LEARNING FROM NOISILY CONNECTED DATA

By

Tianbao Yang

A DISSERTATION

Submitted to
Michigan State University
in partial fulfillment of the requirements
for the degree of

DOCTOR OF PHILOSOPHY

Computer Science

2012

ABSTRACT

LEARNING FROM NOISILY CONNECTED DATA

By

Tianbao Yang

Machine learning is a discipline of developing computational algorithms for learning predictive models from data. Traditional analytical learning methods treat the data as independent and identically distributed (i.i.d) samples from unknown distributions. However, this assumption is often violated in many real world applications that leading to the challenge of learning predictive models. For example, in electronic commerce website, customers could purchase a product by the recommendation of their friends. Hence the purchasement records of customers are not i.i.d samples but correlated.

Nowadays, data become correlated due to collaborations, interactions, communications, and many other types of connections. Effective learning from these connected data not only provides better understanding of the data but also brings significant economic benefits. How to learn from the connected data also brings unique challenges to both supervised learning and unsupervised learning algorithms because these algorithms are designed for i.i.d data and are often sensitive to the noise in the connected data.

In this dissertation, I focus on developing theory and algorithms for learning from connected data. In particular, I consider two types of connections: the first type of connection is naturally formed in real wold networks, while the second type of connection is manually created to facilitate the learning process which is called must-and-cannot link. In the first part of this dissertation, I develop efficient algorithms for detecting communities in the first type of connected data. In the second part of this dissertation, I develop clustering algorithms that effectively utilize both must links and cannot links for the second type of connected

data

A common approach toward learning from connected data is to assume that if two data points are connected, they are likely to be assigned to the same class/cluster. This assumption is often violated in real-word applications, leading to the noisy connection problems. One key challenge of learning from connected data is how to model the noisy pairwise connections that indicates the pairwise class-relationship between two data points. In the problem of detecting communities in networked data, I develop Bayesian approaches that explicitly model the noisy pairwise links by introducing additional hidden variables, besides community memberships, to explain potential inconsistency between the pairwise connections and pairwise class-relationship. In clustering must-and-cannot linked data, I will try to model how the noise is added into the pairwise connections in the manually generating process.

The main contributions of this dissertation include (i) it introduces *popularity* and *productivity* for the first time besides the community memberships to model the generation of noisy links in real networks; the effectiveness of these factors is demonstrated through the task of community detection; (ii) it proposes a discriminative model for the first time that combines the content and link analysis together for detecting communities to alleviate the impact of noisy connections in community detection; (iii) it presents a general approach for learning from noisily labeled data, proves the theoretical convergence results for the first time and applies the approach in clustering noisy must-and-cannot linked data.

ACKNOWLEDGMENTS

First and formost, I would like to thank my thesis advisor Dr. Rong Jin. I feel extremely fortunate to work with and learn from Dr. Rong Jin. His decision back to five years ago on extending me a graduate assistantship offer significantly changed my life. Without this offer, I have no idea where I am going and I can not image I can perform as well as what I did in the past five years. During the five years at MSU, Dr. Rong Jin gave me a great help and had a profound influence on my research. He always inspired me to talckle interesting problems and guided me to solve problems using mathematical skills. His enthusiasm on reading books brought me to a world of interesting and useful books from which I learned a lot in the past five years, and I have no doubt that this influence will continue in the future.

I would also like to thank other committee members Dr. Pang-ning Tan, Dr. Joyce Chai and Dr. Selin Aviyente. I am grateful to my collaborators from NEC Laboratories American and Yahoo Research. I spent three summers at NEC Labs working with Yun Chi and Shenghuo Zhu. I am greatly indebted to them for the help on my first several publications and for introducing me an interesting research topic of social network analysis. I also spent a wonderful summer at Yahoo Research mentored by Olivier Chapelle and Lihong Li. I enjoyed the discussion with Olivier and Lihong on research problems.

I want to thank my colleagues from LINKS lab at MSU, especially Wei Tong, Yang Zhou, Mehrdad Mahdavi, Fengjie Li, Yi Liu and Jinfeng Yi. I also want to thank my colleagues from NEC Labs during the three summers. I also owe great thanks to Ying Liu and I wish her all the best. I gratefully acknowledge the help from the stuff at CSE department of MSU, especially from Linda Moore and Norma Teague.

My final and the most heartfelt acknowledgment must go to my family for their love and support. I also want to specially thank Fangfang. She always patiently listens to me whenever I am feeling down.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

CHAPTER 1

Introduction

Most machine learning problems can be classified into three categories (1) **supervised learning** [92] that aims to learn a prediction function from a set of labeled instances. Most supervised learning tasks can be further divided into the category of classification and regression [38], depending on the type of outputs. (2) **unsupervised learning** [29] that aims to divide a given collection of instances into multiple clusters, with the hope that instances within the same cluster share a larger similarity than instances from different clusters. (3) **semi-supervised learning** [114] that aims to learn a prediction function from a mixture of labeled and unlabeled instances. Other learning paradigms include **reinforcement learning** [5] and **learning to learn** [88].

In classical setup of learning problems, training instances are usually represented by vectors of attributes (also referred to as content information), and in the case of supervised learning, labels are provided for individual training instances. In this dissertation, we study the class of learning problems where the additional relationship information is provided between pairs of instances. The pairwise relationship information, can either be the observed connections between training instances (e.g., hyperlinks between web pages) or the connections derived from side information (e.g., pairwise class-relationship). Training instances along with the physical connections can be represented by a network, and this type of training data is also referred to as **networked data**. Pairwise class-relationships are introduced by Wagstaff et al.[94] to specify the relationship between class assignments of two instances, which are also known as must-and-cannot(M&C) link information. Instances connected by must-link information are said to be in the same class, while instances connected by cannot-link information are in different classes. The pairwise class-relationship connections can be derived from the observed connections with some additional knowledge. We refer to the data that are connected by must-and-cannot link information as **must-and-cannot linked data** or **M&C linked data**. In this dissertation, we refer to both networked data and M&C linked data as **connected data**.

The objective of this study is to take advantage of the connections between instances, to facilitate the learning process. We thus refer to the related learning problems as learning from connected data. In particularly, the research work presented in this dissertation focuses on **unsupervised learning** on **networked data** and **M&C linked data**. The challenge of learning from connected data is how to discriminate and utilize useful connections from those noisy connections in indicating the pairwise class relationship. In the sequel, the discussion and presentation on connected data is divided into networked data and M&C linked data.

1.1 Networks

Networks, as a particular type of connected data, have become an important topic for computer science, physics, biology, communications, statistics, social science, psychology and etc. Many important properties have been studied for networks, including scale-free, small-word, community structure, triads and clustering coefficients, network motifs, etc. Below, we review the three most important properties of networks, i.e. scale-free, small-word, and community structure. For discussion of the other properties of networks, , we refer the reader to overviews in [4, 74, 55, 11, 12, 98].

A scale-free network refers to the network in which a few nodes have a tremendous number of connections to the rest nodes while most nodes in the network only have a handful connections. This property is also known as power-law degree distribution. The nodes with very large numbers of connections are usually referred to as hubs. Over that past several years, researchers have uncovered scale-free structures in a stunning range of systems, including Word Wide Web [6], social networks such as the sexual network[59], network of citations between scientific papers [23], celluar metabolic network [3], protein-protein interaction network [80]. One important characteristic of scale free networks is that they are robust to accidental failures but are vulnerable to coordinated attacks. Scale-free networks have found their applications in many domains, such as computing, medicine, and business. Computer networks with scale-free architectures, such as the world wide web, are highly resistant to accidental failures, but they are very vulnerable to deliberate attacks and sabotage. In medicine, vaccination campaigns against serious viruses, such as smallpox, might be most effective if they concentrate on treating hubs-people who have many connections to others. Mapping out the networks within the human cell could aid researchers in uncovering and controlling the side effects of drugs. Furthermore, identifying the hub molecules involved in certain disease could lead to new drugs that would target those hubs.

A small-world network is a type of network in which although most nodes are not directed connected with each other, most nodes can be reached from each other by a small number of hopes or steps. A typical small-world network is the network of acquaintance between people, where the famous six-degree separation phenomenon [90, 65] is well accepted. Other real wold networks that reveal the small-world properties include road maps [108], food chains [66], electric power grids [7], metabolic processing networks [93], networks of brain neurons [85], telephone call graphs [69], and protein networks [91], etc. In observation, if a network has a degree-distribution which can be fit with a power law distribution, it is taken as a sign that the network is small-world, which is also scale-free networks. Though, scale free networks usually reveal the small world property; however they are by no means the only kind of small-world networks. Networks of very different topology can still quantify as small-world networks as long as they satisfy the properties of the small-world networks.

Community structure is another important property of networks. It refers to the occurrence of groups of nodes in a network that are more densely connected internally than with the rest of the network. This inhomogeneity of connections suggests that the network has certain natural divisions within it. Community structures are quite common in real networks [77, 30, 67, 111, 36]. Social networks [30] often include community groups (the origin of the term, in fact) based on common location, interests, occupation, etc. Metabolic networks [71] have communities based on functional groupings. Citation networks [14] form communities by research topic. Being able to identify these sub-structures within a network can provide insight into how network function and topology affect each other. For more discussion on community structures in networks and methods for automatic community detection, we refer to the survey paper by Poter et al. [76], and the book by Mark Newman [73].

1.2 Must-and-Cannot Link

Must-and-Cannot link was introduced as supervised information for clustering by Wagstaff et al.[94]. It specifies the class relationship between instances. Pairs of instances connected by must-link must be in the same cluster and pairs of instances connected by cannot-link cannot be in the same cluster. These link information thus provide constraints on clustering results. Must-link and cannot-link have some interesting properties. Must-link constraint is an equivalence relation and hence are symmetrical, reflexive and transitive. For example, if instance x and instance y are connected by a must-link, and instance y and instance z are connected by a must-link, then instance x and instance z should also be connected by a must-link. Similarly, must-link constraints can give rise to cannot-link constraints. For example x and y are connected by a must-link , y and z are connected by a cannot-link, then x and z can also be connected by a cannot-link. Though simple, must-and-cannot link are powerful and have been used for improving clustering accuracy [95, 19]. For more discussion on must-and-cannot link and clustering for M&C linked data, we refer the reader to the survey paper by I. Davidson [20].

1.3 Research on leaning from connected data

Research on Networked Data

The study of networked data has attracted an enormous amount of interest in the last few years. Topics include the measurement and structure of networks, mathematical models of networks, theories of dynamical processes in networks, and prediction based on networks. We review some research topics as follows, but they are by no means the complete list.

- **Community Detection** aims to detect the community structure in networks [72, 34].

- **Dynamics of Networks** Networks, especially social networks and the web, are constantly evolving with additions and deletion of nodes and edges taking place all the time. Research on dynamics of networks concerns about the statistical properties and models that govern the generation and evolution of real-world networks [15, 60].

- **Diffusion and Cascading in Networks** Information cascades refers to the phenomena in which an action or idea becomes widely adopted due to influence by others. Studies on diffusion and cascading in networks seek to build models for the flow of information or influence through a large network [33, 49].

- **Statiscal Modeling** concerns about modeling the network from the viewpoint of statistics. It aims to learn a statistical model that generates the network as a whole based on the statistics collected in the network [99, 32].

- **Link Prediction** aims to infer new interactions among nodes of a given network that are likely to occur in the near future, given a snapshot of a social network [58, 87].

- **Link-based Classification** aims to improve classification accuracy by exploiting the link structure besides utilizing the content information of the nodes [62, 22].

Research on M&C linked Data

Research on M&C linked data focuses on constrained clustering, distance metric learning, and kernel learning.

- **Constrained Clustering** aims to improve the clustering accuracy by leveraging the must-link and cannot-link information[95, 19, 8, 81, 18, 53, 24, 18, 95].

- **Distance Metric Learning** aims to learn a distance metric that is consistent with the given M&C linked data, i.e. must-link pairs are separated by a shorter distance than cannot-link pairs [102, 82, 43, 31, 104, 100] .

- **Kernel Learning** aims to learn a kernel matrix or function from the given M&C linked data such that kernel similarity between must-link pairs is larger than kernel similarity between cannot link pairs [51, 42, 41, 56, 13, 46].

CHAPTER 2

Contributions of this Dissertation

This dissertation addresses a number of important problems regarding community detection for networked data, and clustering for M&C linked data.

Most algorithms developed for community detection for networked data are based on probabilistic models. The advantage of these approaches is that they are easy to understand. Furthermore, these models are able to predict links besides detecting community structures. However, there are two major problems with most probabilistic models for community detection. (i) The links between nodes are generated only based on their community memberships. However, in real networks, many more factors could affect the link generation. For example, a webpage could point to another webpage simply because it is a very popular webpage such as Google. (ii) The memberships are assumed to be random variables, independent of the content. However the community memberships are usually determined by the content of the nodes. We address the fist limitation by a link-based probabilistic model that takes into account other factors, such as popularity and productivity, when modeling the generation of links. We address the second limitation by presenting a discriminative approach to incorporate the content information. Below, we describe each of the contributions in detail:

- We propose a novel probabilistic model for link-based community detection in networks, referred to as **Popularity and Productivity Link Model for Community Detection** [105]. Besides community memberships, it introduces two important factors, namely popularity and productivity, for modeling the link generation. We verify both theoretically and empirically that the proposed model fits well with the power-law degree distribution of real networks, a key property of scale-free networks that is overlooked by previous probabilistic models for community detection. This is the first research work that is able to model the community structure and scale-free properties at the same time.

- We present a novel approach to combine link and content information for community detection in networks [106]. Unlike previous work that assume the community memberships are random variables inferred from the link, the proposed approach models the community memberships by the content information using a discriminative model, and infers the model parameters from the data. With the discriminative model, the proposed approach is able to identify the relevant attributes for each community from those irrelevant ones. This is the first work that explores a discriminative model for community memberships when incorporating the content information.

Most previous work on clustering for M&C linked data assumes that both must-link and cannot-link information is perfect. However, the M&C link information is usually derived from side information such as citations between scientific papers, and can be vulnerable to noise. To address this limitation, we present two approaches for clustering the noisy M&C linked data.

- Both approaches the noise issue by estimating the sufficient data statistics with perfect link information from the one with noise link information, and using the estimated sufficient statistics to learn a probabilistic link prediction model.

- The two approaches differ in the assumptions about the data and the noise. We study two different data models and noise models, and devise approaches for estimating the feature statistics under the two different models and compare them empirically.

- We verify, both theoretically and empirically, the effectiveness of the proposed approaches. This is the first work that provides theoretical analysis to show that the model learned from the noisy M&C link information converges to the model learned from the perfect M&C link information.

CHAPTER 3

Literature Survey

3.1 Community Detection for Networked Data

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 [96]. 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.

The goal of community detection is to find a set of communities such that the nodes in each community are densely connected with each other while sparsely connected with others. When the content information is available, we may also expect nodes within the same community to share similar contents. However, most existing studies on community detection only focus on either the link based analysis or the content analysis. Only very recently people started to combine these two types of information for community detection. In the following, we will review community detection methods in these three categories, i.e., link-based analysis, content-based analysis, and analysis based on the combination of link and content information.

3.1.1 Community Detection based on Link Analysis

Link-based approaches for community detection can be roughly classified into two categories, i.e., metric based approaches and probabilistic modeling based approaches. For metric-based approaches, a metric is first defined to measure the quality of any potential community structure; and then, procedures are developed to optimize the proposed metric. Some well-known metrics used for link based community detection include normalized cut proposed by Shi et al. [83], modularity proposed by Newman et al. [72], betweenness proposed by Gregory [34], etc. More recent work in this category starts to introduce probabilistic interpretations for some of these metrics and extend the metrics from undirected networks to directed networks [54, 112]. A main weak point with these metric-based approaches is the proposed metrics are usually heuristic and do not have solid foundations. As a result, it is difficult to reach a consensus on the optimal metric. These approaches are also limited in that they are unable to make prediction for unobserved links, which is important for a number applications.

The second category of approaches are based on probabilistic models. They first define a generative process, in which links are generated based on latent community memberships, and then, develop inference algorithms to derive the latent community memberships from the observed links. Holland et al. [45] proposed the first stochastic block model for community detection, and later on many variations [2, 68, 25, 28, 39, 48] have been proposed. In the stochastic block models, the probability of creating a link between two nodes is assumed to be a constant that depends on the community assignments of the two nodes. Most of the followup works essentially refine this probability. In probabilistic Hyperlink-Induced Topic Search (PHITS) model [16] and Latent Dirchilet allocation link model [10], the probability of observing a link from node i to j is given by the probability for j to be linked by any node from the community of i. In the graph factorization model [109, 79], this probability is interpreted as a joint probability of observing both nodes simultaneously, which is factorized based on the Bayes product rule. In terms of inference, some of these models [45, 16, 109, 79] are based on maximum likelihood estimation, while some are based on full Bayesian inference [2, 68, 25, 28, 39, 48, 10].

Given a large number of approaches have been developed in the family of stochastic block models, it is useful to further classify into subcategories: the *symmetric* approaches [79, 109] that model links by symmetric joint probabilities, and the *conditional* approaches [16, 106] that focus on modeling the conditional probability of receiving links. However, neither of these models is satisfying: a symmetric model misses the semantics of link directions, a key factor that distinguishes directed networks from undirected networks, while a conditional model only captures one type of links, either incoming links or outgoing links, and therefore is unable to characterize nodes in a full spectrum. As an example, in a blog readership network, there are two types of bloggers: "writers" who generate influential blogs read by many, and "readers" who read a lot but seldom write anything for others to read. Evidently, to characterize these two types of bloggers, it is important to examine both incoming links and outgoing links of the network.

3.1.2 Community Detection based on Content Analysis

Given the content description of nodes, it is appealing to cast community detection problem into a clustering problem. Many traditional vector-based clustering algorithm such as K-means [37], hierarchical clustering algoirthms [86, 97] have been applied to content analysis for community detection. In the subsection, we focus on the scenario that the content is represented as a bag-of-words, since nodes in many online networks are usually webpages. One of the most well-known approaches for such content information is the topic model[], where each topic is naturally interpreted as a community. Two well-known topic models are Probabilistic Latent Semantic Analysis(PLSA) [40] and Latent Dirichilet Allocation(LDA) [10]. In these models, each topic is modeled as a probability distribution over words. To generate a document, one first sample a topic from a prior distribution, and then sample words for the given topic. One key drawback of topic models is that they are generative probabilistic models and therefore are vulnerable to the words that are irrelevant to the target topics.

3.1.3 Community Detection based on Combined Link and Content Analysis

Neither link information nor content information is sufficient in detecting the community structure: 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 detection of communities. It is therefore important to combine the link analysis and content analysis for community detection in networks. Recently, several approaches [17, 28, 68, 113, 110] have been proposed to combine these two types of information for community detection. PHITS-PLSA combines PHITS, a link based approach, with PLSA, a content based approach, for community detection [17], and show a significantly improvement over the approaches that only utilizes link information or content information for community detection. E. Erosheva et al. [28] combine LDA with LDA-Link, an extension of PHITS, for network analysis, referred to as LDA-Link-Word model in this paper. R. Nallapti et al. [68] try to improve LDA-Link-Word model by applying different modeling strategies to citing documents from that to cited documents. More specifically, LDA-Link-Word model is applied to the citing documents and PLSA model is applied to the cited documents. Other approaches that exploit LDA for combining link and content analysis can be found in [25, 35]. Despite the significant efforts, the existing approaches for combining the link and content information are based on generative models, and therefore will suffer from the following 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 makes them vulnerable to the irrelevant keywords, and therefore leads to poor performance. In addition to probabilistic models, some other approaches that have been proposed to combine link and content information include matrix factorization[113] and kernel fusion[110] for spectral clustering. Most of them can be more or less viewed as generative models, according to the recent studies on the equivalence between mixture model, k-means, spectral clustering, and matrix factorization [26, 27]. As a result, these approaches usually yield similar performance as the generative models. Since the focus of this dissertation is probabilistic modeling approaches, we therefore did not review them in detail.

3.2 Clustering for Must-and-Cannot(M&C) Linked Data

In this section, we review the related work on clustering for must-and-cannot(M&C) linked data. Most approaches in this subject can be classified into three categories: constrained clustering, distance metric learning and kernel learning.

3.2.1 Constrained Clustering

Constrained clustering tries to improve the accuracy of data clustering by exploiting the M&C linked pairs, also termed as pairwise constraints. They are also known as semi-supervised clustering. In constrained clustering, the clustering algorithm is modified so that the given pairwise constraints are used to bias the search for an appropriate clustering of the data. There are two types of constrained clustering approaches: (1) ones with strict enforcement, which find the best feasible clustering satisfying all the given pairwise constraints [95, 19], and (2) ones with partial enforcement, which find the best clustering while maximally respecting constraints [8, 81, 18, 53]. Several techniques have been developed to fit the clustering results with respect to the constraints: (1) modifying the clustering objective function so that it includes a term (penalty) for satisfying specified constraints [24, 18]. (2) enforcing all constraints to be satisfied by the cluster assignments generated in each step of an iterative clustering algorithm [95]. More discussion on constrained clustering can be found in the [20], and the references therein.

