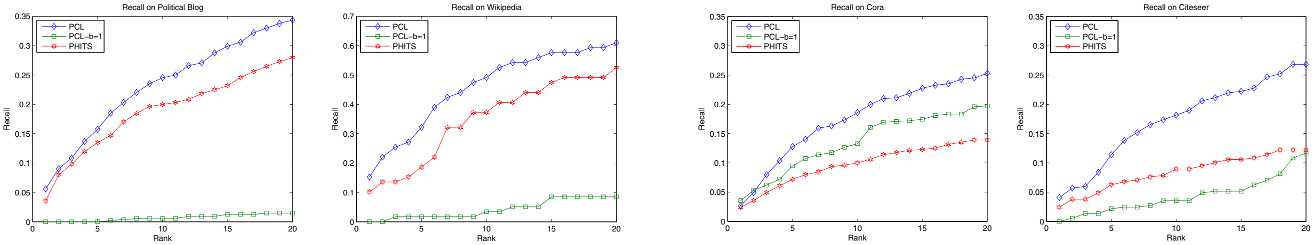
Based on Content, we compare the following: PLSA, LDA-Word, and Spectral Clustering. In spectral clustering, the similarity matrix is the kernel matrix computed from the content of each publication. Here we report two kernels, one is the RBF kernel, and the other is the probabilistic product kernel proposed in [16].

Based on Link and Content, we compare the following: PHITS-PLSA, LDA-Link-Word, Link-Content-Factorization (LCF), Spectral Clustering, PCL-PLSA, PHITS-DC, and PCL-DC. Notice that PHITS-PLSA refers to the combination of PHITS and PLSA proposed in [8], LDA-Link-Word refers to the mixed membership model proposed in [10], LCF refers to the model proposed in [35], Spectral Clustering is applied to linear combined kernel from the link matrix and content kernel, PCL-PLSA refers to the combination of the PCL and the PLSA model as described in Section 5, PHITS-DC refers to the PHITS model combined with the Discriminative Content model, and PCL-DC refers to the PCL model combined with the Discriminative Content model.

In the implementation, the feature vector used in our model is the original word indicator vector without any transformation; the spectral clustering we used is the normalized cut algorithm [31] (NCUT). For the algorithms that are de-

Table 1: Partition Measure on Cora and Citeseer dataset

		Cora				Citeseer			
	Algorithm	NMI	PWF	Modu	NCut	NMI	PWF	Modu	NCut
Link	PHITS	0.0570	0.1894	0.3929	3.2466	0.0101	0.1773	0.4588	2.2370
	LDA-Link	0.0762	0.2278	0.2189	4.5687	0.0356	0.2363	0.2211	3.7457
	PCL	0.0884	0.2055	0.5903	1.9391	0.0315	0.1927	0.6436	1.1181
	NCUT	0.1715	0.2864	0.2701	**0.2732**	0.1833	0.3252	0.6577	**0.1490**
Content	PLSA	0.2107	0.2864	0.2682	4.2686	0.0965	0.2298	0.2885	3.2294
	LDA-Word	0.2310	0.2774	0.2970	3.7820	0.1342	0.2880	0.3022	3.0165
	NCUT(RBF kernel)	0.1317	0.2457	0.1839	4.7775	0.0976	0.2386	0.2133	3.7078
	NCUT(pp kernel)	0.1804	0.2912	0.2487	4.6612	0.1986	0.3282	0.4802	1.8118
Link + Content	PHITS-PLSA	0.3140	0.3526	0.3956	3.2880	0.1188	0.2596	0.3863	2.7397
	LDA-Link-Word	0.3587	0.3969	0.4576	2.8906	0.1920	0.3045	0.5058	2.0369
	LCF	0.1227	0.2456	0.1664	4.8101	0.0934	0.2361	0.2011	3.6721
	NCUT(RBF kernel)	0.2444	0.3062	0.3703	1.6585	0.1592	0.2957	0.4280	1.7592
	NCUT(pp kernel)	0.3866	0.4214	0.5158	0.7903	0.1986	0.3282	0.4802	1.8118
	PCL-PLSA	0.3900	0.4233	0.5503	2.1575	0.2207	0.3334	0.5505	1.6786
	PHITS-DC	0.4359	0.4526	0.6384	1.5165	0.2062	0.3295	0.6117	1.2074
	PCL-DC	**0.5123**	**0.5450**	**0.6976**	1.0093	**0.2921**	**0.3876**	**0.6857**	0.7505