3.2.2 Distance Metric Learning

In distance metric learning, an appropriate distance metric is learned so that must-link pairs are separated by a short distance, while cannot-link pairs are separated by a large distance. Given a learned distance metric, we can then apply the existing clustering algorithms, such as k-means, to find the clusters. Many algorithms [102, 82, 43, 31, 104, 100] have been developed for distance metric learning, such as distance metric learning by convex optimization [102], relevance component analysis [82], discriminative component analysis [43], nearest neighbor component analysis [31], local distance metric learning [104], large margin nearest neighbor classifier [100], information theoretic metric learning [21], distance function learning [89], and learning a Bregman distance function [101]. All these approaches propose to optimize an objective function about the distance metric by satisfying a set of equality/inequality constraints. The M&C link information serve in the objective [104], or the constraints [82, 21], or both [102, 100]. More work on distance metric learning from M&C link information can be found in the survey [103] and references therein.

3.2.3 Kernel Learning

In kernel learning, an appropriate kernel matrix/function for a given data set is learned from M&C link information, and then a similarity based clustering algorithm(e.g. spectral clustering) is applied with the learned kernel matrix. Recent work on kernel learning from M&C linked data include learning low rank kernel matrix [51], nonparametric kernel learning [42], active kernel learning [41], spectral kernel learning [56], Kernel-based metric adaptation [13], and semi-supervised kernel matrix learning by kernel propagation [46]. These approaches either propose a loss function that measures the inconsistency between a kernel function/matrix and pairwise constraints, and find the optimal kernels by minimizing the overall loss, or find a kernel maxtrix with certain property such as low rank to satisfy a set of equality/inequality constraints constructed using the M&C link information.

Most studies on clustering M&C linked data assume the link information is noise-free. However, in many applications, the M&C link information is derived from the side information like the observed connections among instances, making them prone to errors. For example, in classifying research articles with M&C link constructed from paper citations, the cited paper may not share the same research topic as the citing paper. It is important to consider the noise in the M&C link information, since direct learning from the given M&C link information wouldlead to suboptimal results.

CHAPTER 4

Community Detection for Networked Data: A Popularity and Productivity Link Model

4.1 Introduction

In this chapter, we propose a novel probabilistic framework for directed network community detection, termed **Popularity and Productivity Link** model or **PPL** for short, that explicitly addresses the shortcomings of the existing stochastic block models. In particular, we model both outgoing links and incoming links by the introduction of the latent variables *productivity* and *popularity*. We demonstrate the generality of the proposed framework by showing that both the symmetric models and the conditional models can be derived from the proposed framework as special cases, leading to the unification of various seemingly different forms for the existing models. We develop efficient EM-algorithms for computing the maximum likelihood solutions to the models proposed in this paper. Extensive empirical studies show the promising performances of the proposed models in several application domains. Further analysis is conducted to investigate the trade-offs of each stochastic block model when data characteristic varies.

The rest of the chapter is organized as follows. In Section 4.2, we give background information, including notation we will use and details of several previous approaches. In Section 4.3, we present the PPL model, several of its variations, and some of their properties. In Section 4.4, we provide a detailed analysis on the relationship between PPL models and several existing stochastic block models. In Section 4.5, we describe an efficient estimation algorithm. In Section 4.6, we show the results of experimental studies. Finally, we conclude in Section 4.7.

4.2 Background

In this section we first establish some necessary notations for ease of presentation. We then describe the details about two representative existing stochastic block models which are most relevant to our work.

4.2.1 Notation

For a directed network, we denote the nodes by $\mathcal{V} = \{1, \cdots, N\}$, the directed links by $\mathcal{E} = \{(i, j)|s_{ij} \neq 0\}$, where s_{ij} records the value associated with link from node i to node j. s_{ij} can either be binary, to denote whether there is a link from node i to node j, or be non-negative values, to denote the weight of the link. For simplicity, following [106], we assume the "link-in" space (i.e., all possible nodes that can point to a particular node) and "link-out" space(i.e., all possible nodes that can be pointed to by a particular node) of every node to be $\mathcal{V}$, i.e., the complete set of nodes. We use $\mathcal{I}(i) = \{j|s_{ji} \neq 0\}$ to denote the set of all nodes point to node i, and $\mathcal{O}(i) = \{j|s_{ij} \neq 0\}$ to denote the set of all nodes that are pointed to by node i. Let K denote the number of communities, $z_i \in \{1, \cdots, K\}$ denote the community variable of node i, and $\boldsymbol{\gamma}_i = (\gamma_{i1}, \cdots, \gamma_{iK})$ denote the community memberships of node i. In other words, γ_{ik} is the probability for the case $z_i = k$, i.e., node i belongs to community k.

4.2.2 Existing Models

We now review three variants of the well-known stochastic block model [45] that are closely related to the proposed model.

PHITS Model

PHITS [16] is a conditional model that focuses on the conditional link probability of $\Pr(j|i)$, i.e., given that node i produces a link, how likely this link will point to node j among all nodes. To compute $\Pr(j|i)$, a community variable z_i is first sampled from a multinomial distribution with parameter $\boldsymbol{\gamma}_i$ that describes the community membership of node i, then for a given z_i, the conditional link probability $\Pr(j|i, z_i)$ is given by $\Pr(j|i, z_i) = \beta_{jz_i}$, where the parameter β_{jk} represents the likelihood for node j to be pointed to by any node in community k. By integrating out z_i, we get the PHITS model

$$\Pr(j|i) = \sum_k \beta_{jk}\gamma_{ik} \tag{4.1}$$

It is well known that PHITS can be considered as an application of PLSA [40] to networked data.

Symmetric Joint Link (SJL) Model

SJL [16, 109, 79] is a symmetric model for community detection. It models the link structure by the joint probability $\Pr(i, j)$, i.e., the likelihood of creating a link between node i and j, as follows

$$\Pr(i, j) = \sum_k \Pr(j|k)\Pr(i|k)\Pr(k) = \sum_k \beta_{jk}\beta_{ik}\pi_k \tag{4.2}$$

In Equation (4.2), π_k is the prior probability for a link to be produced in community k, and β_{ik} and β_{jk} are the conditional probabilities that nodes i and j are selected as the two ends of the link. Given the symmetric treatment, i.e., $\Pr(i, j) = \Pr(j, i)$, it is evident that SJL may not be suitable for *directed* network community detection.

4.3 Popularity and Productivity Link (PPL) Model

In this section, we first present our *popularity and productivity link* (PPL) model in its general form, then give three variations of the general PPL model, and finally discuss several properties of the PPL models.

4.3.1 General Form of PPL

PPL models the joint link probability $\Pr(i, j)$, i.e., how likely there is a directed link from node i to node j. In order to emphasize the different roles played by i and j, we write $\Pr(i, j)$ as $\Pr(i_{\rightarrow},\, j_{\leftarrow})$, denoting that node i plays the role of producing the link, and node j plays the role of receiving the link. Following the idea of SJL, we model $\Pr(i_{\rightarrow},\, j_{\leftarrow})$ as follows

$$\Pr(i_{\rightarrow}, j_{\leftarrow}) = \sum_k \Pr(i_{\rightarrow}|k)\Pr(j_{\leftarrow}|k)\Pr(k) = \sum_k \left(\frac{\gamma_{ik} a_i}{\sum_{i'} \gamma_{i'k} a_{i'}} \frac{\gamma_{jk} b_j}{\sum_{i'} \gamma_{i'k} b_{i'}} \sum_{i'} \gamma_{i'k} c_{i'} \right) \tag{4.3}$$

where

- γ_{ik}: the probability for node i to belong to community k
- a_i: the *productivity* of node i, i.e., among all the nodes, how likely a link is produced by node i
- b_j: the *popularity* of node j, i.e., among all the nodes, how likely a link is received by node j
- c_i: the weight of node i in terms of deciding the community prior $\Pr(k)$ (which will be elaborated momentarily).

To handle scale invariance, we normalize so that $\sum_i a_i = \sum_j b_j = \sum_i c_i = 1$.

Generative Process

We explain Equation (4.3) by the following generative process of PPL:

- Sample a community z according to a prior distribution $\pi_1, \cdots, \pi_K$, where π_k is computed by $\pi_k = \sum_{i=1}^N \gamma_{ik} c_i$.
- Given community z, the conditional link probability is given by

$$\Pr(i_{\rightarrow}, j_{\leftarrow}|z) = \Pr(i_{\rightarrow}|z)\Pr(j_{\leftarrow}|z) = \frac{\gamma_{iz} a_i}{\sum_{i'} \gamma_{i'z} a_{i'}} \frac{\gamma_{jz} b_j}{\sum_{i'} \gamma_{i'z} b_{i'}} \tag{4.4}$$

There are two unique features in the above generative process:

- Prior probability $\pi_k = \sum_i \gamma_{ik} c_i$ is constructed as the weighted sum of node memberships γ_{ik}, where c_i is used to weight node i in the combination. This construction enforces the consistency between node memberships γ_{ik} and community prior $\{\pi_k\}_{k=1}^{K}$. This specific construction of community priors also simplify relation between the proposed framework and some existing models for community detection.
- In Equation (4.4), the two ends of link $i \to j$ are treated differently when modeling $\Pr(i_{\to}, j_{\leftarrow}|z)$: besides the dependence on community memberships γ_{ik} and γ_{jk}, $\Pr(i_{\to}|z)$ and $\Pr(j_{\leftarrow}|z)$ are modeled by a_i (i.e., the productivity of node i) and b_j (i.e., the popularity of node j), respectively, leading to the differentiation of the roles played by the two nodes.

With the joint link probability defined in Equation (4.3), the log-likelihood for links can be written as

$$\mathcal{L}(a, b, c, \gamma) = \sum_{(i,j)\in\mathcal{E}} s_{ij} \log \sum_k \frac{\gamma_{ik} a_i}{\sum_{i'} \gamma_{i'k} a_{i'}} \frac{\gamma_{jk} b_j}{\sum_{i'} \gamma_{i'k} b_{i'}} \sum_{i'} \gamma_{i'k} c_{i'} \tag{4.5}$$

Note that we use original data s_{ij} in the joint link model rather than normalized data $\hat{s}_{ij} = \dfrac{s_{ij}}{\sum_j s_{ij}}$ used in conditional link models [16, 106] . Parameters γ, a, b, and c can be inferred by maximizing the log-likelihood $\mathcal{L}(a, b, c, \gamma)$.

4.3.2 Variants of the General PPL Model

In this subsection, we show three variants of PPL model by introducing different restrictions on parameters a, b and c.

Popularity Link (PoL) Model

In the first restricted variation, we enforce $c_i = a_i, \forall i$ in Equation (4.3), leading to the following expression for $\Pr(i_{\to}, j_{\leftarrow})$

$$\Pr(i_{\to}, j_{\leftarrow}) = \sum_k \frac{\gamma_{jk} b_j}{\sum_{i'} \gamma_{i'k} b_{i'}} \gamma_{ik} a_i \tag{4.6}$$

We refer to this variant as the Popularity Link (PoL) Model. By assuming $c_i = a_i$, we essentially assume that the prior probability of each community (i.e., $\sum_i \gamma_{ik} c_i$) is identical to the prior probability for a link to be produced from that community (i.e., $\sum_i \gamma_{ik} a_i$).

Productivity Link (PrL) Model

In the second restricted variation, we enforce c_i to be equal to b_i, leading to the following expression for $\Pr(i_\rightarrow, j_\leftarrow)$,

$$\Pr(i_\rightarrow, j_\leftarrow) = \sum_k \frac{\gamma_{ik} a_i}{\sum_{i'} \gamma_{i'k} a_{i'}} \gamma_{jk} b_j \tag{4.7}$$

We refer to this variant as the Productivity Link (PrL) Model. By assuming $c_i = b_i$, we essentially assume that the prior probability of each community (i.e., $\sum_i \gamma_{ik} c_i$) is identical to the prior probability for a link to be received by that community (i.e., $\sum_i \gamma_{ik} b_i$).

Regularized PPL (PPL-D) Model

In this variation, instead of enforcing the relationship between c_i and a_i or b_i, we learn c_i from data, under certain regularization. In particular, we introduce a Dirichlet prior for parameters $c = (c_1, \ldots, c_N)$, i.e., $\Pr(c) \propto \prod_i c_i^\alpha$, where α is the hyper-parameter of Dirichlet distribution. Using the prior $\Pr(c)$ as the regularization, we obtain an MAP estimation of parameters by maximizing the following log-posterior probability

$$\mathcal{L}(a, b, c, \gamma) + \log \Pr(c) \tag{4.8}$$

where $\mathcal{L}(a, b, c, \gamma)$ is given in Equation (4.5). We call this PPL model regularized by the Dirichlet prior the PPL-D model.

4.3.3 Properties of PPL Model

In this subsection, we show two important properties of the PoL, PrL, and general PPL model.

Equivalence between the Variants of PPL Models

The first property is about the relationship between PoL model, PrL model, and general PPL model. Surprisingly, although their formulas are different, the optimal solutions for the three models actually result in identical joint link probability and therefore identical data likelihood. This property is described in the following theorem.

Theorem 1. *Under the optimal solution, the joint link probability* $\Pr(i_{\rightarrow}, j_{\leftarrow})$ *of PoL model, PrL model and general PPL model are the same. That is,* $\Pr^1(i_{\rightarrow}, j_{\leftarrow}|a^1, b^1, \gamma^1) = \Pr^2(i_{\rightarrow}, j_{\leftarrow}|a^2, b^2, \gamma^2) = \Pr^3(i_{\rightarrow}, j_{\leftarrow}|a^3, b^3, c^3, \gamma^3)$*, where* $\Pr^1(i_{\rightarrow}, j_{\leftarrow})$*,* $\Pr^2(i_{\rightarrow}, j_{\leftarrow})$*,* $\Pr^3(i_{\rightarrow}, j_{\leftarrow})$ *are the joint link probabilities of PoL model, PrL model, and general PPL model, respectively;* $\{a^1, b^1, \gamma^1\}$*,* $\{a^2, b^2, \gamma^2\}$ *and* $\{a^3, b^3, c^3, \gamma^3\}$ *are the optimal solutions to maximizing the log-likelihood of PoL model, PrL model and general PPL model, respectively. In particular, denoting the log-likelihood of PoL, PrL and general PPL model by* $\mathcal{L}_1(a, b, \gamma), \mathcal{L}_2(a, b, \gamma), \mathcal{L}_3(a, b, c, \gamma)$ *respectively, we have* $\mathcal{L}_1(a^1, b^1, \gamma^1) = \mathcal{L}_2(a^2, b^2, \gamma^2) = \mathcal{L}_3(a^3, b^3, c^3, \gamma^3)$*.*

In order to prove Theorem 1, we first state the following lemma about the optimal solution to the PPL model given in Equation (4.3).

Lemma 1. *Given that* $(a^3, b^3, c^3, \gamma^3)$ *is the optimal solution to maximizing the log-likelihood of PPL model, we define* $\pi_k = \sum_i \gamma^3_{ik} c^3_i$*. Then we can obtain one set of parameters* (a^1, b^1, γ^1) *such that* $\sum_i \gamma^1_{ik} a^1_i = \pi_k$ *and* (a^1, b^1, γ^1) *is the optimal solution to maximizing the log-likelihood of PoL model. Similarly, we can obtain another set of parameters* (a^2, b^2, γ^2) *such that* $\sum_j \gamma^2_{jk} b^2_j = \pi_k$ *and* (a^2, b^2, γ^2) *is the optimal solution to maximizing the log-likelihood of PrL model.*

Proof. we first show how to construct such (a^1, b^1, γ^1) and (a^2, b^2, γ^2). Given $(a^3, b^3, c^3, \gamma^3)$ and $\pi_k = \sum_i \gamma^3_{ik} c^3_i$, we can define $\hat{q}$ such that $\sum_i \gamma^3_{ik} a^3_i \hat{q}_k = \pi_k$. We then construct $\gamma^1_{ik} =$

$\frac{\gamma_{ik}^3 \hat{q}_k}{\sum_k \gamma_{ik}^3 \hat{q}_k}$, $a_i^1 = a_i^3 \sum_k \gamma_{ik}^3 \hat{q}_k$, $b_j^1 = \frac{b_j^3 \sum_k \gamma_{jk}^3 \hat{q}_k}{\sum_{j'} b_{j'}^3 \sum_k \gamma_{j'k}^3 \hat{q}_k}$, and we can show that

$$\sum_i \gamma_{ik}^1 a_i^1 = \pi_k$$

We can also define $\tilde{q}$ such that $\sum_j \gamma_{jk}^3 b_j^3 \tilde{q}_k = \pi_k$. We then construct $\gamma_{ik}^2 = \frac{\gamma_{ik}^3 \tilde{q}_k}{\sum_k \gamma_{ik}^3 \tilde{q}_k}$, $a_i^2 = \frac{a_i^3 \sum_k \gamma_{ik}^3 \tilde{q}_k}{\sum_{i'} a_{i'}^3 \sum_k \gamma_{i'k}^3 \tilde{q}_k}$ and $b_j^2 = b_j^3 \sum_k \gamma_{jk}^3 \tilde{q}_k$, and we can show that

$$\sum_j \gamma_{jk}^2 b_j^2 = \pi_k$$

With constructed (a^1, b^1, γ^1) and (a^2, b^2, γ^2) we can show that

$$\mathcal{L}_3(a^3, b^3, c^3, \gamma^3) = \mathcal{L}_1(a^1, b^1, \gamma^1) = \mathcal{L}_2(a^2, b^2, \gamma^2)$$

Next, we need to show that (a^1, b^1, γ^1) is the optimal solution to PoL model, (a^2, b^2, γ^2) is the optimal solution to PrL model. We prove this by contradiction. Assume their exists another set of parameters (a^*, b^*, γ^*) such that $\mathcal{L}_1(a^*, b^*, \gamma^*) > \mathcal{L}_1(a^1, b^1, \gamma^1) = \mathcal{L}_3(a^3, b^3, c^3, \gamma^3)$, then

$$\mathcal{L}_3(a^*, b^*, a^*, \gamma^*) = \mathcal{L}_1(a^*, b^*, \gamma^*) > \mathcal{L}_3(a^3, b^3, c^3, \gamma^3)$$

which contradicts that $(a^3, b^3, c^3, \gamma^3)$ is the optimal solution to PPL model. Similarly, we can show (a^2, b^2, γ^2) is the optimal solution to PrL model. Thus, we complete the proof. □

Following the above lemma, we can easily prove Theorem 1. One implication of this theorem is that the space of the optimal solution to the general PPL model is not a unique fixed point. As a consequence, if in addition to the joint link probability, we also care about the exact solution to the community membership γ, then we should not directly apply PPL in its general form. Instead, we should either choose PoL and PrL if the MLE solution is needed, or choose PPL-D if the MAP solution is needed.

Perfect Fitting of the Distributions of Indegree and Outdegree

The second property of the PPL model is about degree fitting. It turns out that the optimal solutions to PoL model, PrL model, and general PPL model all fit exactly the degree distributions (both indegree and outdegree) in the network data. This is described in the following theorem, whose proof is given in the appendix.

Theorem 2. *The model outdegree distribution* $\Pr(i_{\rightarrow})$ *and model indegree distribution* $\Pr(j_{\leftarrow})$ *of PoL model, PrL model and general PPL model fit exactly the actual indegree and outdegree distributions of the network data.*

Proof. We can easily show that the optimal solution to a_i in PoL model is equal to the normalized outdegree of node i, i.e., $a_i^1 = \frac{\sum_j s_{ij}}{\sum_{ij} s_{ij}}$; and the optimal solution to b_j in PrL model is equal to the normalized indegree of node j, i.e., $b_j^2 = \frac{\sum_i s_{ij}}{\sum_{ij} s_{ij}}$. From the model formulation in Equation (4.6) for PoL model, we have

$$\Pr^1(i_{\rightarrow}|a^1, b^1, \gamma^1) = \sum_j \Pr^1(i_{\rightarrow}, j_{\leftarrow}|a^1, b^1, \gamma^1) = a_i^1$$

So the model outdegree distribution of PoL model fits exactly the actual outdegree distribution of the network.

From the model formulation in Equation (4.7) for PrL model, we have

$$\Pr^2(j_{\leftarrow}|a^2, b^2, \gamma^2) = \sum_i \Pr(i_{\rightarrow}, j_{\leftarrow}|a^2, b^2, \gamma^2) = b_j^2$$

So the model indegree distribution of PrL model fits exactly the actual indegree distribution of the network. Following Theorem 1 we have

$$\Pr^3(i_{\rightarrow}|a^3, b^3, c^3, \gamma^3) = \Pr^2(i_{\rightarrow}|a^2, b^2, \gamma^2) = \Pr^1(i_{\rightarrow}|a^1, b^1\gamma^1) = a_i^1$$

and

$$\Pr^3(j_{\leftarrow}|a^3, b^3, c^3, \gamma^3) = \Pr^1(j_{\leftarrow}|a^1, b^1, \gamma^1) = \Pr^2(j_{\leftarrow}|a^2, b^2, \gamma^2) = b_j^2$$

We conclude that the model indegree and outdegree distributions estimated from PoL model, PrL model and PPL model fit exactly the actual indegree and outdegree distributions of the network. □

This property of degree fitting is a consequence of the concepts of *productivity* and *popularity*. We argue that degree fitting is a very important property for a generative model. This is because in real world, most networks have heavy-tailed (or power-law) degree distribution. So far, no existing stochastic block models can guarantee to generate degree distributions fitting both indegree and outdegree distributions of real-world networks.

4.4 Relationship with Existing Models

In this section, we describe the relationship between PPL and several existing models, including conditional link models, namely PCL [106] and PHITS [16], and symmetric joint link model (SJL) [79, 109]. It turns out that these existing models all can be considered as special cases of PPL, with different constraints. Such a connection demonstrates that PPL provides a consistent framework to unify the existing models.

4.4.1 Relationship with Conditional Link Models

We show that the Popularity Conditional Link(PCL) model in [106] is a *conditional* version of the Popularity Link(PoL) model described in Section 4.3.2. Starting from the joint probability given in Equation (4.6), we can express the conditional probability of the PoL model as

$$\Pr(j_{\leftarrow}|i_{\rightarrow}) = \frac{\Pr(i_{\rightarrow}, j_{\leftarrow})}{\Pr(i_{\rightarrow})} = \sum_k \frac{\gamma_{jk} b_j}{\sum_{i'} \gamma_{i'k} b_{i'}} \gamma_{ik} \tag{4.9}$$

Note that in the above derivation, we use the fact that $\Pr(i_{\rightarrow}) = a_i$, which is obtained in the proof of Theorem 2. Equation (4.9) is exactly the same as the Popularity Conditional Link(PCL) model proposed in [106]. Because of this connection, in the following discussion, we also refer to the PCL model described in Equation (4.9) (and in [106]) as PoCL model.

Following a similar idea, from Productivity Link(PrL) model we can derive a Productivity Conditional Link model by computing the conditional probability $\Pr(i_{\rightarrow}|j_{\leftarrow})$ from Equation (4.7) as the following

$$\Pr(i_{\rightarrow}|j_{\leftarrow}) = \frac{\Pr(i_{\rightarrow}, j_{\leftarrow})}{\Pr(j_{\leftarrow})} = \sum_k \frac{\gamma_{ik} a_i}{\sum_{i'} \gamma_{i'k} a_{i'}} \gamma_{jk} \tag{4.10}$$

In the above derivation, we use the fact that $\Pr(j_{\leftarrow}) = b_j$, which is obtained in the proof of Theorem 2. Because of its connection to PrL, we refer to this new conditional model as PrCL.

As we can see, PoCL and PrCL capture the conditional link probability in different directions. PoCL depends on the *popularity* of the receiving node j while PrCL depends on the *productivity* of the producing node i. In addition, both PoCL and PrCL can be naturally derived from the PPL models, i.e., PoL and PrL.

Because as we have discussed before, PHITS is a relaxed version of PoCL, obviously it can also be derived from PPL.

4.4.2 Relationship with Symmetric Joint Link Models

To show its relationship with SJL, we enforce that $c_i = a_i = b_i, \forall i$ in the general PPL model. From a probabilistic view point this restricts that for each node, the probability for producing links is equal to that for receiving links. With this restriction, Equation (4.3) is reduced to

$$\Pr(i_{\rightarrow}, j_{\leftarrow}) = \sum_k \frac{\gamma_{ik} c_i}{\sum_{i'} \gamma_{i'k} c_{i'}} \frac{\gamma_{jk} c_j}{\sum_{j'} \gamma_{j'k} c_{j'}} \sum_{i'} \gamma_{i'k} c_{i'}$$

The following theorem, whose proof is given in the appendix, shows that this restricted version of PPL is exactly the SJL model.

Theorem 3. *Under the constraint that $a_i = b_i = c_i, \forall i$, the general PPL model is equivalent to the SJL model.*

Proof. The joint link probability of SJL model is given in Equation (4.2), i.e.,

$$\Pr(i_{\rightarrow}, j_{\leftarrow}) = \sum_k \beta_{ik}\beta_{jk}\pi_k$$

The community membership of SJL model is defined as[109, 79]

$$\gamma^f_{ik} = \frac{\beta_{jk}\pi_k}{\sum_k \beta_{ik}\pi_k} \tag{4.11}$$

We can also define c^f_i as

$$c^f_i = \sum_k \beta_{ik}\pi_k \tag{4.12}$$

Similarly, given solution (γ, c) to PPL model with $a = b = c$, we can define

$$\pi^p_k = \sum_i \gamma_{ik}c_i, \quad \beta^p_{ik} = \frac{\gamma_{ik}c_i}{\sum_i \gamma_{i'k}c_{i'}} \tag{4.13}$$

All we need to show is that given that (β, π) is the solution to SJL model, (γ^f, c^f) defined as in Equations (4.11,4.12) is the solution to PPL model under the restriction of $a = b = c$; and given that (γ, c) is the optimal solution to PPL model under the restriction of $a = b = c$, (π^p, β^p) defined in Equation (4.13) is the optimal solution to SJL model. First note that

$$\mathcal{L}_0(\beta, \pi) = \mathcal{L}_3(\gamma^f, c^f)$$
$$\mathcal{L}_3(\gamma, c) = \mathcal{L}_0(\beta^p, \pi^p)$$

where $\mathcal{L}_0$ and $\mathcal{L}_3$ are the log-likelihood of SJL model and PPL model respectively. Given that (β, π) is the optimal solution to SJL model, if there exists (γ^*, c^*) such that $\mathcal{L}_3(\gamma^*, c^*) > \mathcal{L}_3(\gamma^f, c^f) = \mathcal{L}_0(\beta, \pi)$, then we can construct (β^*, π^*) as in Equation (4.13) such that $\mathcal{L}_0(\beta^*, \pi^*) = \mathcal{L}_3(\gamma^*, c^*) > \mathcal{L}_0(\beta, \pi)$, which contradicts the assumption that (β, π) is the optimal solution to SJL model. Similarly, given that (γ, c) is the optimal solution to PPL model, if there exists (π^*, β^*) such that $\mathcal{L}_0(\pi^*, \beta^*) > \mathcal{L}_0(\pi^p, \beta^p) = \mathcal{L}_3(\gamma, c)$, then we can construct (γ^*, c^*) as in Equations (4.11,4.12) such that $\mathcal{L}_3(\gamma^*, c^*) = \mathcal{L}_0(\pi^*, \beta^*) > \mathcal{L}_3(\gamma, c)$, which contradicts the assumption that (γ, c) is the optimal solution to PPL model. Therefore, we prove that PPL model under the restriction of $a = b = c$ is equivalent to SJL model. □

The relationship revealed by Theorem 3 shows that SJL is a special PPL with the constraint that nodes having the same probability in terms of producing and receiving links, which is appropriate only for modeling *undirected* networks.

4.4.3 Summary

In Table 4.1, we summarize all the models discussed in this paper. Models that are newly developed in this paper are print in bold. We believe such a unified picture, offered through the PPL model, will be very helpful for understanding and further studying different stochastic block models for community detection.

Table 4.1. Taxonomy of the models categorized by type and variables

	popularity	productivity	both
conditional	PHITS, PoCL	**PrCL**	
joint	**PoL**	**PrL**	**PPL**, **PPL-D**
symmetric			SJL

4.5 Estimation Algorithm

In this section, we present efficient EM algorithms for computing the MLE solutions to PoL and PrL and the MAP solution to PPL-D. Because the derivation of the algorithms is rather lengthy, here we only present the final form of the algorithms as well as offer several observations, and we provide the detailed derivation in the appendix.

Theorem 4. *The following EM algorithms converge to the MLE solutions to PoL and PrL, and the MAP solution to PPL-D.*

E-step:

$$q_{ijk} \propto \mathrm{Pr}^{t-1}(i, j, k)$$

where t-1 indicates the result in the previous iteration

M-step:

$$PoL: \quad \gamma_{ik} = \frac{n_{ik}}{m^{\tau}_{k} b_i + n_{out}(i)}, \quad b_i = \frac{n_{in}(i)}{\sum_k m^{\tau}_{k} \gamma_{ik}}, \quad a_i = \frac{n_{out}(i)}{\sum_i n_{out}(i)}$$

$$PrL: \quad \gamma_{ik} = \frac{n_{ik}}{m^{\eta}_{k} a_i + n_{in}(i)}, \quad a_i = \frac{n_{out}(i)}{\sum_k m^{\eta}_{k} \gamma_{ik}}, \quad b_i = \frac{n_{in}(i)}{\sum_i n_{in}(i)}$$

$$\begin{aligned} PPL\text{-}D: \quad \gamma_{ik} &= \frac{n_{ik} + m^{\zeta}_{ik}}{m^{\eta}_{k} a_i + m^{\tau}_{k} b_i + m^{\zeta}_{i}}, c_i = \frac{m^{\zeta}_{i} + e\alpha}{\sum_i (m^{\zeta}_{i} + e\alpha)} \\ a_i &= \frac{n_{out}(i)}{\sum_k m^{\eta}_{k} \gamma_{ik}}, \quad b_i = \frac{n_{in}(i)}{\sum_k m^{\tau}_{k} \gamma_{ik}} \end{aligned}$$

where e is the summation of all s_{ij} and the rest variables are defined as:

$$\eta_k = \sum_{i'} \gamma^{t-1}_{i'k} a^{t-1}_{i'}, \; \tau_k = \sum_{j'} \gamma^{t-1}_{j'k} b^{t-1}_{j'}, \; \zeta_{ik} = \frac{\gamma^{t-1}_{ik} c^{t-1}_{i}}{\sum_{i'} \gamma^{t-1}_{i'k} c^{t-1}_{i'}}$$

$$n_{in}(i,k) = \sum_{j \in \mathcal{I}(i)} s_{ji} q_{jik}, \quad n_{out}(i,k) = \sum_{j \in \mathcal{O}(i)} s_{ij} q_{ijk}$$

$$n_{in}(i) = \sum_k n_{in}(i,k), \quad n_{out}(i) = \sum_k n_{out}(i,k)$$

$$n_{ik} = n_{in}(i,k) + n_{out}(i,k), \quad m_k = \sum_{(i \to j) \in \mathcal{E}} s_{ij} q_{ijk}$$

$$m^{\tau}_{k} = \frac{\sum_{(i \to j) \in \mathcal{E}} s_{ij} q_{ijk}}{\tau_k}, \quad m^{\eta}_{k} = \frac{\sum_{(i \to j) \in \mathcal{E}} s_{ij} q_{ijk}}{\eta_k}$$

$$m^{\zeta}_{ik} = \zeta_{ik} \sum_{(i \to j) \in \mathcal{E}} s_{ij} q_{ijk}, \quad m^{\zeta}_{i} = \sum_k m^{\zeta}_{ik}.$$

Proof. In the E-step, we would bound the log-likelihood from below. The key point is to apply the Jensen inequality $\log \sum_k p_k \geq \sum_k q_k \log p_k / q_k$, where $\sum_k q_k = 1$, to the log-sum-term in the log-likelihood and to apply the inequality $-\log x \geq 1 - \dfrac{x}{y} - \log y$ to the summation term in the denominator of the log-sum-term in the log-likelihood. In particular, at the t-th iteration the log-sum-term is lower bounded as

$$\log \sum_k \Pr(i,j,k) \geq \sum_k q_{ijk} \log \Pr(i,j,k)/q_{ijk}$$

with q_{ijk} computed as $q_{ijk} \propto \Pr^{t-1}(i,j,k)$ where superscript $t-1$ means the probability is computed under the values of the parameters in the $(t-1)$-th iteration. Then the denominator term in $\Pr(i,j,k)$ would be lower bounded as

$$-\log \sum_{i'} \gamma_{i'k} a_{i'} \geq 1 - \frac{\sum_{i'} \gamma_{i'k} a_{i'}}{\eta_k} - \log \eta_k$$
$$-\log \sum_{i'} \gamma_{i'k} b_{i'} \geq 1 - \frac{\sum_{i'} \gamma_{i'k} b_{i'}}{\tau_k} - \log \tau_k$$

with η_k, τ_k computed as

$$\eta_k = \sum_{i'} \gamma_{i'k}^{t-1} a_{i'}^{t-1} \quad \tau_k = \sum_{j'} \gamma_{j'k}^{t-1} b_{j'}^{t-1}$$

and the summation term $\sum_{i'} \gamma_{i'k} c_{i'}$ in PPL model is lower bounded as

$$\log \sum_{i'} \gamma_{i'k} c_{i'} \geq \sum_{i'} \zeta_{i'k} \log \gamma_{i'k} c_{i'} / \zeta_{i'k}$$

with ζ computed as $\zeta_{ik} = \dfrac{\gamma_{ik}^{t-1} c_i^{t-1}}{\sum_{i'} \gamma_{i'k}^{t-1} c_{i'}^{t-1}}$. Due to the limit of space, we omit the details about deriving the lower bound of the three log-likelihoods.

In the M-step, we will maximize the corresponding lower bound over the corresponding parameters as follows:

$$\text{PoL} : \sum_{(i,j)\in\mathcal{E}} s_{ij} \sum_k q_{ijk} \left(\log \gamma_{ik} \gamma_{jk} b_j - \sum_{j'} \frac{\gamma_{j'k} b_{j'}}{\tau_k} \right)$$

$$\text{PrL} : \sum_{(i,j)\in\mathcal{E}} s_{ij} \sum_k q_{ijk} \left(\log \gamma_{jk} \gamma_{ik} a_i - \sum_{i'} \frac{\gamma_{i'k} a_{i'}}{\eta_k} \right)$$

$$\text{PPL-D} : \sum_{(i,j)\in\mathcal{E}} s_{ij} \sum_k q_{ijk} \left(\log \gamma_{ik} a_i \gamma_{jk} b_j - \sum_{i'} \frac{\gamma_{i'k} a_{i'}}{\eta_k} \right.$$
$$\left. - \sum_{i'} \frac{\gamma_{i'k} b_{i'}}{\tau_k} + \sum_{i'} (\zeta_{i'k} + \alpha) \log c_i + \sum_{i'} \zeta_{i'k} \log \gamma_{i'k} \right)$$

By taking the derivatives of the expressions and setting them to zero, we can obtain the corresponding formulas in the M-step. □

It can be observed from the EM algorithm that in every iteration (and therefore in the final solutions) for each node i, its productivity a_i is proportional to its outdegree and its popularity b_i is proportional to its indegree. This is consistent with our intentions that the productivity of a node reflects how likely it produces a link and the popularity of a node reflects how likely it receives a link.

In addition, it is worth mentioning that in the real implementation, we avoid to explicitly compute all q_{ijk}'s (whose number is N^2K, which can be extremely large). Instead, q_{ijk}'s are computed in an "on-demand" fashion. We can show that the complexity (per iteration) of our EM algorithms is linear in the number of links in the network. Therefore, the algorithm is very efficient because in most real applications, networks are sparse and so the number of links is usually manageable.

4.6 Experiments

In this section, we show experiment results. We evaluate a variety of models (variations of PPL and existing models) on two tasks: community detection and link prediction. In addition, we also investigate the issue of degree fitting. We start by describing the data sets used in the experiments.

4.6.1 Data Sets

In the following experiments, we use a blog network and two paper citation networks.

Political Blog Network

This is a directed network of hyperlinks between a set of weblogs about US politics, recorded by Adamic and Glance [1]. In this network, there are totally 1,490 nodes and 19,090 links. Each node is labeled as either conservative or liberal.

Paper Citation Networks

We use the Cora paper citation network and the Citeseer paper citation network processed by Getoor et al.[1]. There are totally 2,708 nodes and 5,429 links in Cora network, and 3,312 nodes and 4,732 links in Citeseer network. Each paper in Cora network is categorized into one of 7 classes (e.g., Genetic Algorithms, Neural Networks, etc.), and each paper in Citeseer network is labeled as one of 6 classes.

4.6.2 Community Detection

In the first task, communities are to be detected from the networks. In this task, the real class labels in the data sets are used as the ground truth to evaluate the communities detected by different models. More specifically, we use the following evaluation metrics.

Evaluation Metrics for Community Detection

We use three commonly used metrics for evaluating the performance of community detection, i.e. normalized mutual information (NMI), pairwise F measure (PWF), and modularity (Modu). We first give detailed description about the three metrics.

Normalized mutual information (NMI) is defined as follows: given the true community structure $C = \{C_1, \cdots, C_K\}$, where C_k denote the set of nodes in the k-th community, and the community structure $C' = \{C'_1, \cdots, C'_K\}$ obtained from a model, the mutual information is computed as

$$MI(C, C') = \sum_{C_k, C'_l} p(C_k, C'_l) \log \frac{p(C_k, C'_l)}{p(C_k)p(C'_l)}$$

where $p(C_k)$ denotes the probability that a randomly selected node belongs to C_k, and $p(C_k, C'_l)$ denotes the joint probability that a randomly selected node belongs to C_k and C'_l. The normalized mutual information is defined as

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

[1] http://www.cs.umd.edu/projects/linqs/projects/lbc/

where $H(C) = \sum_k p(C_k) \log \frac{1}{p(C_k)}$ is the entropy of partition C.

Pairwise F measure (PWF) is another commonly used measure for evaluating clustering algorithms. Assume T is the set of node pairs (i, j) where nodes i and j belong to the same community in the ground truth, and S is the set of node pairs that belong to the same community in the outcome of a specific model. Then the pairwise F measure is computed from pairwise precision and recall as

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

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

where $|\cdot|$ indicates the cardinality of a set.

Note that to compute the normalized mutual information and pairwise F measure, the ground truth must be used. However, in some cases, the ground truth does not necessarily faithfully reflect the link structure. Therefore, we also use another measure called directed modularity (Modu), which is proposed by Leicht et al. [54] for measuring community partitions in directed networks without using ground truth. The definition of the directed modularity is given by

$$Modu = \frac{1}{e} \sum_{ij} \left(s_{ij} - \frac{d_{out}(i) d_{in}(j)}{e} \right) \delta(c_i, c_j)$$

where $d_{in}(i)$ and $d_{out}(i)$ are the indegree and outdegree of node i in the network, e is the number of directed links in the network, and c_i denotes the community of node i assigned by a model, and $\delta(\cdot, \cdot)$ is the Kronecker delta function.

For all the three metrics, i.e., NMI, PWF, and Modu, larger values correspond to better performances.

Performance on Community Detection

The community detection performances for different models on the three data sets are given in Tables 4.2, 4.3, and 4.4. Among the models, PHITS, PoCL, and PrCL are conditional link models. PHITS [17] represents the model described in Equation (5.4); PoCL represents the Popularity Conditional Link model [106] described in Equation (**??**); PrCL represents the Productivity Conditional Link model described in Equation (4.10). SJL represents symmetric link model as described in [79]. PoL, PrL, and PPL-D are the joint link models proposed in this work.

All the EM algorithms for MLE and MAP are run with 100 iterations, which according to our observation is more than enough for convergence. In order to alleviate the problem of local minimum of EM algorithms, for each test we conduct 10 trials with different random initializations, and choose the one giving the largest likelihood. The prior α for the parameter c in PPL-D is set to 1. Actually, we found the performance not sensitive to α—we tested different values for α ranging from 0.01 to 1, and the results are almost the same.

From the performance results, we can make the following comparisons and observations.

Joint link models vs. conditional link models Joint link models clearly outperform conditional link models. These can be seen from that our joint link models, i.e., PoL, PrL, and PPL-D always have the top performances and clearly outperform their conditional counterparts PoCL and PrCL. Even the symmetric joint link model SJL outperforms its conditional counterpart PHITS in most of the cases. This result verifies our assumption that modeling both behavior in receiving links (popularity) and that in producing links (productivity) is better than modeling just one behavior or none at all.

Non-symmetric vs. symmetric joint link models Comparing the performances of non-symmetric link models, i.e., PoL, PrL, and PPL-D, with that of traditional SJL model, which is symmetric, we can see that the non-symmetric models consistently outperform SJL and the improvement is quite significant in many cases. This verifies the benefit of separating the behavior of nodes in receiving links and that in producing links over simply ignoring the direction of links.

PPL models without vs. with restrictions Comparing PPL-D, which does not restrict c other than providing a weak prior, with PoL, PrL and SJL, which enforce $c = a$, $c = b$, and $c = a = b$, respectively, we can see that PPL-D has the best performance. However, as shown in Section 4.3.3 we can always derived PoL and PrL from PPL-D that give the identical data likelihood, and so the above result suggests that PPL-D tends to find better solutions for community memberships.

Popularity vs. productivity If we can only choose one feature between popularity and productivity for community detection in our data sets, it seems that popularity has a small edge over productivity. This can be observed both in joint link models (i.e., PoL over PrL) and conditional models (i.e., PoCL over PrCL). Such a result suggests that to determine the community membership of a node i in these three data sets, those nodes point to i may be more important than those nodes pointed to by i.

Table 4.2. Community detection performance on the Political Blog data set.

Algo.	NMI	PWF	Modu
PHITS	0.3829	0.7152	0.4200
PoCL	0.4905	0.7947	0.4270
PrCL	0.4569	0.7776	0.4243
SJL	0.4409	0.7425	0.4323
PoL	0.5156	0.8072	**0.4324**
PrL	0.5178	0.8091	**0.4324**
PPL-D	**0.5365**	**0.8167**	**0.4324**

Table 4.3. Community detection performance on the Cora data set.

Algo.	NMI	PWF	Modu
PHITS	0.0591	0.1862	0.3594
PoCL	0.0797	0.1982	0.5982
PrCL	0.0211	0.1666	0.4959
SJL	0.0602	0.1840	0.6091
PoL	0.0886	0.2014	0.6310
PrL	0.0870	0.1993	0.6307
PPL-D	**0.0972**	**0.2085**	**0.6381**

4.6.3 Link Prediction

In this task, we study the performance of the *joint link* models on predicting the links (both incoming links and outgoing links). Specifically, for each node in the network we randomly hide one of its incoming links and one of its outgoing links and ask each model to recover the

Table 4.4. Community detection performance on the Citeseer data set.