(a) Recall on Political Blog (b) Recall on Wikipedia (c) Recall on Cora (d) Recall on Citeseer

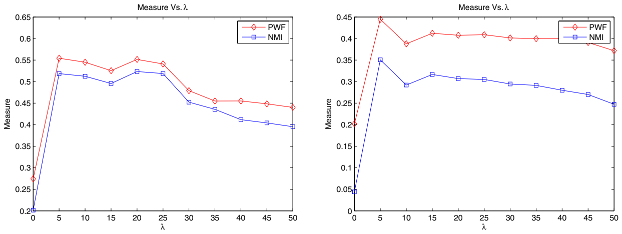
Figure 1: Recall on the four data sets

pendent on some parameters such as the σ parameter in RBF kernel, the combination coefficient in PHITS-PLSA, the combination coefficient of link matrix and content kernel for spectral clustering, the combination coefficient in PCL-PLSA, the regularization coefficient in PHITS-DC, we experiment on a wide range of values and choose the best one in terms of normalized mutual information and pairwise F-measure. For example, the combination coefficients in PHITS-PLSA, PCL-PLSA, and combined link matrix and content kernel are tuned from 0.1 to 0.9 with 0.1 as the step size. The regularization coefficient for PHITS-DC is tuned from 0 to 50 with 5 as the step size. The regularization coefficient for PCL-DC is set to a fixed value of 10. All the iterative algorithms are run until the relative difference of the objective is within 10^{-8}.

Tables 1 show the results on the Cora data set and the Citeseer data set. For both data sets, PCL outperforms PHITS in all the cases, either using link only (PCL outperforms PHITS), or combining link and content (PCL-PLSA outperforms PHITS-PLSA and PCL-DC outperforms PHITS-DC). When considering content, the approaches that discriminatively combine content (DC) outperform the approaches that combine content using PLSA. That is, PHITS-DC outperforms PHITS-PLSA, and PCL-DC outperforms PCL-PLSA. These results further confirm that the discriminative models (either the link model, or the content model, or the combination of the two) achieve better performance than the generative ones.

We also compared PCL and PCL-DC with the following algorithms. In the link-only case, the spectral clustering (NCUT) outperforms PCL. LDA-Link outperforms PCL in some metrics. When combining link and content, PCL-DC outperforms all algorithms except for the spectral clustering (NCUT) algorithm in the normalized cut (NCut) metric. The main reason for the spectral clustering (NCUT) to have the best performance in terms of normalized cut is that it directly minimizes this metric. However, we argue that people would consider the NMI and PWF metrics as equally important, because the NMI and PWF metrics measure how good the partition derived by the algorithms matches the ground truth.

Finally, to reveal the performance of our model under different parameters, we show the performance of the PCL-DC model under different regularization coefficient λ on the two data sets in Figure 2. In both data sets, the performance achieves the highest level when $\lambda = 5$. After that, the PCL-DC algorithm is not very sensitive to λ.



(a) Cora (b) Citeseer

Figure 2: Partition Measure of PCL-DC vs. λ

7. CONCLUSION

In this paper, we proposed a unified model to combine link and content analysis for community detection. To accurately model the link patterns, a conditional link model is proposed to capture the popularity of nodes. In order to alleviate the problem caused by the irrelevant attributes, a discriminative model, instead of a generative model, is proposed for modeling the contents of nodes. The link model and content model are combined via a probabilistic framework through the shared variables of community memberships. We observed that the combined model obtains significant improvement over the state-of-the-art approaches for community detection. For future work, we plan to consider a full Bayesian model to compute the posterior of membership and parameters rather than computing the maximum likelihood estimation, and try to look at the performance of the proposed model on more data sets.

Acknowledgement: We would like to thank Prof. Andrew McCallum for providing the Cora dataset and Prof. C. Lee Giles for providing the CiteSeer dataset. We also thank anonymous reviews for their valuable comments.

8. REFERENCES

[1] L. Adamic and N. Glance. The political blogosphere and the 2004 U.S. election: divided they blog. In *LinkKDD*, 2005.

[2] E. M. Airoldi, D. M. Blei, S. E. Fienberg, and E. P. Xing. Mixed membership stochastic block models for relational data with application to protein-protein interactions. In *IBS*, 2006.