Algo.	NMI	PWF	Modu
PHITS	0.0117	0.1788	0.4374
PoCL	0.0292	0.1909	0.6214
PrCL	0.0131	0.1805	0.5954
SJL	0.0236	0.1896	0.6348
PoL	0.0292	0.1921	0.6648
PrL	0.0263	0.1904	0.6612
PPL-D	**0.0317**	**0.1948**	**0.6687**

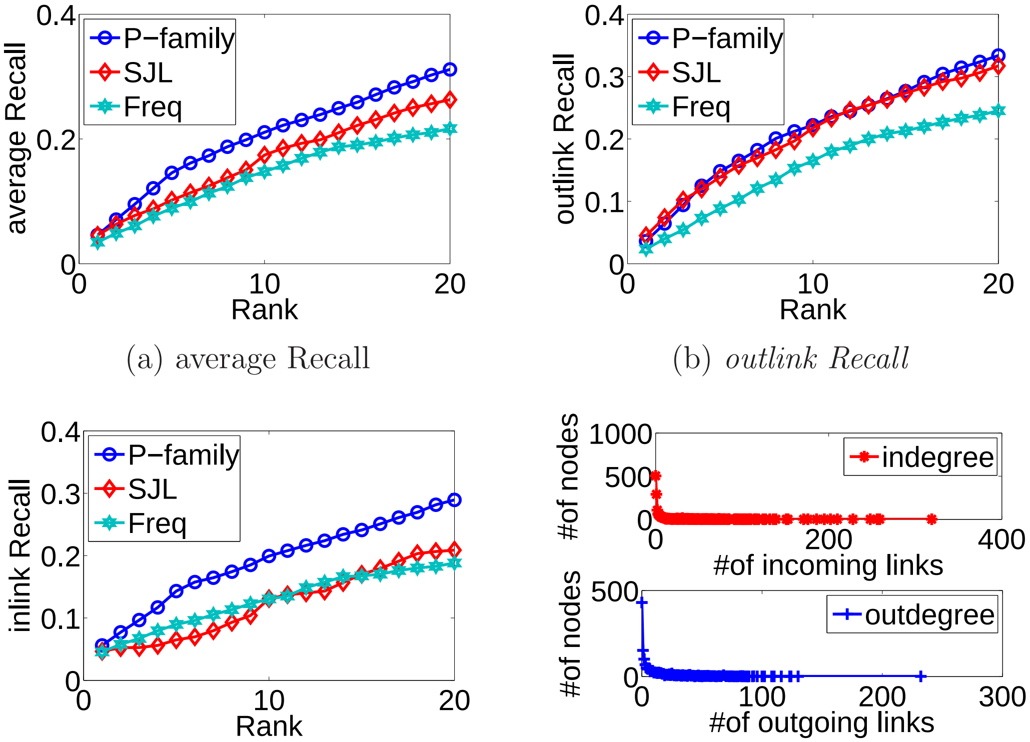
missing links. Such a task has practical values in applications such as friend recommendation in social networks and citation suggestion in citation networks.

Evaluation Metric for Link Prediction

We measure the performance of link prediction by Recall measure. Two types of recall are presented, namely *outlink recall* and *inlink recall*. The outlink recall measures the ability of a model to predict nodes pointed to by a given node. The inlink recall measures the ability of a model to predict the nodes point to a given node. To compute outlink recall for node i, we first compute the outlink probabilities $\Pr(j_{\leftarrow}|i_{\rightarrow})$ for node i to all other nodes by $\Pr(j_{\leftarrow}|i_{\rightarrow}) = \dfrac{\Pr(i_{\rightarrow}, j_{\leftarrow})}{\sum_j \Pr(i_{\rightarrow}, j_{\leftarrow})}$. The resulting probabilities assign an outlink rank to each node j. The outlink recall at rank position K is defined as the fraction of nodes whose top-K ranked predictions contain the true missing link. Inlink recall is defined similarly based on $\Pr(j_{\rightarrow}|i_{\leftarrow})$. In addition, we also report the average of the inlink and outlink recalls.

Performance on Link Prediction

The recalls at top-1 through top-20 on the three data sets are given in Figures 4.1, 4.2, and 4.3. All the results are averaged over 10 trials with different randomly selected missing links. Because we have that PoL, PrL and general PPL model have equal link probabilities, and because we also found that PPL-D achieves almost the same performance as PoL and PrL, we will only report one result for these models which are denoted by P-family models. We also report the results of a naive baseline, the Frequency-based model, where the outgoing



(a) average Recall

(b) *outlink Recall*

(c) *inlink* Recall

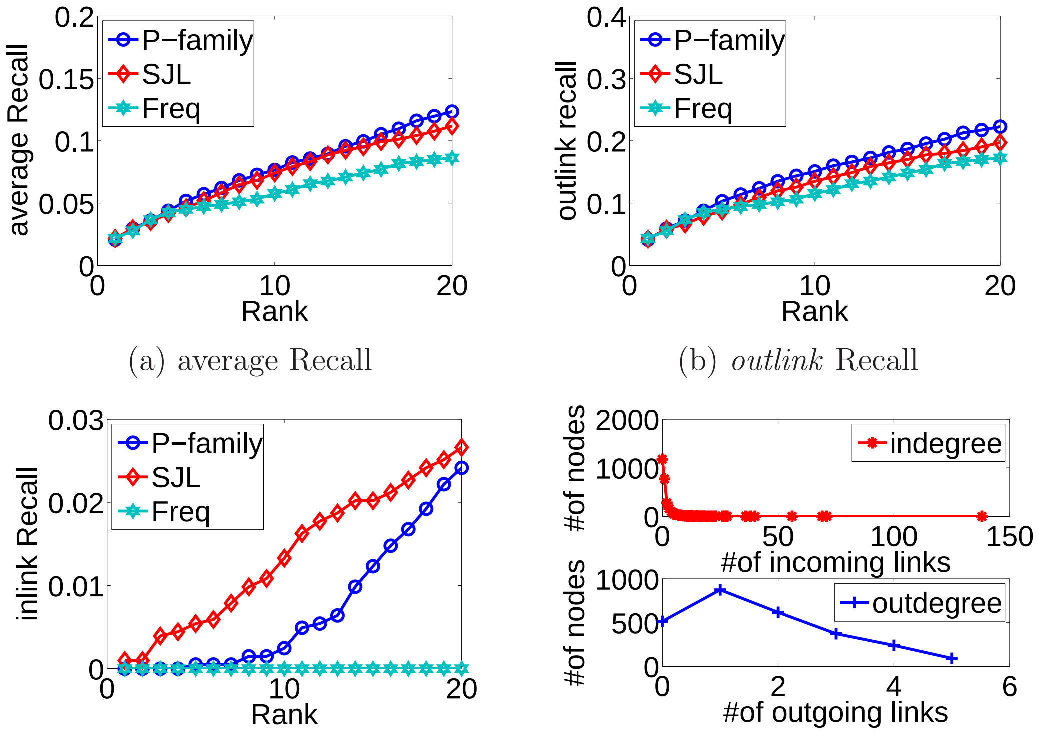
(d) degree distribution

Figure 4.1. (a)∼(c): Recalls on Political Blog data. (d): Degree distribution.

link probabilities are proportional to the indegree of nodes, i.e., $\Pr(j_{\leftarrow}|i_{\rightarrow}) \propto d_{in}(j)$, and the incoming link probabilities are proportional to the outdegree of nodes, i.e., $\Pr(j_{\rightarrow}|i_{\leftarrow}) \propto d_{out}(j)$ 20.

As can be seen from the figures, compared to SJL and the Frequency-based baseline, P-family models perform the best in all the cases except the *inlink* recall for Cora network. This result illustrates that most of the time, it is beneficial to use productivity and popularity to model indegree and outdegree distributions separately in a directed network.

However, the inlink recall for Cora network is an abnormal case, where SJL performs the best, P-family models perform worse, and the Frequency-based model has extremely poor performance (almost constantly zero). To see why this case is special, we show the degree distributions of the three networks in the rightmost panels of Figures 4.1, 4.2, and 4.3. All the degree distributions follow a power-law distribution except the outdegree distribution in



(a) average Recall

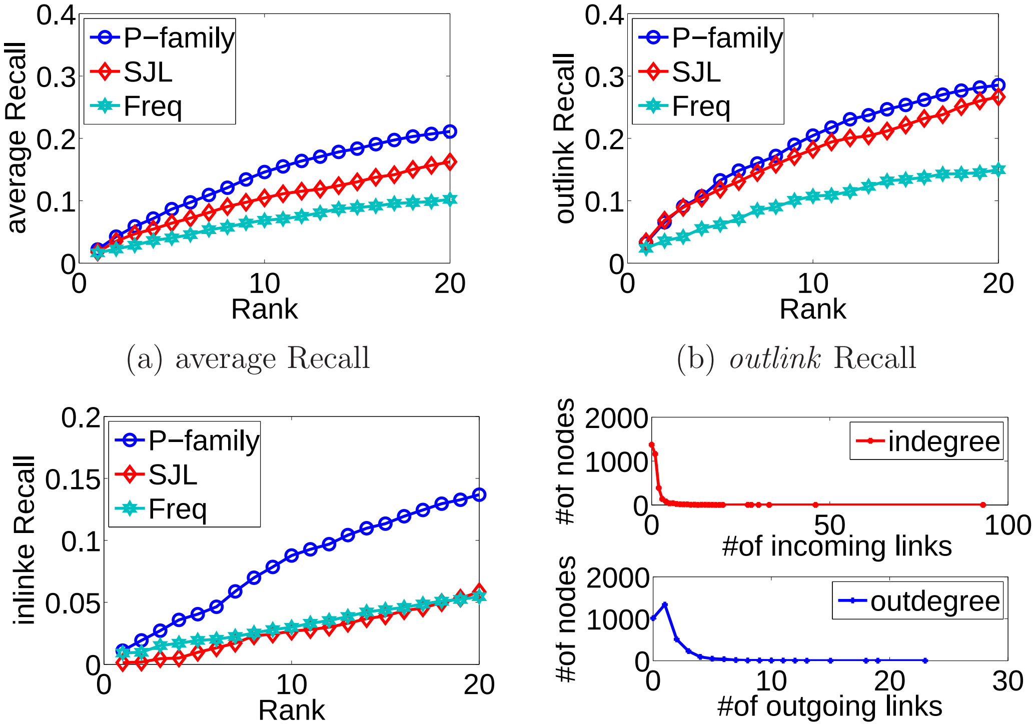
(b) *outlink* Recall

(c) *inlink* Recall

(d) degree distribution

Figure 4.2. (a)∼(c): Recalls on Cora data. (d): Degree distribution.

Cora network. The outdegree in Cora follows a rather uniform distribution with outdegree no lager than 5. (We suspect such a distribution is due to the small scale of the Cora data which leads to many references, and therefore outlinks, to be outside the data set.) Because of such a uniform distribution, the outdegrees of nodes are not informative, which explains the extremely poor performance of the Frequency-based model. The P-family models treat indegree and outdegree equally importantly and therefore also suffer from the uninformative outdegree distribution. SJL, in comparison, ignores the link direction and as a result makes the more informative indegree distribution dominate the uninformative outdegree distribution and therefore suffers the least. This special case actually reveals some trade-offs made by different models.



(a) average Recall

(b) *outlink* Recall

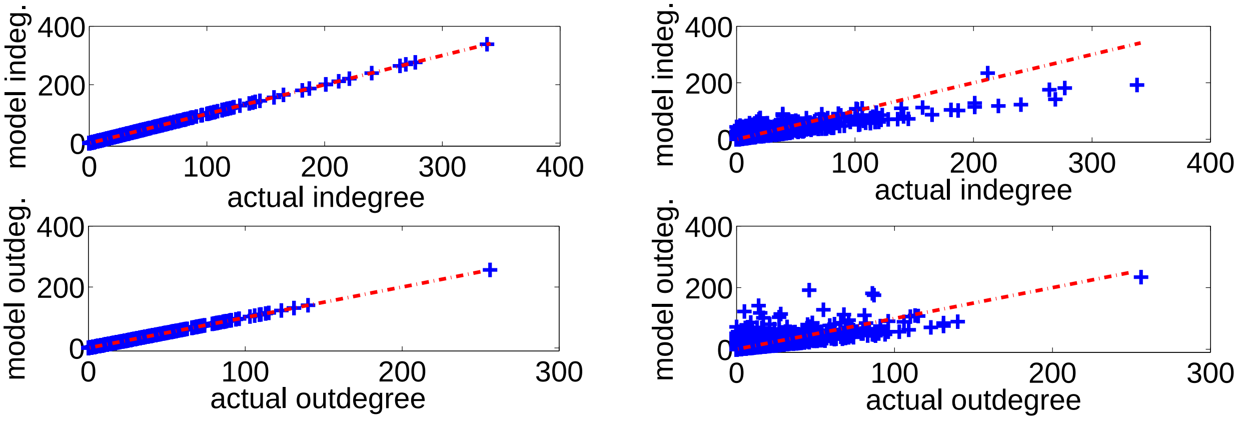
(c) *inlink* Recall

(d) degree distribution

Figure 4.3. (a)∼(c): Recalls on Citeseer data. (d): Degree distribution.

4.6.4 Degree Fitting

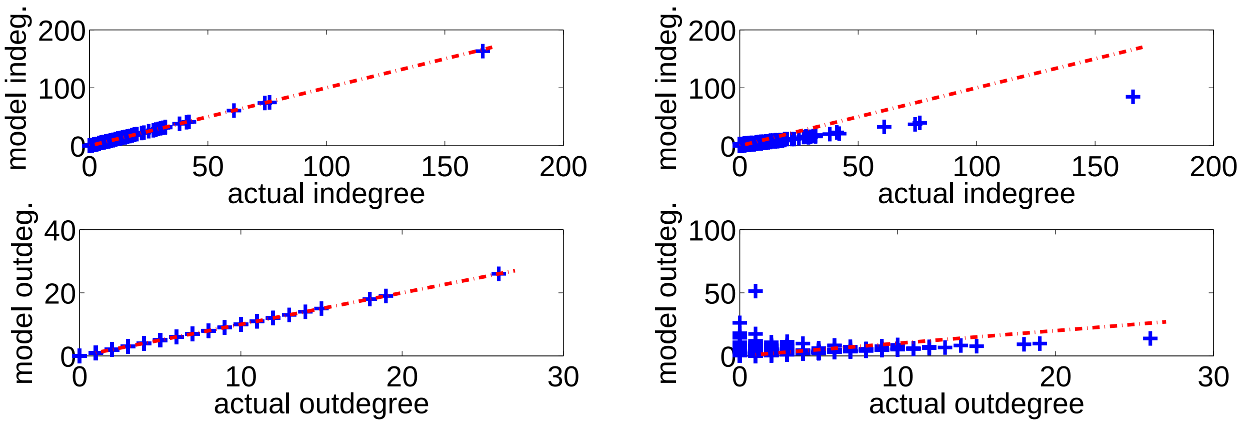
Finally, we verify the degree fitting properties of PPL models. Figure 4.4(a) shows the scatter plots for the indegree and outdegree fitting of PPL models on the Political Blog data set. Note that PoL, PrL and PPL-D again give almost the same result is this experiment and so we refer to them together as PPL. Each point in the plot represents a node, where its position on the horizontal axis is determined by its actual degree (indegree or outdegree) and its position on the vertical axis is determined by the degree predicted by the model. Therefore, a point fell on the diagonal line (the red lines in the plots) indicates a perfect degree match. As can be seen from the figure, all the points fall on the red line, which indicates that PPL captures the degree distributions for each node exactly. In comparison, as shown in Figure 4.4(b), SJL has very poor performance in terms of degree fitting. Similar results are obtained for the paper citation data sets, where in Figures 4.5(a) and 4.5(b) we show some of the results. These empirical studies clearly validate the degree fitting property of the PPL models that we previously stated in Section 4.3.3.



(a) degree fitting by PPL

(b) degree fitting by SJL

Figure 4.4. Degree fitting on the Political Blog data



(a) degree fitting by PPL

(b) degree fitting by SJL

Figure 4.5. Indegree fitting on Cora and outdegree fitting on Citeseer

4.7 Conclusions

Stochastic block model is a promising probabilistic model for community detection. In this paper, we present a new stochastic block model, PPL, for community detection in *directed* networks. On one hand, our model is *complete*, in that it captures the roles of each node both as a link producer and as a link receiver whereas a consistent community membership serves both the roles; on the other hand, our model is *unified*, in that it offers a unified framework to connect and to understand existing models. We believe such a complete and unified model provides a solid foundation for further studies in stochastic block models for community detection.

CHAPTER 5

Community Detection for Networked Data: A Discriminative Approach for Combining Link and Content

5.1 Introduction

In addition to the link information, nodes in networks 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.

In this chapter, 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 chapter 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 chapter is organized as follows. In Section 5.2 we present and analyze the conditional link model. In Section 5.3, we extend the link model to include the content information. Also in Section 5.3, we describe the two-stage optimization algorithm. In Section 5.4, we show extensions by combining our link model and content model with other existing content and link models. In Section 5.5, we show extensive experimental results on benchmark data sets. Finally, we give conclusions in Section 5.6.

5.2 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 5.3, we incorporate the content information into the proposed link model by a discriminative model.

5.2.1 Popularity 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, 63] 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

$$\Pr(j|i; z_i, b) = \frac{\gamma_{jz_i} b_j}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'z_i} b_{j'}} \tag{5.1}$$

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

$$\Pr(j|i; \gamma, b) = \sum_k \gamma_{ik} \frac{\gamma_{jk} b_j}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k} b_{j'}} \tag{5.2}$$

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

In Eq. (5.2), 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

$$\Pr(j|i; b) = \sum_k \gamma_{ik} \frac{\gamma_{jk} b_{jk}}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k} b_{j'k}} \tag{5.3}$$

Comparing Eq. (5.3) to (5.2), we see that Eq. (5.3) 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. (5.2) achieves better performance because of the reduced number of variables.

5.2.2 Analysis of the PCL Model

In this section, we analyze our link model by establishing the relation and comparing to PHITS model [16]. 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. (7.6), 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

$$\Pr(j|i; \gamma, \beta) = \sum_k \gamma_{ik}\beta_{jk} \tag{5.4}$$

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. (5.4) is equivalent to the link model with* $\Pr(j|i)$ *specified in Eq. (5.3).*

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. (5.2) from the PHITS model in Eq. (5.4) 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. (5.2) is equivalent to the PHITS model specified in Eq. (5.4) 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.

5.2.3 Maximum Likelihood Estimation

In this section, we present the method of maximum likelihood for the PCL model specified in Eq. (5.2). Observing the directional links $\mathcal{E} = \{(i \to j)|s_{ij} \neq 0\}$, we write the log-likelihood as

$$\log \mathcal{L} = \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij} \log \sum_k \gamma_{ik} \frac{\gamma_{jk}b_j}{\sum_{j'\in\mathcal{LO}(i)}\gamma_{j'k}b_{j'}} \tag{5.5}$$

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 5. *The log-likelihood* $\log \mathcal{L}$ *in Eq. (5.5) at the* t^{th} *iteration is lower bounded as follows*

$$\begin{aligned} \log \mathcal{L} \geq & \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk} \left(\log\gamma_{ik} + \log\gamma_{jk} + \log\frac{b_j}{\tau_{ik}} + 1 - \sum_{j'\in\mathcal{LO}(i)} \frac{\gamma_{j'k}b_{j'}}{\tau_{ik}} \right) \\ & - \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk}\log q_{ijk} \end{aligned}$$

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

$$\tau_{ik} = \sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k}^{t-1} b_{j'}^{t-1} \tag{5.6}$$

$$q_{ijk} \propto \gamma_{ik}^{t-1} \frac{\gamma_{jk}^{t-1} b_j^{t-1}}{\sum_{j' \in \mathcal{LO}(i)} \gamma_{j'k}^{t-1} b_{j'}^{t-1}} \quad s.t. \sum_k q_{ijk} = 1 \tag{5.7}$$

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.,

$$\max_{\gamma, b} \sum_{(i \to j) \in \mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk} \left(\log \gamma_{ik} \gamma_{jk} b_j - \sum_{j' \in \mathcal{LO}(i)} \frac{\gamma_{j'k} b_{j'}}{\tau_{ik}} \right) \tag{5.8}$$

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

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

$$n_{in}(i,k) = \sum_{j \in \mathcal{I}(i)} \hat{s}_{ji} q_{jik} \qquad n_{out}(i,k) = \sum_{j \in \mathcal{O}(i)} \hat{s}_{ij} q_{ijk}$$

$$n_{in}(i) = \sum_k n_{in}(i,k) \qquad n_{out}(i) = \sum_k n_{out}(i,k)$$

$$n(i,k) = n_{in}(i,k) + n_{out}(i,k) \qquad m(i,k) = \sum_{j \in \mathcal{LI}(i)} \frac{n_{out}(j,k)}{\tau_{jk}}$$

Theorem 6. *The optimal solution to Eq. (5.8) satisfies the following conditions*

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

$$\gamma_{ik} = \frac{n(i,k)}{m(i,k) b_i + n_{out}(i)}, \quad b_i = \frac{n_{in}(i)}{\sum_k m(i,k) \gamma_{ik}} \tag{5.9}$$

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

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

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

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

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

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

Remark: As revealed in Eq. (5.9), 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.

5.3 Discriminative Content Model

In this section, we extend our link model to incorporate the content information of nodes. As we discussed in Sections 5.1, 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

$$\Pr(z_i = k) = y_{ik} = \frac{\exp(w_k^T x_i)}{\sum_l \exp(w_l^T x_i)} \tag{5.10}$$

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. (5.2) 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. (5.10). 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 $\phi(x) : \mathbb{R}^d \to \mathbb{R}^m$ 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

$$\log \mathcal{L} = \sum_{(i \to j) \in \mathcal{E}} \hat{s}_{ij} \log \sum_k y_{ik} \frac{y_{jk} b_j}{\sum_{j' \in \mathcal{LO}(i)} y_{j'k} b_{j'}} \tag{5.11}$$

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. (5.11). 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:

$$\max_{w,b} \sum_{(i \to j) \in \mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk} \left(\log y_{ik} y_{jk} b_j - \sum_{j' \in \mathcal{LO}(i)} \frac{y_{j'k} b_{j'}}{\tau_{ik}} \right) \tag{5.12}$$

subject to non-negative constraints on b.

Instead of maximizing over w, we convert Eq. (7.1) into a constraint optimization problem over y and b by

$$\max_{y\in\Delta,b} \sum_{(i\to j)\in\mathcal{E}} \hat{s}_{ij} \sum_k q_{ijk} \left(\log y_{ik} y_{jk} b_j - \sum_{j'\in\mathcal{LO}(i)} \frac{y_{j'k} b_{j'}}{\tau_{ik}} \right) \tag{5.13}$$

where the domain Δ is defined as

$$\Delta = \left\{ y | \exists w, y_{ik} = \frac{\exp(w_k^T \phi(x_i))}{\sum_l \exp(w_l^T \phi(x_i))} \right\} \tag{5.14}$$

By having the domain of y given in Eq. (5.14) as a convex set, we can take a projection method to maximize the problem of Eq. (5.13), 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. (5.13) with only sum-to-one constraint on y_{ik} and non-negative constraints on b using the result in Theorem 6. 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

$$\max_w \sum_i \sum_k \tilde{y}_{ik} \log \frac{\exp(w_k^T \phi(x_i))}{\sum_l \exp(w_l^T \phi(x_i))} - \frac{\lambda}{2} \sum_k w_k^T w_k \tag{5.15}$$

where λ is the regularization coefficient. This problem is a convex problem [9] and has a unique optimal solution, and can be maximized efficiently by the Newton-Raphson method.