[3] J. Baumes, M. Goldberg, and M. Magdon-ismail. Efficient identification of overlapping communities. In *IEEE ISI*, 2005.

[4] C. M. Bishop. *Pattern recognition and machine learning*. Springer-Verlag, 2006.

[5] D. M. Blei, A. Y. Ng, M. I. Jordan, and J. Lafferty. Latent dirichlet allocation. *J. of MLR*, 3, 2003.

[6] A. Clauset, M. E. J. Newman, and C. Moore. Finding community structure in very large networks. *Phy. Rev. E*, 70, 2004.

[7] D. Cohn and H. Chang. Learning to probabilistically identify authoritative documents. In *ICML*, 2000.

[8] D. Cohn and T. Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. In *NIPS*, 2001.

[9] L. Dietz, S. Bickel, and T. Scheffer. Unsupervised prediction of citation influences. In *ICML*, 2007.

[10] E. Erosheva, S. Fienberg, and J. Lafferty. Mixed membership models of scientific publications. *PNAS*, 101, 2004.

[11] L. Getoor, N. Friedman, D. Koller, and B. Taskar. Learning probabilistic models of link structure. *J. of MLR*, 3, 2002.

[12] S. Gregory. An algorithm to find overlapping community structure in networks. In *PKDD*, 2007.

[13] A. Gruber, M. Rosen-Zvi, and Y. Weiss. Latent topic models for hypertext. In *UAI*, 2008.

[14] J. M. Hofman and C. H. Wiggins. A Bayesian approach to network modularity. *Phy. Rev. Letters*, 100, 2008.

[15] T. Hofmann. Probabilistic latent semantic indexing. In *SIGIR*, 1999.

[16] T. Jebara, R. Kondor, A. Howard, K. Bennett, and N. Cesa-bianchi. Probability product kernels. *J. of MLR*, 5, 2004.

[17] J. M. Kleinberg. Authoritative sources in a hyperlinked environment. *J. of ACM*, 46, 1999.

[18] S. Lacoste-Julien, F. Sha, and M. I. Jordan. DiscLDA: Discriminative learning for dimensionality reduction and classification. In *NIPS*, 2008.

[19] A. McCallum and K. Nigam. A comparisoin of event models for naive bayes text classification. *AAAI Workshop*, 1998.

[20] A. McCallum, K. Nigam, J. Rennie, and K. Seymore. Automating the contruction of internet portals with machine learning. *J. of Info. Retr.*, 3, 2000.

[21] R. M. Nallapati, A. Ahmed, E. P. Xing, and W. W. Cohen. Joint latent topic models for text and citations. In *KDD*, 2008.

[22] M. E. J. Newman. Fast algorithm for detecting community structure in networks. *Phy. Rev. E*, 69, 2004.

[23] M. E. J. Newman. Modularity and community structure in networks. *PNAS*, 103, 2006.

[24] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Phy. Rev. E*, 69, 2003.

[25] M. E. J. E. Newman and E. A. A. Leicht. Mixture models and exploratory analysis in networks. *PNAS*, 104, 2007.

[26] A. Y. Ng, M. I. Jordan, and Y. Weiss. On spectral clustering: Analysis and an algorithm. In *NIPS*, 2001.

[27] K. Nowicki and T. A. B. Snijders. Estimation and prediction for stochastic blockstructures. *J. of ASA*, 96, 2001.

[28] L. Page, S. Brin, R. Motwani, and T. Winograd. The PageRank citation ranking: Bringing order to the web. In *Technical report, Stanford Digital Library Technologies Project, Stanford University, Stanford, CA, USA*, 1998.

[29] G. Palla, I. Derényi, I. Farkas, and T. Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435, 2005.

[30] W. Ren, G. Yan, X. Liao, and L. Xiao. Simple probabilistic algorithm for detecting community structure. *Phy. Rev. E.*

[31] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Trans. on PAMI*, 22, 2000.

[32] X. Wang, N. Mohanty, and A. McCallum. Group and topic discovery from relations and their attributes. In *NIPS*, 2005.

[33] K. Yu, S. Yu, and V. Tresp. Soft clustering on graphs. In *NIPS*, 2005.

[34] S. Yu, B. D. Moor, and Y. Moreau. Clustering by heterogeneous data fusion: framework and applications. *NIPS workshop*, 2009.

[35] S. Zhu, K. Yu, Y. Chi, and Y. Gong. Combining content and link for classification using matrix factorization. In *SIGIR*, 2007.