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

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. (5.6) and (5.7) using $y_{ik}$ rather than $\gamma_{ik}$
4. in the M-step,
    • compute $\gamma_{ik}$, and $b_i$ as in Theorem 6
    • update $w_k$ by maximizing the objective in Eq. (5.15) 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
```

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. (5.15) by the Newton-Raphson method.

5.4 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.

5.4.1 PCL + PLSA

Similar to [17] 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

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

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

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

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

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

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.

5.4.2 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

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

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

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

5.5 Experiments

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.

5.5.1 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. [35]. 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 [35].

Cora Data Set is a subset of the larger Cora citation data set [64]. This data set includes publications from the machine learning area, each of which is classified into 7 sub-categories 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 (`http://citeseer.ist.psu.edu/`). The Citeseer data set consists of 3312 scientific publications labeled as one of 6 classes and 4732 links. Each 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 $K = 6$.

5.5.2 Evaluation 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. [72] 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 [72], 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 ([83], 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 $vol(C_i) = \sum_{p \in C_i} \sum_q w_{pq}$.

5.5.3 Link Prediction

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 5.5.1. The performance is reported in Figure 5.1 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.

Table 5.1. Community detection performance on Cora and Citeseer dataset

		Cora				Citeseer			
	Algorithm	NMI	PWF	Modu	NCut	NMI	PWF	Modu	NCut
L	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**
C	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
LC	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

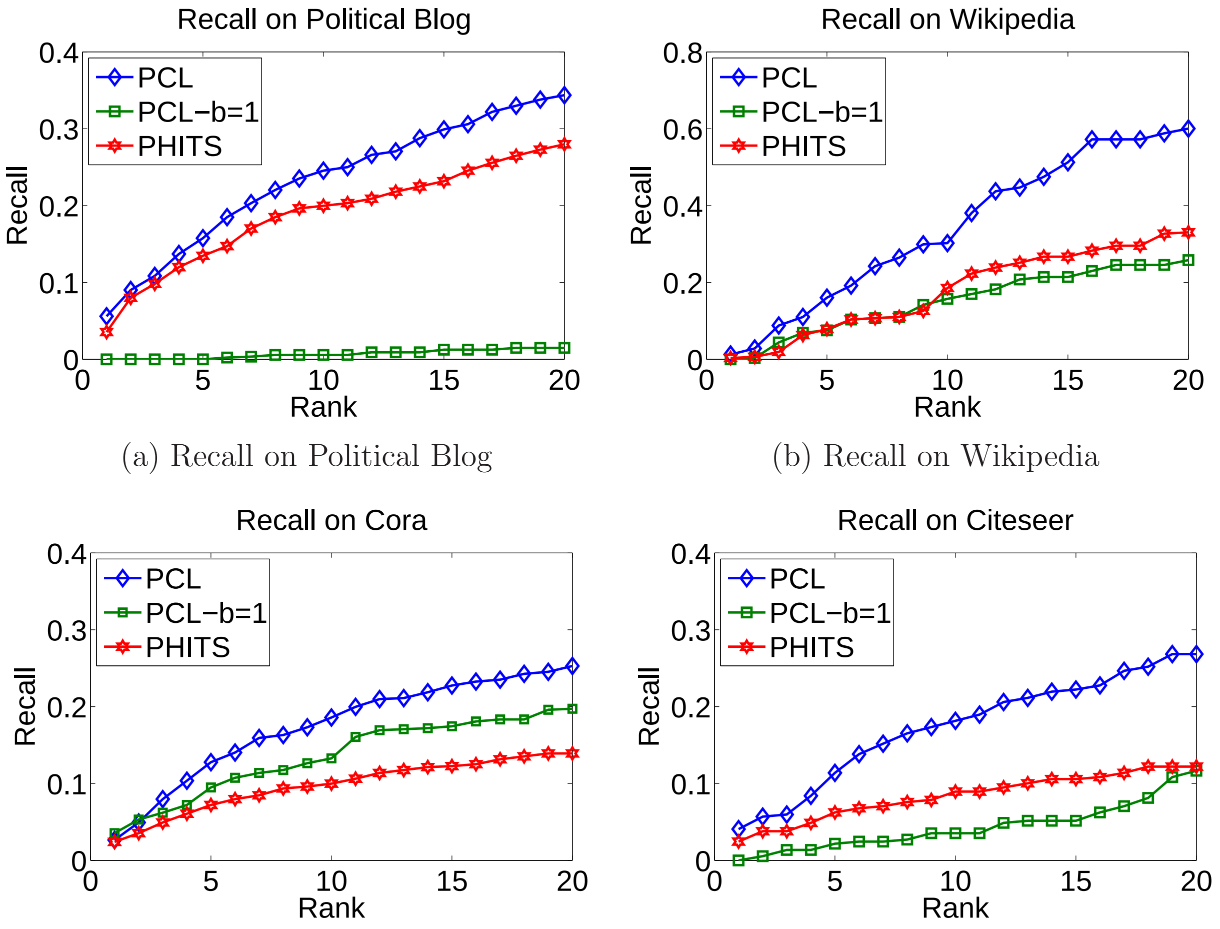
5.5.4 Community Detection

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 [47].

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 [17], LDA-Link-Word refers to the mixed membership model proposed in [28], LCF refers



(a) Recall on Political Blog

(b) Recall on Wikipedia

(c) Recall on Cora

(d) Recall on Citeseer

Figure 5.1. Recall on the four data sets

to the model proposed in [113], 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 5.4, 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.

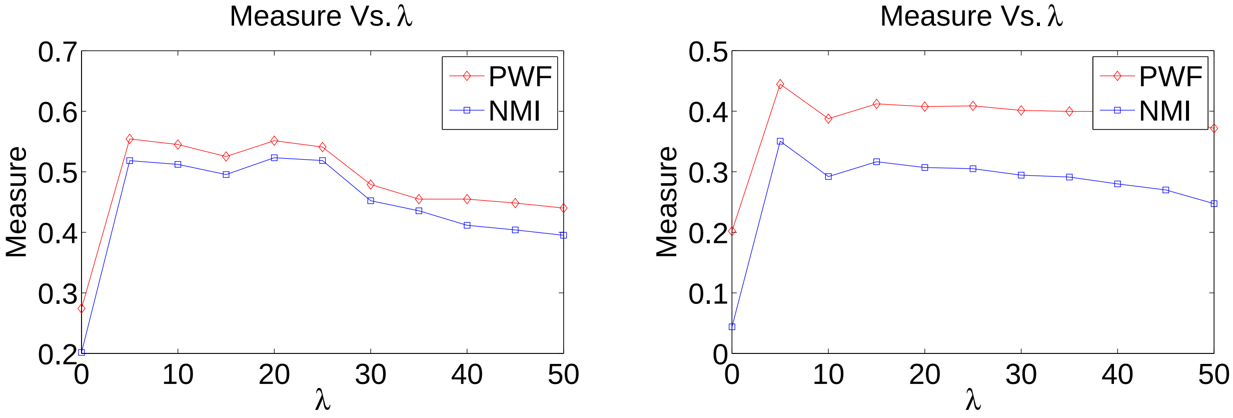
In the implementation, the feature vector used in our model is the original word indicator vector without any transformation; the spectral clustering we used is the normalized cut algorithm [83] (NCUT). For the algorithms that are dependent on some parameters such as the σ parameter in RBF kernel, the 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 5.1 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 5.2. 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 5.2. Partition Measure of PCL-DC vs. λ

5.6 Conclusions

In this chapter, 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.

CHAPTER 6

Clustering for Noisy M&C Linked Data: A Probabilistic Link Prediction Model (I)

6.1 Introduction

In this chapter and the next chapter, we explore the problem of clustering for noisy M&C linked data. Learning from M&C linked data has been studied extensively and has found its application in distance metric learning [102], constrained clustering [8], and kernel learning [42]. The M&C link information include the must-link for pairs in the same class and cannot-link for pairs in different classes. They are also termed as *positive (pairwise) constraints* for must-link pairs, and *negative (pairwise) constraints* for cannot-link pairs. M&C link information can often be derived from data, making it more attractive than the standard setup of supervised learning. For instance, in classifying research articles, we can derive the pairwise constraints based on the citations between papers.

Although various algorithms have been proposed for learning from M&C linked data, most of them assume *perfect* M&C link information. In contrast, in this study, we focus on the problem of **learning from noisy M&C link information** in which some of the pairwise constraints are labeled incorrectly. This is important because the pairwise constraints extracted from data tend to be noisy and inaccurate. In the example of classifying research articles with pairwise constraints constructed from paper citations, the cited paper may not share the same research topic as the citing paper.

To cluster the noisy M&C linked data, we proposed to learn a combined kernel from the noisy M&C linked data. We proposed a probabilistic link prediction model to learn the combination weights based on a generalized maximum entropy model or equivalently a regularized logistic regression model. We proposed two different approaches for estimating the sufficient statistics from the noisy M&C link information under different assumptions. We show that under the claimed assumptions, the probabilistic model trained from the noisy M&C link information converges to that trained from the perfect M&C link information. Extensive experimental results verify the efficacy of the proposed framework for clustering noisy M&C linked data.

The remainder of this chapter is organized as follows. In section 6.2, we present the probabilistic link predication model for learning from noisy M&C link information. We present experimental results in section 6.3, 7.3, and conclude our study in section 7.4.

6.2 Learning from Noisy M&C Linked Data

We start with the basic formulation for maximum entropy learning from perfect M&C link information, followed by its generalization and its equivalence to regularized logistic regression model. We then extend to the case of noisy M&C link information. For the purpose of presentation, we first introduce the notations that are used throughout this chapter.

6.2.1 Notations

Let $\mathcal{D} = \{\mathbf{x}_i \in \mathcal{X}, i = 1, \cdots N\}$ be a collection of data points, $\mathcal{P} = \{(\mathbf{x}_i^1, \mathbf{x}_i^2, \widehat{y}_i) | \mathbf{x}_i^1, \mathbf{x}_i^2 \in \mathcal{D}, i = 1, \ldots, n, \widehat{y}_i \in \{+1, -1\}\}$ be a collection of observed labeled pairs. We slightly abuse the terminology of labeled and unlabeled examples by referring to the examples in $\mathcal{D}$ that also occur in $\mathcal{P}$ as labeled examples, and to the remaining examples in $\mathcal{D}$ as unlabeled examples. We denote by y_i the true label for the pair $(\mathbf{x}_i^1, \mathbf{x}_i^2)$. We refer to the pairs with $y_i = +1$ as *perfect must-link pairs* or *perfect positive constraints* and the pairs with $y_i = -1$ as *perfect cannot-link pairs* or *perfect negative constraints*. Similarly, we refer to the pairs with $\widehat{y}_i = +1$

Table 6.1. Notations

Expectation	Empirical Average
$\mathrm{E}^{\delta}_{y}[\mathbf{k}] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2,\mathrm{Y}}[\delta(\mathrm{Y},y)\mathbf{k}(\mathrm{X}^1,\mathrm{X}^2)]$	$a^{\delta}_{y}[\mathbf{k}] = \frac{1}{n}\sum_{i=1}^{n}\delta(y_i,y)\mathbf{k}_i$
$\widehat{\mathrm{E}}^{\delta}_{y}[\mathbf{k}] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2,\widehat{\mathrm{Y}}}[\delta(\widehat{\mathrm{Y}},y)\mathbf{k}(\mathrm{X}^1,\mathrm{X}^2)]$	$\widehat{a}^{\delta}_{y}[\mathbf{k}] = \frac{1}{n}\sum_{i=1}^{n}\delta(\widehat{y}_i,y)\mathbf{k}_i$
$\mathrm{E}[\mathbf{k}] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2}\left[\mathbf{k}(\mathrm{X}^1,\mathrm{X}^2)\right]$	$a[\mathbf{k}] = \frac{1}{n}\sum_{i=1}^{n}\mathbf{k}_i$
$\mathrm{E}^{o}[\mathbf{k}] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2,\mathrm{Y}}\left[\mathrm{Y}\mathbf{k}(\mathrm{X}^1,\mathrm{X}^2)\right]$	$a^{o}[\mathbf{k}] = \frac{1}{n}\sum_{i=1}^{n}y_i\mathbf{k}_i$
$\mathrm{E}^{c}_{y}[\mathbf{k}] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2\vert\mathrm{Y}=y}\left[\mathbf{k}(\mathrm{X}^1,\mathrm{X}^2)\right]$	—–
$\widehat{\mathrm{E}}^{c}_{y}[\mathbf{k}] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2\vert\widehat{\mathrm{Y}}=y}\left[\mathbf{k}(\mathrm{X}^1,\mathrm{X}^2)\right]$	$\widehat{a}^{c}_{y}[\mathbf{k}] = \dfrac{\sum_{i=1}^{n}\delta(\widehat{y}_i,y)\mathbf{k}_i}{\sum_{i=1}^{n}\delta(\widehat{y}_i,y)}$

as *noisy must-link pairs* or *noisy positive constraints* and the pairs with $\widehat{y}_i = -1$ as *noisy cannot-link pairs* or *noisy negative constraints* . We use $\bar{y} = -y$ for complement of y. We use $\kappa_j(\mathbf{x}^1,\mathbf{x}^2)$ for the $j^{\mathrm{th}} \in \{1,\cdots,m\}$ jth kernel feature function defined on $\mathcal{X}\times\mathcal{X}$. We denote by $\mathbf{k}_i = (\kappa_1(\mathbf{x}^1_i,\mathbf{x}^2_i),\cdots,\kappa_m(\mathbf{x}^1_i,\mathbf{x}^2_i))^\top$ the feature vector for pair $(\mathbf{x}^1_i,\mathbf{x}^2_i)$. Throughout the paper, we use capital letters $\mathrm{X},\mathrm{Y},\widehat{\mathrm{Y}}$ for the corresponding random variables. In the sequel, we use the notations defined in Table 6.1, where $\delta(y_i,y)$ is the Kronecker delta function that outputs 1 if $y_i = y$ and zero, otherwise. We let $a^{\delta}_{y}[\kappa_j]$ denote the jth element in $a^{\delta}_{y}[\mathbf{k}]$, and similarly for $\widehat{a}^{\delta}_{y}[\kappa_j]$. The empirical averages $a^{\delta}_{y}[\mathbf{k}]$ and $a^{o}[\mathbf{k}]$ in Table 6.1 are referred to sufficient statistics.

6.2.2 Generalized Maximum Entropy Model

We proposed to learn a probabilistic link predication model $\Pr(\mathrm{Y}|\mathrm{X}^1,\mathrm{X}^2)$, i.e. given a pair $\mathrm{X}^1,\mathrm{X}^2$, how likely they are related by must-link or cannot-link. We first consider the maximum entropy model for learning the conditional link prediction model from perfect M&C link information. We cast the problem of learning from M&C link information into a binary classification problem where the objective is to classify each pair $(\mathbf{x}^1_i,\mathbf{x}^2_i)$ into two categories, i.e., a positive pair ($y_i = +1$) and a negative pair ($y_i = -1$). Using maximum entropy model, we aim to learn the conditional distribution $\Pr(\mathrm{Y} = y|\mathrm{X}^1,\mathrm{X}^2)$, which leads to the following optimization problem:

$$\begin{aligned} \max \quad & \sum_{i=1}^{n} H(p|\mathbf{x}_i^1, \mathbf{x}_i^2) \\ s.t. \quad & \frac{1}{n}\sum_{i=1}^{n} p(y|\mathbf{x}_i^1, \mathbf{x}_i^2)\kappa_j(\mathbf{x}_i^1, \mathbf{x}_i^2) = a_y^{\delta}[\kappa_j], \forall y, j \end{aligned} \tag{6.1}$$

where $H(p|\mathbf{x}_i^1, \mathbf{x}_i^2) = -\sum_y p(y|\mathbf{x}_i^1, \mathbf{x}_i^2) \ln p(y|\mathbf{x}_i^1, \mathbf{x}_i^2)$. The solution to (6.1) is given by

$$p(y|\mathbf{x}_i^1, \mathbf{x}_i^2) = \frac{1}{1+\exp(-y\mathbf{w}^\top \mathbf{k}_i)} = \frac{\exp(y\mathbf{w}^\top \mathbf{k}_i/2)}{\exp(y\mathbf{w}^\top \mathbf{k}_i/2) + \exp(-y\mathbf{w}^\top \mathbf{k}_i/2)}$$

where $\mathbf{w} \in \mathbb{R}^m$ are the dual variables and are obtained by solving the following optimization problem,

$$\min_{\mathbf{w}\in\mathbb{R}^m} \sum_{i=1}^{n} \ln\left(1+\exp(-y_i\mathbf{w}^\top \mathbf{k}_i)\right) = -\frac{1}{2}\sum_{i=1}^{n} y_i\mathbf{w}^\top \mathbf{k}_i + \sum_{i=1}^{n} \ln\left(\sum_y \exp(y\mathbf{w}^\top \mathbf{k}_i/2)\right)$$

One major problem with the maximum entropy model in (6.1) is the equality constraint, which is unlikely to hold if for each pair $(\mathbf{x}_i^1, \mathbf{x}_i^2)$, y_i is a random sample from the distribution $p(y|\mathbf{x}_i^1, \mathbf{x}_i^2)$. We denote by $a_y^p[\kappa_j]$ the left side of equality constraint in problem (6.1), i.e.

$$a_y^p[\kappa_j] = \frac{1}{n}\sum_{i=1}^{n} p(y|\mathbf{x}_i^1, \mathbf{x}_i^2)\kappa_j(\mathbf{x}_i^1, \mathbf{x}_i^2)$$

The following theorem shows that $a_y^p[\kappa_j]$ and $a_y^{\delta}[\kappa_j]$ could differ significantly if n is small. The difference between the two quantities will diminish only when n approaches infinity.

Theorem 7. *Assume $(\mathbf{x}_i^1, \mathbf{x}_i^2, y_i)$ are i.i.d. samples from an unknown distribution $P(\mathrm{X}^1, \mathrm{X}^2, \mathrm{Y})$, and the kernel function is bounded $|\kappa_j(\mathbf{x}^1, \mathbf{x}^2)| \le R, \forall j$, then the equality constraint in (6.1) for any j and y holds with probability 1 when the number of instances approaches infinity. In particular, for any $\epsilon > 0$ we have*

$$\Pr\left(\left|a_y^p[\kappa_j] - a_y^{\delta}[\kappa_j]\right| \ge \epsilon\right) \le 4\exp\left(-\frac{\epsilon^2 n}{8R^2}\right)$$

The theorem can be proved by noting that $\mathrm{E}[a_y^{\delta}[\kappa_j]] = \mathrm{E}[a_y^p[\kappa_j]]$ and applying McDiarmid's inequality. Details are provided in the appendix. To address the case that $a_y^p[\kappa_j]$ and $a_y^{\delta}[\kappa_j]$

could be different, we propose a generalization to the traditional maximum entropy model in (6.1). Given the finite number of training data, we relax the equality constraints in (6.1) into inequality ones, leading to the following formulation for learning from side information

$$\begin{aligned} \max \quad & \frac{1}{n}\sum_{i=1}^{n} H(p|\mathbf{x}_i^1, \mathbf{x}_i^2) - \frac{1}{2\lambda}\sum_{y} \|\epsilon_y\|^2 \qquad (6.2) \\ s.t. \quad & \frac{1}{n}\sum_{i} p(y|\mathbf{x}_i^1, \mathbf{x}_i^2)\kappa_j(\mathbf{x}_i^1, \mathbf{x}_i^2) \geq a_y^{\delta}[\kappa_j] - \epsilon_{yj},\ \forall y, j \end{aligned}$$

where $\epsilon_y = (\epsilon_{y1}, \ldots, \epsilon_{ym})^\top$ and $\|\cdot\|$ is a norm that measures the length of vector ϵ_y. The key features of the generalized maximum entropy model in (6.2) are:

- Replacing equality constraints with inequality ones. As a result, we have

$$a_y^{\delta}[\kappa_j] - \epsilon_{yj} \leq a_y^{p}[\kappa_j] \leq a_y^{\delta}[\kappa_j] + \epsilon_{\bar{y}j}.$$

  Note that although only one side inequality is included in (6.2), the upper bound of $a_y^p[\kappa_j]$ can be easily derived by using the relation $a_y^p[\kappa_j] + a_{\bar{y}}^p[\kappa_j] = a_y^\delta[\kappa_j] + a_{\bar{y}}^\delta[\kappa_j]$.

- The positive dummy variables ϵ are introduced to account for the difference between the two empirical means $a_y^p[\kappa_j]$ and $a_y^\delta[\kappa_j]$. A regularization term $\|\epsilon_y\|^2/(2\lambda)$ is introduced into the objective in order to determine these variables automatically.

We further justify the generalized maximum entropy model by showing it is equivalent to the regularized logistic regression model.

Proposition 3. *When $\|\cdot\| = \|\cdot\|_2$, the dual problem of (6.2) is equivalent to the regularized logistic regression model, i.e.,*

$$\max_{\mathbf{w}\in\mathbb{R}^m} \frac{1}{n}\sum_{i=1}^{n} \ln p(y_i|\mathbf{x}_i^1, \mathbf{x}_i^2) - \frac{\lambda}{2}\|\mathbf{w}\|_2^2 \qquad (6.3)$$

or equivalently,

$$\max_{\mathbf{w}\in\mathbb{R}^m} \quad \frac{1}{2}\mathbf{w}^\top a^o[\mathbf{k}] - \frac{\lambda}{2}\|\mathbf{w}\|_2^2 - \frac{1}{n}\sum_{i=1}^{n}\ln\sum_{y}\exp\left(\frac{1}{2}y\mathbf{w}^\top\mathbf{k}_i\right) \qquad (6.4)$$

6.2.3 Estimating the Sufficient Statistics

In this section, we extend the framework of learning a probabilistic link prediction model to the case when pairwise constraints are noisy, i.e., $\widehat{y}_i \neq y_i$ for some pairs. The strategies we used in this section, is to estimate the sufficient statistics $a_y^\delta[\mathbf{k}]$ or $a^o[\mathbf{k}]$ from the noisy labels without having to know which labels are incorrect. We present an approach for estimating the sufficient statistics under certain assumptions.

In order to estimate $a_y^\delta[\mathbf{k}]$ in the case of noisy M&C link information, we make the following assumptions.

Assumption 1. *We assume (1.a)* $\Pr(\widehat{\mathrm{Y}}|\mathrm{X}^1, \mathrm{X}^2, \mathrm{Y}) = \Pr(\widehat{\mathrm{Y}}|\mathrm{Y})$*, (1.b)* $\Pr(\widehat{\mathrm{Y}} = y|\mathrm{Y} = y) = c_y$ *is given.*

In the above assumption, (1.a) assumes $\widehat{\mathrm{Y}}$ is conditionally independent of $(\mathrm{X}^1, \mathrm{X}^2)$ given Y, (1.b) assumes the group-level knowledge about the noise in the pairwise constraintsWith these two assumptions, the following theorem shows that it is possible to express empirical mean $a_y^\delta[\mathbf{k}]$ in terms of $\widehat{a}_y^\delta[\mathbf{k}]$, i.e., the empirical mean estimated from the noisy side information.

Theorem 8. *Assuming*$(\mathbf{x}_i, y_i, \widehat{y}_i), i = 1, \ldots, n$ *are i.i.d samples, we have, with a probability at least* $1 - \delta$,

$$\left\| a_y^\delta[\mathbf{k}] - b_y^\delta[\mathbf{k}] \right\|_2 \leq \sqrt{\frac{8mR^2}{(c_+ + c_- - 1)^2 n} \ln\left(\frac{4m}{\delta}\right)}, \forall y$$

where

$$b_y^\delta[\mathbf{k}] = \frac{\widehat{a}_y^\delta[\mathbf{k}]}{(c_y + c_{\bar{y}} - 1)} - \frac{1}{n}\frac{(1 - c_{\bar{y}})}{c_y + c_{\bar{y}} - 1}\sum_{i=1}^{n} \mathbf{k}(\mathbf{x}_i^1, \mathbf{x}_i^2)$$

The proof can be found in the appendix. As indicated by Theorem 8, under Assumption 1, we can approximate $a_y^\delta[\mathbf{k}]$ by $b_y^\delta[\mathbf{k}]$. It is interesting to note that the convergence rate is $O\left(1/[|c_y + c_{\bar{y}} - 1|\sqrt{n}]\right)$, not $O(1/\sqrt{n})$. Similar to Theorem 8, we can have the following corollary to bound the difference between $a_y^p[\mathbf{k}]$ and $b_y^\delta[\mathbf{k}]$.

Corrolary 9. *Assuming* $(\mathbf{x}_i, y_i, \widehat{y}_i), i = 1, \ldots, n$ *are i.i.d samples, we have, with a probability at least* $1 - \delta$,

$$\left\| a_y^p[\mathbf{k}] - b_y^\delta[\mathbf{k}] \right\|_2 \leq \sqrt{\frac{8mR^2}{(c_+ + c_- - 1)^2 n} \ln\left(\frac{4m}{\delta}\right)}, \quad \forall y$$

Given $b_y^\delta[\mathbf{k}]$ is an estimate of $a_y^\delta[\mathbf{k}]$ and $a^o[\mathbf{k}] = a_+^\delta[\mathbf{k}] - a_-^\delta[\mathbf{k}]$, we can compute an estimate of $a^o[\mathbf{k}]$ under **Assumption 1** by

$$b^o[\mathbf{k}] = b_+^\delta[\mathbf{k}] - b_-^\delta[\mathbf{k}] = \frac{1}{c_+ + c_- - 1}\widehat{a}^o[\mathbf{k}] - \frac{c_+ - c_-}{c_+ + c_- - 1}a[\mathbf{k}] \tag{6.5}$$

Similar to Theorem 8, we have the following corollary to bound the difference between $a^o[\mathbf{k}]$ and $b^o[\mathbf{k}]$

Corrolary 10. *Assuming* $(\mathbf{x}_i, y_i, \widehat{y}_i), i = 1, \ldots, n$ *are i.i.d samples, we have, with a probability at least* $1 - \delta$,

$$\|a^o[\mathbf{k}] - b^o[\mathbf{k}]\|_2 \leq \sqrt{\frac{8mR^2}{(c_+ + c_- - 1)^2 n} \ln\left(\frac{4m}{\delta}\right)}$$

With theorem 8, corollary 9 and corollary 10, we finally reach to the following generalized maximum entropy for learning from noisily labeled data

$$\begin{aligned} \widehat{p}(y|\mathbf{x}^1, \mathbf{x}^2) = \arg \max_{p(y|\mathbf{x})} \quad & \frac{1}{n}\sum_{i=1}^n H(p|\mathbf{x}_i^1, \mathbf{x}_i^2) - \frac{1}{2\lambda}\sum_y \|\boldsymbol{\epsilon}_y\|^2 \\ s.t. \quad & \frac{1}{n}\sum_{i=1}^n p(y|\mathbf{x}_i^1, \mathbf{x}_i^2)\mathbf{k}(\mathbf{x}_i^1, \mathbf{x}_i^2) \geq b_y^\delta[\mathbf{k}] - \boldsymbol{\epsilon}_y, \forall y \end{aligned} \tag{6.6}$$

or the following regularized logistic regression model for learning from noisily labeled data

$$\mathbf{w}_b^* = \arg \min_{\mathbf{w} \in \mathbb{R}^m} \frac{\lambda}{2}\|\mathbf{w}\|_2^2 - \frac{1}{2}\mathbf{w}^\top b^o[\mathbf{k}] + \frac{1}{n}\sum_{i=1}^n \ln\left[\exp\left(\frac{1}{2}y\mathbf{w}^\top \mathbf{k}_i\right)\right] \tag{6.7}$$

6.2.4 Convergence Analysis

The resulting conditional distribution $p(y|\mathbf{x}^1, \mathbf{x}^2)$ from (6.6) is given by

$$\widehat{p}(y = 1|\mathbf{x}^1, \mathbf{x}^2) = \frac{\exp(\mathbf{w}_b^{*\top}\mathbf{k}(\mathbf{x}^1, \mathbf{x}^2))}{1 + \exp(\mathbf{w}_b^{*\top}\mathbf{k}(\mathbf{x}^1, \mathbf{x}^2))} \tag{6.8}$$

Next, we show how the solution $\mathbf{w}$ will be affected when replacing $a_y^\delta[\mathbf{k}]$ with $b_y^\delta[\mathbf{k}]$, i.e., the empirical mean computed from the noisy pairwise constraints.

Theorem 11 (Lemma 6[78]). *Let $\mathbf{w}_a^*$ be the solution to (6.4) with $a^o[\mathbf{k}]$, and $\mathbf{w}_b^*$ be the solution to (6.7) with $b^o[\mathbf{k}]$. We have*

$$\|\mathbf{w}_a^* - \mathbf{w}_b^*\|_2 \leq \frac{1}{\lambda}\|a^o[\mathbf{k}] - b^o[\mathbf{k}]\|_2$$

The proof for the theorem as well as the following theorems can be found in the appendix. Combining Theorem 11 and Corollary 10, we have the following theorem showing the impact of replacing $a^o[\mathbf{k}]$ with $b^o[\mathbf{k}]$.

Theorem 12. *Let $\widehat{p}(y|\mathbf{x}^1, \mathbf{x}^2)$ be the conditional model derived from noisy pairwise constraints using (6.6) using $\|\cdot\|_2$, and $p(y|\mathbf{x}^1, \mathbf{x}^2)$ be the conditional model derived from the perfect pairwise constraints using (6.2). Under Assumption 1, with probability $1-\delta$, for any $\mathbf{x}^1$, $\mathbf{x}^2$ and y, we have*

$$\left|\widehat{p}(y|\mathbf{x}^1, \mathbf{x}^2) - p(y|\mathbf{x}^1, \mathbf{x}^2)\right| \leq \frac{mR^2}{\lambda c}\sqrt{\frac{8}{n}\ln\left(\frac{4m}{\delta}\right)}$$

where $c = |c_+ + c_- - 1|$.

As indicated by the above theorem, the difference between two conditional models will be reduced at the rate of $1/[|c_+ + c_- - 1|\sqrt{n}]$. Finally, since our algorithm depends on the knowledge of c_+ and c_-, we further analyze the behavior of the proposed algorithm with inaccurate estimation of c_+ and c_-. We denote by $\widehat{c}_+$ and $\widehat{c}_-$ the estimates of c_+ and c_-, respectively. We define $\widetilde{p}(y|\mathbf{x}^1, \mathbf{x}^2)$ the conditional model derived from the noisy side information using $\widehat{c}_+$ and $\widehat{c}_-$. We measure the difference (c_+, c_-) and their estimates by $\Delta = \max(|c_+ - \widehat{c}_+|, |c_- - \widehat{c}_-|)$. The next theorem shows the difference between $\widetilde{p}(y|\mathbf{x}^1, \mathbf{x}^2)$ and $p(y|\mathbf{x}^1, \mathbf{x}^2)$.

Theorem 13. *Let $\|\cdot\| = \|\cdot\|_2$. Let $\widetilde{p}(y|\mathbf{x}^1, \mathbf{x}^2)$ be the conditional model derived from noisy pairwise constraints with $\widehat{c}_+$ and $\widehat{c}_-$, and $p(y|\mathbf{x}^1, \mathbf{x}^2)$ be the conditional model derived from the perfect pairwise constraints using (6.2). Assume $|c_+ + c_- - 1| \geq \rho$ with $\rho \geq 0$ and $\Delta \leq \rho/4$. Under Assumption 1, with probability $1-\delta$, for any $\mathbf{x}^1$, $\mathbf{x}^2$ and y, we have*

$$\left|\widetilde{p}(y|\mathbf{x}^1, \mathbf{x}^2) - p(y|\mathbf{x}^1, \mathbf{x}^2)\right| \leq \frac{mR^2}{\lambda c}\sqrt{\frac{8}{n}\ln\frac{4m}{\delta}} + \frac{mR^2\Delta(8+2c)}{\lambda\rho^2}$$

We finally mention that $\mathbf{w}$ can serve as weights for combining kernels, i.e., $\sum_j \mathbf{w}_j \kappa_j$. However, the combined kernel may not be positive semi-definite, because some weights $\mathbf{w}_j$ are negative. To ensure the combined kernel to be valid, we introduce one more constraint $\mathbf{w}_j \geq 0, j = 1, \ldots, m$ to the optimization problem in (6.6). The optimization problems are solved by Nesterov method [70].

6.3 Experiments

We evaluate the proposed algorithm by clustering the linked documents. We first present the experiments on clustering linked documents with noisy pairwise constraints derived from the link information. We then examine the behavior of the proposed algorithm in more details. Before presenting the experimental results, we first introduce the data sets, baselines and evaluation metric.

Data Sets We select three linked document data sets, i.e. Cora, Citeseer, Terrorist Attacks(TeAt) for our evalution. We choose these data sets because they have relatively low noise ($20\% \sim 35\%$) in their pairwise constraints derived from links that satisfies the condition $c_+ + c_- - 1 > 0$ in Assumption 1. They were processed by the research group of Lise Getoor (see `www.cs.umd.edu/projects/linqs/projects/lbc/`). Each data set contains (1) a set of documents described by binary vectors indicating the presence and absence of words from a dictionary, (2) links among documents (e.g., citations between research articles), and (3) the class assignment for each document. The statistics of these three data sets are summarized in Table 1. In the experiments, the attributes for each document are normalized by first dividing the sum of the attributes and then taking the square root [47].

Evaluation In order to evaluate the proposed algorithm, we apply it to kernel learning as described at the end of Section 3. In particular, for each attribute j, we construct a linear kernel matrix $\kappa_j(\mathbf{x}^1, \mathbf{x}^2) = \mathbf{x}^1[j]\mathbf{x}^2[j]$ for paired documents $(\mathbf{x}^1, \mathbf{x}^2)$, where $\mathbf{x}[j]$ is the j^{th} normalized attribute of document $\mathbf{x}$. The proposed algorithm will be applied to learn the

Table 6.2. Statistics of Data sets

name	#examples	#words	#links	#classes
Cora	2708	1433	5429	7
Citeseer	3312	3703	4732	6
TeAt	1293	106	571	6

combination of multiple kernel matrices from the noisy pairwise constraints derived from links. The ℓ_2 norm is used in the proposed algorithm. Given the learned kernel matrix, a spectral clustering algorithm [84] is applied for document clustering. We evaluate the clustering result by comparing it to the class assignment information provided in each data set. *Normalized mutual information*(NMI) [107] is used as our evaluation metric. For all the experiments, we set γ in the proposed algorithm to be $0.01/c^2$, where $c = c_+ + c_- - 1$.

Baseline We compare the proposed algorithm to the following metric/kernel learning algorithms: (a) **GDM**, the global distance metric learning algorithm [102], (b) **DCA**, the discriminative component analysis algorithm [43], (c) **ITML**, the information theoretic metric learning algorithm proposed by [21], and (d) **SKL**, the spectral kernel learning algorithm [44]. For fair comparison, the distance metric A learned by the metric learning algorithms will be used to construct a kernel matrix $K = XAX^\top$, where X is the data matrix, and the same spectral clustering algorithm will be applied to K for document clustering. We also evaluate the proposed algorithm against the metric pairwise constrained K-means clustering algorithm [8], referred to as **MPCK**. In order to improve the robustness of **MPCK** to noisy constraints, we follow [61] and weight the noisy positive constraints and the noisy negative constraints by c_+, c_- respectively to reduce their impact on the clustering results. As the reference point, we compute a linear kernel for both labeled and unlabeled examples, without using the provided pairwise constraints. We refer to this baseline as **base**. Finally, we refer to as **GMEns** the proposed generalized maximum entropy model for learning from noisy side information, and as **GMEs** the generalized maximum entropy model without considering the noise in side information. All the experiments are run five times and the clustering accuracy averaged over five runs is reported in our study.

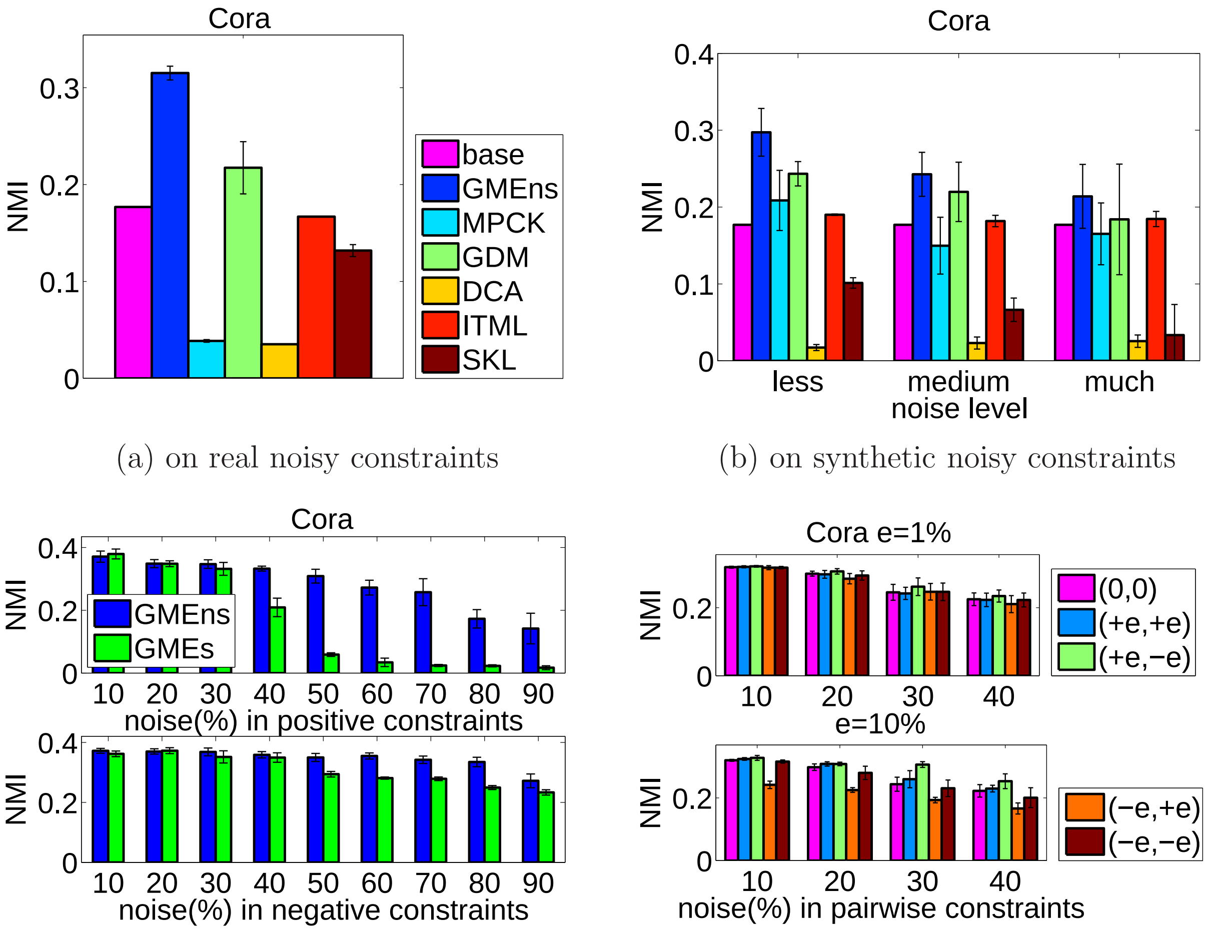
6.3.1 Experiments with Real Noise

We conduct experiments of document clustering with the noisy pairwise constraints derived from the links between documents. In particular, we use all the linked document pairs as the positive constraints. The same number of document pairs without link are sampled to construct the negative constraints. To obtain the noise levels of the pairwise constraints, we sample a total of 100 pairwise constraints and estimate c_+ and c_- based on the correctness of the sampled constraints. These sampled pairwise constraints with their true labels are also used by the other baseline methods for computing distance metrics and kernel matrices. Figures 6.1(a), 6.2(a) and 6.3(a) show the clustering accuracy measured in NMI for the three data sets. The mean values of the estimated c_+ and c_- are listed under each figure. We observe that given the noisy pairwise constraints, all the algorithms except **ITML** perform significantly worse on at least one data set than the reference method **base**. In contrast, the proposed algorithm for learning from noisy pairwise constraints outperforms the reference method significantly for all three data sets. We thus conclude that the proposed algorithm is overall more robust to noise in the side information.

6.3.2 Experiments with Synthetic Noise

In this section, we examine the robustness of the proposed algorithm to (a) different noise levels in synthetically generated pairwise constraints, and (b) the estimated values for c_+ and c_-.

Robustness to the Noise We first sample $10,000$ pairwise constraints from each data set, with $5,000$ positive constraints and $5,000$ negative constraints. Random noise is introduced to the synthetic constraints by randomly flipping the label of a pair with a probability $p\%$, where $p\%$ specifies the noise level. We set c_+ and c_- to be $1 - p\%$, with the assumption that the knowledge of noise level is perfect. To examine the impact of noisy positive



(a) on real noisy constraints

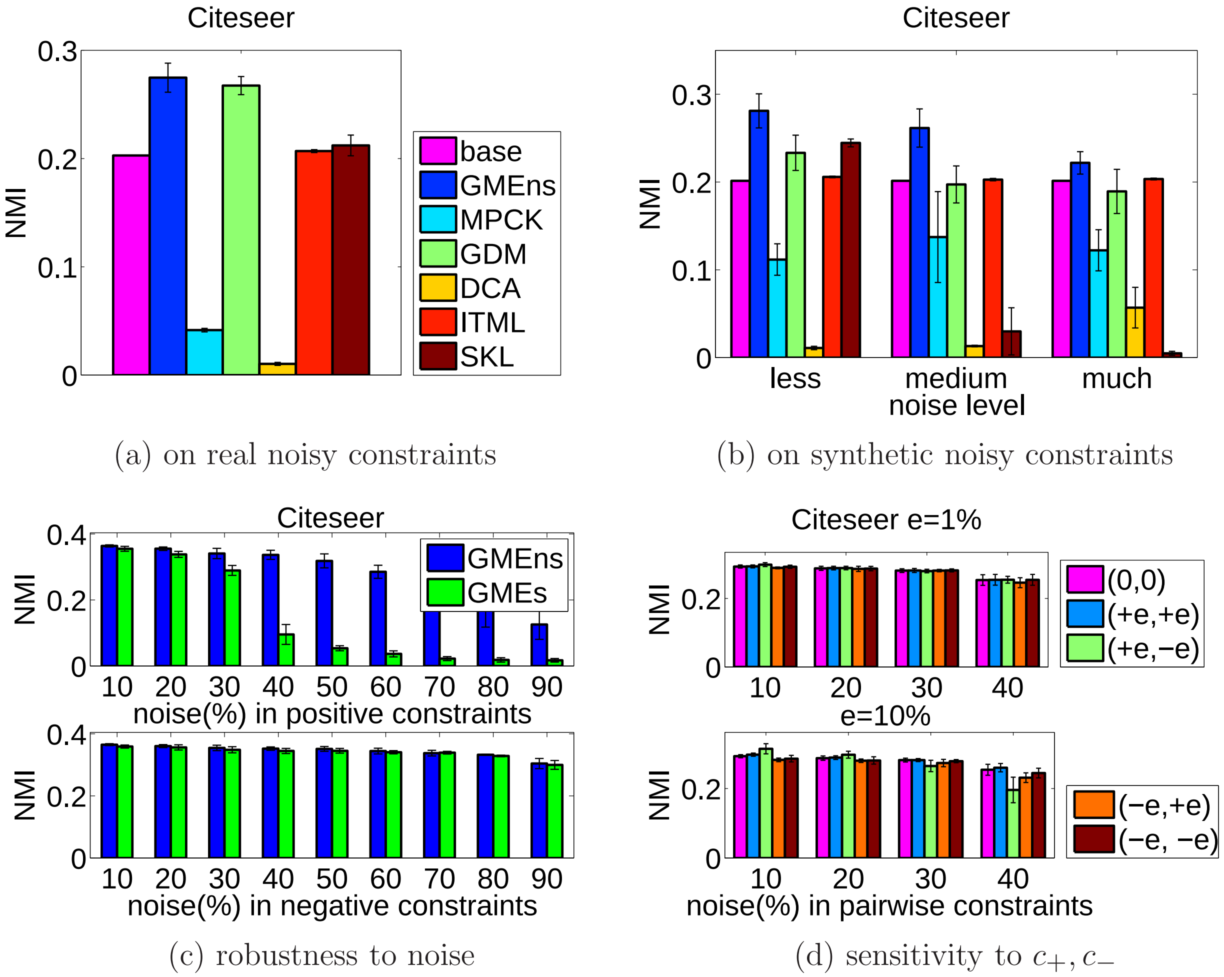
(b) on synthetic noisy constraints

(c) robustness to noise

(d) sensitivity to c_+, c_-

Figure 6.1. Experimental results on Cora data set

constraints and noisy negative constraints separately, for each data set, with a given noise level $p\%$, we conduct two experiments, one with corrupted positive constraints but perfect negative constraints, and the other with corrupted negative constraints but perfect positive constraints. Figures 6.1(c), 6.2(c) and 6.3(c) compare the clustering results for **GMEns** and **GMEs** with the noise levels in the synthetic pairwise constraints varied from 10% to 90% on the three data sets. We observe that **GMEns**, the generalized maximum entropy model for noisy side information, is significantly more robust to the noise in the pairwise constraints than **GMEs** which does not take into account the noise in side information. We also observe that the noisy positive constraints have significantly higher adverse impact on the clustering



(a) on real noisy constraints

(b) on synthetic noisy constraints

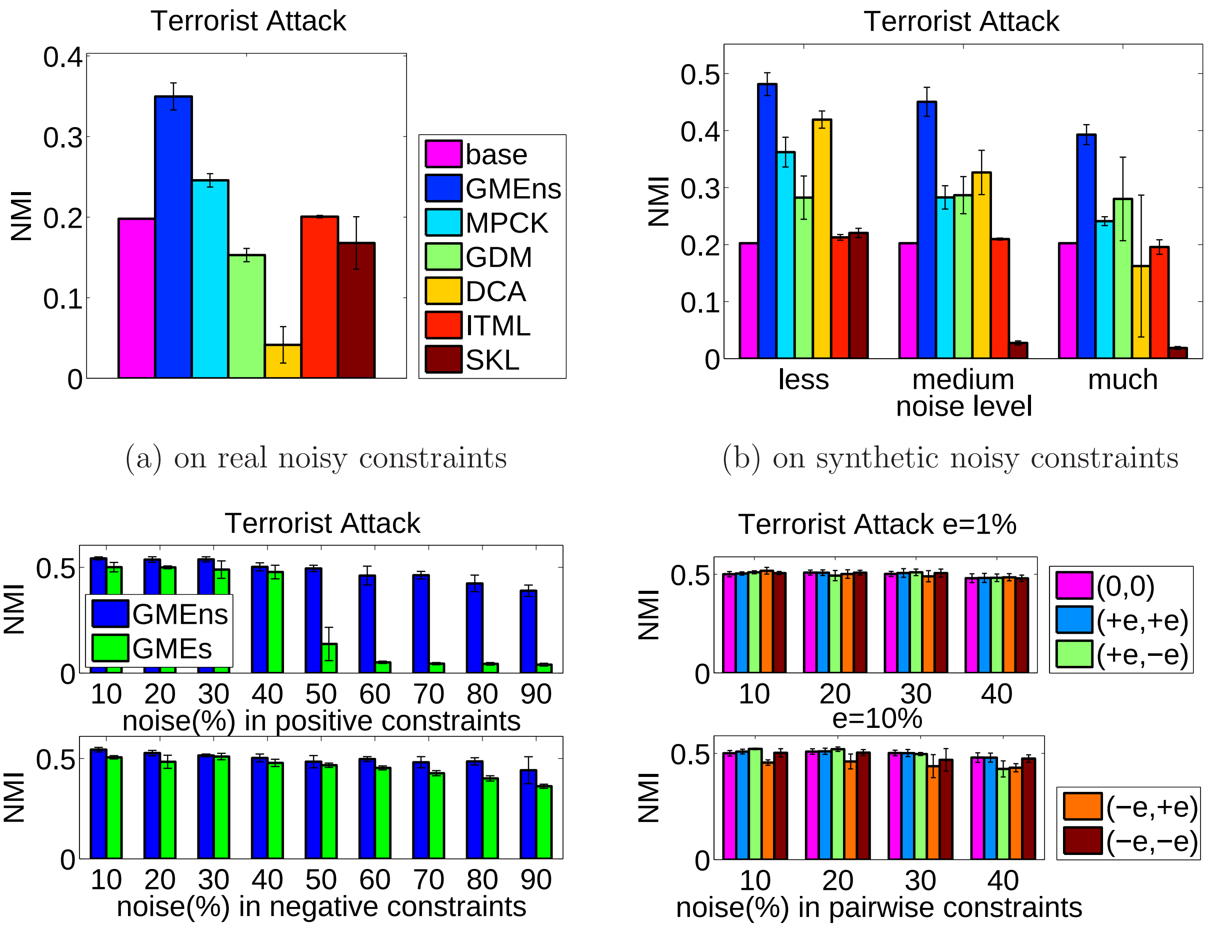
(c) robustness to noise

(d) sensitivity to c_+, c_-

Figure 6.2. Experimental results on Citeseer data set

results than the noisy negative constraints.

Sensitivity to c_+, c_- We use the same set of $10,000$ randomly sampled pairwise constraints for this study. We add the same noise level to both positive constraints and negative constraints. To investigate the sensitivity to c_+, c_-, instead of setting them to be $1 - p\%$, we perturb these parameters by setting them to be $(1 - p\%)(1 \pm e)$. Figures 6.1(d), 6.2(d) and 6.3(d) show the results of **GMEns** on the three data sets with four noise levels $p\% = 10\% \sim 40\%$ for $e = 1\%, 10\%$. We observe that **GMEns** is overall robust to modest perturbation level, making the proposed algorithm applicable even when the assumed noise levels are inaccurate.



(a) on real noisy constraints

(b) on synthetic noisy constraints

(c) robustness to noise

(d) sensitivity to c_+, c_-

Figure 6.3. Experimental results on Terrorist Attack data set

Finally, we compare the proposed algorithm to the baselines on the synthetic noisy constraints by varying the level of noise. Due to the fact that some of the baseline algorithms are time consuming, one thousand pairs are sampled for positive constraints and negative constraints, respectively. We show the results on the noise added to the positive constraints due to its stronger effect on the performance. Figures 6.1(b), 6.2(b) and 6.3(b) show the clustering results of all algorithms at three noise levels: low(10%), medium(40%), high(70%) on the three data sets. We observe that the proposed algorithm is able to outperform all the baseline algorithms for all the cases.

6.4 Conclusions

In this chapter, we have proposed a generalized maximum entropy model for learning from noisy M&C link information, and applied it to learning an optimal weight for combining multiple kernels for clustering. Our theoretical analysis shows that the model trained from the noisy link information converges to the model trained from the perfect link information. Extensive experimental results verify the efficacy of the proposed model.

CHAPTER 7

Clustering for Noisy M&C Linked Data: A Probabilistic Link Prediction Model (II)

In this chapter, we present an alternate approach for estimating the feature statistics under different assumptions about the data and the noise. We begin with a description of the notations.

7.1 Notations

Let $\mathcal{D} = \{\mathbf{x}_j \in \mathcal{X}\}$ be a collection of observed data, and $\mathcal{P} = \{(\mathbf{x}_i^1, \mathbf{x}_i^2, \widehat{y}_i) : \mathbf{x}_i^1, \mathbf{x}_i^2 \in \mathcal{D}, \widehat{y}_i \in \{1, -1\}, i = 1, \cdots, n\}$ be a collection of n pairwise constraints, where $\widehat{y}_i$ is the *noisy* label given to the pair $(\mathbf{x}_i^1, \mathbf{x}_i^2)$ indicating if the pair is a positive constraint ($\widehat{y}_i = 1$) or a negative constraint ($\widehat{y}_i = -1$). Let y_i denote the underlying true label for the pair $(\mathbf{x}_i^1, \mathbf{x}_i^2)$ which is unknown in our setting. Let $k_j(\cdot, \cdot) : \mathcal{X} \times \mathcal{X} \to \mathbb{R}, j = 1, \cdots, m$ be a set of m base kernels. Our goal is to learn a combination of the multiple kernels $k(\cdot, \cdot) = \sum_{j=1}^m w_j k_j(\cdot, \cdot)$, given the noisy must-and-cannot links and use the combined kernel to do clustering.

A few more words about the notations used in this study. For any pair $(\mathbf{x}^1, \mathbf{x}^2)$, we use $\mathbf{k}(\mathbf{x}^1, \mathbf{x}^2) = (k_1(\mathbf{x}^1, \mathbf{x}^2), \cdots, k_m(\mathbf{x}^1, \mathbf{x}^2))^\top$ denote the similarities between $\mathbf{x}^1$ and $\mathbf{x}^2$ using all m kernel functions. We use the shorthand $\mathbf{k}_i$ for $\mathbf{k}(\mathbf{x}_i^1, \mathbf{x}_i^2)$. We use $\bar{y} = -y$ for complement of y. We use subscript $_+$ and $_-$ for conditions $_{y=1}$ and $_{y=-1}$, respectively.

7.2 Learning from Noisy M&C Linked Data

7.2.1 A Probabilistic Model

To learn the combination of multiple kernels, we construct a conditional model $\Pr(y|\mathbf{x}^1, \mathbf{x}^2)$ for computing the pairwise classification probability, where $y \in \{1, -1\}$ is the underlying true label to indicate whether the pair $(\mathbf{x}^1, \mathbf{x}^2)$ belongs to the same class or not. Given multiple kernels $k_j(\cdot, \cdot), j = 1, \dots, m$, we formulate $\Pr(y|\mathbf{x}^1, \mathbf{x}^2)$ by

$$\Pr(y|\mathbf{x}^1, \mathbf{x}^2) = \frac{1}{1 + \exp(-y\mathbf{w}^\top \mathbf{k}(\mathbf{x}^1, \mathbf{x}^2))} = \frac{\exp(y\mathbf{w}^\top \mathbf{k}(\mathbf{x}^1, \mathbf{x}^2)/2)}{\exp(-\mathbf{w}^\top \mathbf{k}(\mathbf{x}^1, \mathbf{x}^2)/2) + \exp(\mathbf{w}^\top \mathbf{k}(\mathbf{x}^1, \mathbf{x}^2)/2)}$$

where $\mathbf{w} = (w_1, \cdots, w_m)^\top \in \mathbb{R}_+^m$ are the non-negative weights that need to be learned. We constrain $\mathbf{w}$ to be non-negative to ensure that the resulting combined kernel is positive semi-definite. By optimizing the log-likelihood of the true labels for the observed pairs, we obtain the optimal weights for kernel combination. More specifically, we need to solve the following optimization problem

$$\max_{\mathbf{w} \in \mathbb{R}_+^m} \frac{1}{n} \sum_{i=1}^{n} \ln p(y_i|\mathbf{x}_i^1, \mathbf{x}_i^2) - \frac{\lambda}{2} \|\mathbf{w}\|_2^2 \tag{7.1}$$

or equivalently,

$$\max_{\mathbf{w} \in \mathbb{R}_+^m} \quad \frac{1}{2} \mathbf{w}^\top a^o[\mathbf{k}] - \frac{\lambda}{2} \|\mathbf{w}\|_2^2 - \frac{1}{n} \sum_{i=1}^{n} \ln \sum_{y} \exp\left(\frac{1}{2} y \mathbf{w}^\top \mathbf{k}_i\right) \tag{7.2}$$

where $a^o[\mathbf{k}] = \frac{1}{n} \sum_i y_i \mathbf{k}(\mathbf{x}_i^1, \mathbf{x}_i^2)$ is what we call sufficient statistics. The main challenge is that the true labels for the observed pairs are unknown, making it difficult to apply the maximum likelihood estimation. Below, we present an approach that effectively approximates the sufficient statistics using the noisy labels.

7.2.2 Estimating Sufficient Statistics

In this section, we present an alternative approach for estimating the sufficient statistics $a^o[\mathbf{k}]$ under a different assumption from that in Chapter 6. We first present the assumption.

Assumption 2. *We assume (2.a)* $\Pr(|\mathrm{X}^1, \mathrm{X}^2|\mathrm{Y}, \widehat{\mathrm{Y}}) = \Pr(\mathrm{X}^1, \mathrm{X}^2|\mathrm{Y})$*, (2.b)* $\Pr(\mathrm{Y} = y|\widehat{\mathrm{Y}} = y) = d_y$ *and* $\Pr(\mathrm{Y} = y) = p_y$ *is given.*

Compared to **Assumption 1**, we can see that **Assumption 2** assumes that the data pair $\mathrm{X}^1, \mathrm{X}^2$ is independent of the noisy label $\widehat{\mathrm{Y}}$ given the true label Y, which is essentially equivalent to assumption (1.a). Nevertheless, we assume different knowledge about the noise, i.e., assumption (2.b), which is different from assumption (1.b). Comparing the two assumptions, assumption 1 may be advantageous if we know that how much noise is added on top the true labels, however, the conditional probabilities in assumption (2.b) may be estimated more accurately by a sampling approach, which samples a part of examples among noisily labeled pairs and query their true labels. Given the knowledge of p_y, we rewrite the expectation $\mathrm{E}^o[\mathbf{k}]$ by

$$\mathrm{E}^o[\mathbf{k}] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2,\mathrm{Y}}\left[\mathrm{Y}\mathbf{k}(\mathrm{X}^1, \mathrm{X}^2)\right] = \mathrm{E}_{\mathrm{Y}}\left[\mathrm{Y}\mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2|\mathrm{Y}}\mathbf{k}(\mathrm{X}^1, \mathrm{X}^2)\right] = p_+\mathrm{E}^c_+[\mathbf{k}] - p_-\mathrm{E}^c_-[\mathbf{k}]$$

where $\mathrm{E}^c_y[\mathbf{k}]$ is defined in Table 1. Estimating $\mathrm{E}^o[\mathbf{k}]$ is therefore reduced to estimating $\mathrm{E}^c_y[\mathbf{k}]$. Given the independence assumption, we have

$$\begin{aligned}\mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2|\widehat{\mathrm{Y}}=y}[\mathbf{k}(\mathrm{X}^1, \mathrm{X}^2)] =& \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2|\mathrm{Y}=y}[\mathbf{k}(\mathrm{X}^1, \mathrm{X}^2)] \Pr(\mathrm{Y} = y|\widehat{\mathrm{Y}} = y) + \\ & \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2|\mathrm{Y}=\bar{y}}[\mathbf{k}(\mathrm{X}^1, \mathrm{X}^2)] \Pr(\mathrm{Y} = \bar{y}|\widehat{\mathrm{Y}} = y)\end{aligned}$$

Writing in the matrix form, we have

$$\begin{pmatrix} \widehat{\mathrm{E}}^c_+[\mathbf{k}]^\top \\ \widehat{\mathrm{E}}^c_-[\mathbf{k}]^\top \end{pmatrix} = \mathbf{B}\begin{pmatrix} \mathrm{E}^c_+[\mathbf{k}]^\top \\ \mathrm{E}^c_-[\mathbf{k}]^\top \end{pmatrix} \tag{7.3}$$

where $\widehat{\mathrm{E}}^c_y[\mathbf{k}]$ is defined in Table 6.1, and $\mathbf{B}$ is defined as $\mathbf{B} = \begin{pmatrix} d_+ & 1 - d_+ \\ 1 - d_- & d_- \end{pmatrix}$. Equation (7.3) allows us to estimate $\mathrm{E}^c_y[\mathbf{k}]$ (and therefore $\mathrm{E}^o[\mathbf{k}]$) from $\widehat{\mathrm{E}}^c_y[\mathbf{k}]$, a quantity that can be computed from the noisy labels. In particular, we approximate $\widehat{\mathrm{E}}^c_y[\mathbf{k}]$ by its sample average $\widehat{a}^c_y[\mathbf{k}]$ which is computed by

$$\widehat{a}^c_y[\mathbf{k}] = \frac{\sum_i \delta(\widehat{y}_i, y)\mathbf{k}(\mathbf{x}^1_i, \mathbf{x}^2_i)}{\sum_i \delta(\widehat{y}_i, y)}$$

By replacing $\widehat{\mathrm{E}}_y^c[\mathbf{k}]$ with $\widehat{a}_y^c[\mathbf{k}]$ in (7.3), we obtain the estimation for $\mathrm{E}_y^c[\mathbf{k}]$, denoted by $b_y^c[\mathbf{k}]$, by solving the following least square problem

$$\min_{b_y^c[\mathbf{k}]} \quad \left\| \mathbf{B} \begin{pmatrix} b_+^c[\mathbf{k}]^\top \\ b_-^c[\mathbf{k}]^\top \end{pmatrix} - \widehat{\mathbf{A}} \right\|_F \tag{7.4}$$

where $\widehat{\mathbf{A}} = \begin{pmatrix} \widehat{a}_+^c[\mathbf{k}]^\top \\ \widehat{a}_-^c[\mathbf{k}]^\top \end{pmatrix}$. To obtain a more robust estimation of $\mathrm{E}_y^c[\mathbf{k}]$, we note that $\sum_y p_y \mathrm{E}_y^c[\mathbf{k}] = \mathrm{E}_{\mathrm{X}}[\mathbf{k}(\mathrm{X})] = \mathrm{E}[\mathbf{k}]$, and therefore add the following constraint for the estimator $b_y^c[\mathbf{k}]$

$$\sum_y p_y b_y^c[\mathbf{k}] = \frac{1}{n} \sum_{i=1}^n \mathbf{k}(\mathbf{x}_i^1, \mathbf{x}_i^2) = a[\mathbf{k}] \tag{7.5}$$

Combining Equation (7.4) and (7.5), we have the following constrained least square problem for $b_y^c[\mathbf{k}]$, an estimator of $\mathrm{E}_y^c[\mathbf{k}]$

$$\min_{b_y^c[\mathbf{k}]} \quad \left\| \mathbf{B} \begin{pmatrix} b_+^c[\mathbf{k}]^\top \\ b_-^c[\mathbf{k}]^\top \end{pmatrix} - \widehat{\mathbf{A}} \right\|_F, \qquad s.t. \quad \sum_y p_y b_y^c[\mathbf{k}] = a[\mathbf{k}] \tag{7.6}$$

It can be shown that the solution for $b_y^c[\mathbf{k}]$ is given by

$$\begin{pmatrix} b_+^c[\mathbf{k}]^\top \\ b_-^c[\mathbf{k}]^\top \end{pmatrix} = (\mathbf{B}^\top \mathbf{B})^{-1} \left(\frac{\mathbf{p} a[\mathbf{k}]^\top}{\mathbf{p}^\top (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{p}} + \left[\mathbf{I} - \frac{\mathbf{p}\mathbf{p}^\top (\mathbf{B}^\top \mathbf{B})^{-1}}{\mathbf{p}^\top (\mathbf{B}^\top \mathbf{B})^{-1} \mathbf{p}} \right] \mathbf{B}^\top \widehat{\mathbf{A}} \right) \tag{7.7}$$

where $\mathbf{p} = (p_+, p_-)^\top$. Note that the solution in (7.7) requires $\mathbf{B}$ to be non-singular, implying $d_+ + d_- \neq 1$. Given $\mathrm{E}_y^c[\mathbf{k}]$ is estimated by $b_y^c[\mathbf{k}]$, we have $\mathrm{E}^o[\mathbf{k}]$ estimated as follows

$$\mathrm{E}^o[\mathbf{k}] \approx b^o[\mathbf{k}] = p_+ b_+^c[\mathbf{k}] - p_- b_-^c[\mathbf{k}]$$

leading to the following maximizing the approximate log-likelihood of the true labels for the observed pairs

$$\max_{\mathbf{w} \in \mathbb{R}_+^m} \quad \frac{1}{2} \mathbf{w}^\top b^o[\mathbf{k}] - \frac{\lambda}{2} \|\mathbf{w}\|_2^2 - \frac{1}{n} \sum_i \ln \sum_y \exp\left(\frac{1}{2} y \mathbf{w}^\top \mathbf{k}_i \right) \tag{7.8}$$

7.2.3 Convergence Analysis

In this section, we present the convergence analysis showing that the combination weights learned by the proposed approach converge to the optimal ones learned from perfectly labeled pairs. All the proofs for the lemmas and theorems presented in this section are included in supplementary material.

Comparing (7.2) with (7.8), we see that the only difference between the two optimization problems is that $a^o[\mathbf{k}]$ in (7.2) is replaced with $b^o[\mathbf{k}]$ in (7.8). The following lemma showing the bounds for $\mathbf{w}$ when replacing $a^o[\mathbf{k}]$ with $b^o[\mathbf{k}]$.

Lemma 14 (Lemma 6[78])**.** *Let $\mathbf{w}_a^*$ be the solution to (7.2) with $a^o[\mathbf{k}]$, and $\mathbf{w}_b^*$ be the solution to (7.8) with $b^o[\mathbf{k}]$. We have*

$$\|\mathbf{w}_a^* - \mathbf{w}_b^*\|_2 \leq \frac{1}{\lambda} \|a^o[\mathbf{k}] - b^o[\mathbf{k}]\|_2$$

Next, we try to bound the difference between $a^o[\mathbf{k}]$ and $b^o[\mathbf{k}]$. Note that

$$\|a^o[\mathbf{k}] - b^o[\mathbf{k}]\|_2 \leq \|a^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2 + \|b^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2 \tag{7.9}$$

The following two lemmas allow us to bound the two terms in (7.9), respectively.

Lemma 15. *With the bounded kernel function $k_j(\mathbf{x}^1, \mathbf{x}^2)$, i.e., $|k_j(\mathbf{x}^1, \mathbf{x}^2)| \leq R, j = 1, \ldots, m$, the following inequality holds with probability at least $1 - \delta$ for any $\delta > 0$*

$$\|a^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2 \leq \sqrt{\frac{2mR^2}{n} \ln\left(\frac{2m}{\delta}\right)}$$

Lemma 16. *With bounded kernel function $k_j(\mathbf{x}^1, \mathbf{x}^2)$, i.e. $|k_j(\mathbf{x}^1, \mathbf{x}^2)| \leq R, j = 1, \ldots, m$, and significant large number n_+ of pairs labeled as positive and large number n_- of pairs labeled as negative, i.e., there exists some positive constant $\rho > 0$ such that $\min(n_+/n, n_-/n) \geq \rho$. Under the independence assumption, the following inequality holds with probability at least $1 - \delta$ for any $\delta > 0$*

$$\|b^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2 \leq \sqrt{C\frac{2mR^2}{n} \ln\left(\frac{6m}{\delta}\right)}$$

with C defined by

$$C = \frac{4\kappa^2}{\|\mathbf{p}\|_2^2} + \frac{32(1+\kappa^3)}{\rho^2\sigma_{\min}} \leq \frac{4\hat{d}^4}{\|\mathbf{p}\|_2^2 d^4} + \frac{32(d^6\hat{d} + \hat{d}^7)}{\rho^2 d^8}$$

where $\sigma_{\max}, \sigma_{\min}$ are the maximum and minimum eigenvalue of $\mathbf{B}^\top\mathbf{B}$, $\kappa = \sigma_{\max}/\sigma_{\min}$, $d = d_+ + d_- - 1$, and $\hat{d} = 1 + |d_+ - d_-|$.

Combining the results in Lemma 1, Lemma 2 and Lemma 3, we have the following theorem.

Theorem 17. *With the same conditions as in Lemma 3, the following inequality holds with probability at least* $1-\delta$ *for any* $\delta > 0$

$$\left\|\mathbf{w}_a^* - \mathbf{w}_b^*\right\|_2 \leq \frac{1}{\lambda}\sqrt{\frac{2mR^2}{n}}\left(\sqrt{C\ln\frac{12m}{\delta}} + \sqrt{\ln\frac{4m}{\delta}}\right)$$

Theorem 4 indicates $\mathbf{w}_b^*$, the optimal solution to problem (7.8) converges to $\mathbf{w}_a^*$, the optimal solution to problem (7.2), as the number of the pairs n approaches infinity. It is interesting to observe that the bound is proportional to C, which is inversely proportional to $|d|$ that is related to noisy level. Hence, the larger the $|d|$, the better the approximate solution will be. This will be further validated by our experiments.

Our next result is regarding the difference between the estimated solution $\mathbf{w}_b^*$ and the optimal solution $\mathbf{w}^*$ obtained by solving (7.1) with an infinite number of perfectly labeled pairs.

Theorem 18. *With the same conditions as in Lemma 3, the following inequality holds with probability at least* $1-\delta$ *for any* $\delta > 0$

$$\left\|\mathbf{w}_b^* - \mathbf{w}^*\right\|_2 \leq \frac{1}{\lambda}\sqrt{\frac{2mR^2}{n}}\left(\sqrt{C\ln\frac{24m}{\delta}} + \sqrt{\ln\frac{8m}{\delta}}\right) + \sqrt{\frac{4\rho\sqrt{m}R}{\lambda\sqrt{n}}\left(1 + \sqrt{\frac{1}{2}\ln\frac{4}{\delta}}\right)}$$

Finally, we note that our analysis relies on accurate estimate of $\mathbf{p}, \mathbf{B}$. In supplementary material, we provide the error bound with inaccurate estimate of $\mathbf{p}, \mathbf{B}$.

7.3 Experiments

In this section, we validate the proposed approach by clustering on linked documents using the combined kernels learned from noisy pairwise constraints. We conduct two sets of experiments: in the first set of experiments, we synthesize the noisy pairwise constraints by random sampling, and in the second setup, we derive the pairwise constraints from the links among documents. We use the same cora and citeseer data sets. The baselines for comparison are described as follows.

Data sets Two paper citation data sets, **Cora** and **Citeseer**, processed by Lise Getoor's research group, are used in our study. Each paper in the data sets is described by a binary vector indicating the presence and absence of corresponding words, and is assigned to one of the given classes. Besides the attributes and class assignments of papers, the citations between papers are also available in these two datasets. The statistics of the two data sets are summarized in Table 7.1. Following [47], we normalize the attributes of each document by first dividing the sum of the attributes and then taking the square root.

Baselines We compare the proposed algorithm to the following two algorithms that learn kernel matrices from pairwise constraints: (1) **NPK**, a state-of-the-art non-parametric kernel learning algorithm from pairwise constraints [42], and (2) **SKL**, a spectral kernel learning algorithm [44]. We did not compare to the kernel learning algorithms [50, 52] from pairwise constraints because they are outperformed by NPK according to [42], and to algorithm in [57] because it requires solving a SDP problem, which does not scale well for large data sets. Besides kernel learning algorithms, we also compare the proposed algorithm to three representative methods for distance metric learning from pairwise constraints: (1) **GDM**, a global distance metric learning algorithm [102], (2) **DCA**, a discriminative component analysis algorithm [43], and (3) **ITML**, a information-theoretic based metric learning algorithm [21]. We did not compare to the other metric learning algorithms because they either are not as effective as these three metric learning algorithms (e.g. [31]) or require labeled data for training instead of pairwise constraints (e.g., [100]). The learned metric/kernel matrix is used by a spectral clustering algorithm [84] to cluster documents. Since these learning algorithms do not address noisy pairwise constraints, the comparison will allow us to verify if the proposed algorithm is robust to the noise in pairwise constraints. The last two baselines are constrained clustering algorithms: **MPCK** [8], a state-of-the-art algorithm for constrained clustering, and **LCVQE** [75], a recently developed k-means algorithm with noisy constraints. To improve the robustness of **MPCK** to noisy constraints, we follow [61]

Table 7.1. Statistics of Data sets

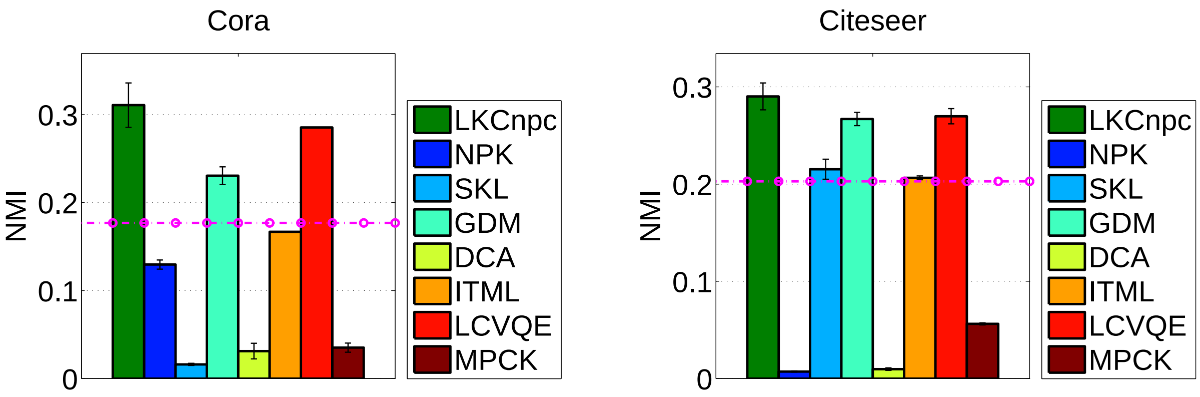
data	#papers	#attr	#citations	#classes
Cora	2708	1433	5429	7
Citeseer	3312	3703	4591	6

by weighting the noisy pairwise constraints by probability d_y. Since **MPCK** and **LCVQE** are claimed to be robust to noisy constraints, the comparison will allow us to see if the proposed approach is effective for noisy pairwise constraints. Finally, we include the baseline that combines kernels with equal weights, referred to as **LK**.

For the proposed approach, each candidate kernel k_j is the linear kernel on the j^{th} attribute. Given the learned combination of kernel matrices, the same spectral clustering algorithm will be used to cluster documents. In all the experiments, we set the regularization parameter λ as $\lambda = 0.01/n$. We refer to the proposed approach as **LKCnpc**. To evaluate the clustering results, we compute the *normalized mutual information* [107] by comparing the cluster assignments predicted by the clustering algorithms to the class assignments given in the data set.

7.3.1 Experiments with Real Noise

In this subsection, we show experimental results on noisy pairwise constraints derived from citation information. In particular, a positive pairwise constraint is created if two papers are linked by a citation, and a negative pairwise constraint is created when two papers do not cite each other. Since the number of unlinked paper pairs is much larger than that of linked pairs, for computational efficiency, we randomly sample the same number of unlinked paper pairs as that of linked paper pairs. As a result, we have on average about 20%-25% pairwise constraints being incorrect. To obtain p_y, d_y we randomly sample 1% of document pairs and label them according to their class assignments, and compute p_y, d_y from the noisy labels and the correct labels for these sampled pairs. For fair comparison, the correct labels of these sampled pairs are used by the baselines.



(a) Cora

(b) Citeseer

Figure 7.1. Comparison of clustering performance on real noise.

Table 7.2. Clustering performance on unseen data

data	observed data	unseen data
Cora	0.3326±0.013	0.3028±0.012
Citeseer	0.2822±0.012	0.2502±0.020

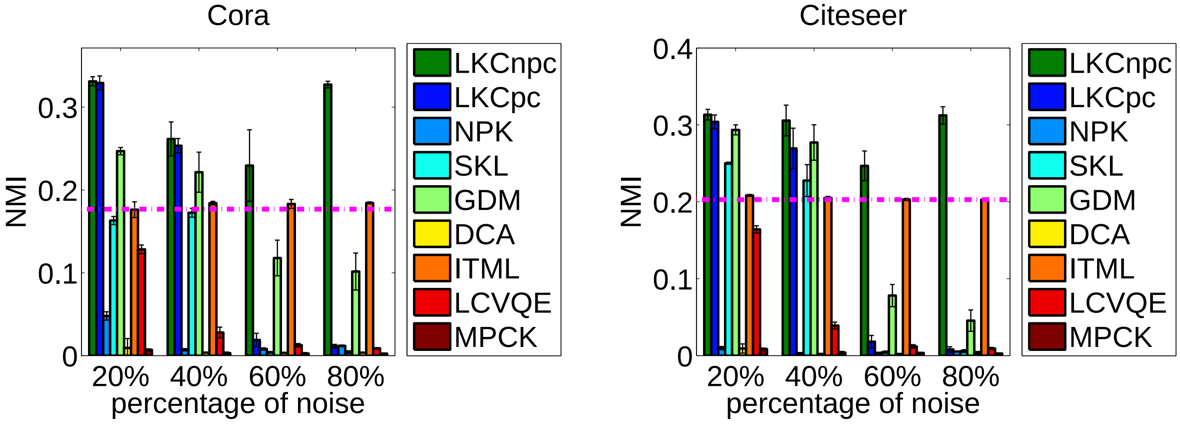
Figure 7.1(a) and 7.1(b) show the clustering results on the two data sets for the proposed approach and baseline methods. We observe that most baseline methods, except for **GDM, LCVQE**, are unable to outperform the reference method **LK**. In particular, we observe that **MPCK**, a state-of-the-art constrained clustering algorithm, performs significantly worse than **LK**, indicating the importance of addressing noisy pairwise constraints in constrained clustering. Overall, we observe that the proposed approach outperforms all the baseline methods on both data sets. We thus conclude that the proposed approach is effective for learning from noisy pairwise constraints.

Finally, we briefly show the clustering results applied on the unseen data with learned combinations of multiple kernels from the observed data. We randomly choose 80% papers as observed data and derive noisy pairwise constraints for these observed data from citations, and the remaining 20% papers as unseen data for testing the learned model. The NMI averaged over 5 trials for the learned model on the observed data and unseen data for the two data sets is shown in Table 7.2. We can see that the combination weights learned from observed data also works well on unseen data.

7.3.2 Experiments with Synthetic Noise

In this subsection, we show experimental results using synthesized noisy pairwise constraints. We first examine the robustness of the proposed approach to the noisy pairwise constraints by varying the level of noise. In particular, for each data set, we randomly sample $10,000$ pairs in the same class and $10,000$ pairs in different classes. To obtain the noisy label $\widehat{y}_i$ for each pair, we randomly flip the correct label y_i with a probability of $p\%$, where $p\%$ specifies the noise level. These $20,000$ document pairs together with their noisy labels are used to train both the proposed approach and the eight baseline methods for document clustering. The values of p_y and d_y are computed from the correct labels and noisy labels. We choose four noise levels, $p\% = 20\%, 40\%, 60\%, 80\%$ in our study. Figure 7.2(a) and 7.2(b) compare the clustering performance of the proposed approach to that of the baseline methods on the two data sets. The results are averaged over five independent experiments. For the convenience of comparison, we also include the clustering result of the proposed approach but without modeling the noise in the pairwise constraints, i.e. simply using $\widehat{a}^o[\mathbf{k}]$ in place of $a^o[\mathbf{k}]$, as referred to **LKCpc**. The comparison to **LKCpc** allows us to see how the noise in pairwise constraints affects the clustering results.

It is clear that the proposed approach **LKCnpc** is more resilient to the noisy pairwise constraints than the baseline methods. In particular, we observe that even the noise level is relatively low(20%), most of the baseline methods, except for **LKCpc, GDM**, are unable to outperform the reference method **LK**. As the noise level increases, except for **ITML**, the performance of all the other baseline methods degrade significantly and become much worse than that of **LK** when the noise level exceeds 50%. For all the cases, the proposed approach yields the best performance among all the methods in comparison. More interestingly, on both data sets, we observe that the clustering performance of the proposed approach increases when the noise level goes beyond or below 50%. This result seems surprising at the first glance, however, by noting that our approach relies on the knowledge of $d_y = \Pr(\mathrm{Y} = y|\widehat{\mathrm{Y}} = y)$ related to noise level, when the noise level is above 50%, the simple approach by flipping labels of pairwise constraints to their opposite can obtain pairwise constraints with smaller errors, leading to better clustering performance, and so of course does the proposed approach. Also, this result is consistent with our theoretical analysis in Theorem 4, because the difference between the model learned from noisily labeled pairs and the model learned from perfectly labeled pairs decreases as $|d| = |d_1 + d_0 - 1|$ increases. The corresponding $|d|$ for the four noise levels $p\% = 20\%, 40\%, 60\%, 80\%$ are $0.6, 0.2, 0.2, 0.6$.



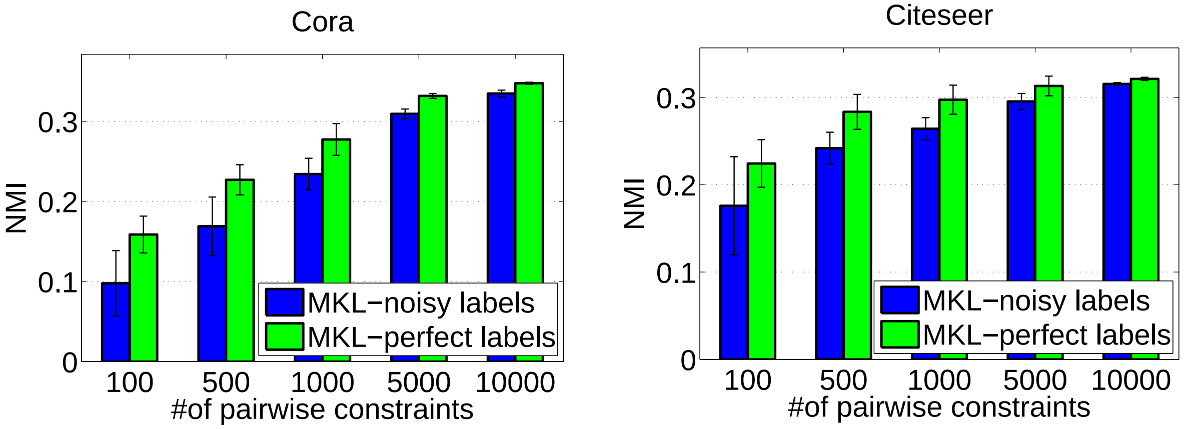
(a) Cora

(b) Citeseer

Figure 7.2. Comparison of clustering performance with different noise levels.

Second, we show the proposed approach is not sensitive to the estimated values of p_y and d_y. With 20,000 synthetic pairwise constraints, and $p\% = 20\%$ noise, we randomly perturb p_y and d_y by either adding or subtracting 10 percentage from their "true" values computed from the ground truth. We found the clustering performance is almost the same, as shown in Table 7.3, where p_y, d_y refer to their "true" values, i.e. $p_y = \sum_i \delta(y_i, y)/n, d_y = \sum_i \delta(y_i, y)\delta(\widehat{y}_i, y)/\sum_i \delta(\widehat{y}_i, y)$, and $\hat{p}_y, \hat{d}_y$ refer to randomly perturbed values $\hat{p}_+ = (1 \pm 10\%)p_+, \hat{d}_y = (1 \pm 10\%)d_y$, and the results for perturbed values are averaged over 10 random trials.

Third, we show the experimental results to verify the convergence behavior in the number



(a) Cora

(b) Citeseer

Figure 7.3. Convergence behavior of the proposed approach

Table 7.3. Clustering performance with perturbed (p_y, d_y)

data	(p_y, d_y)	$(\hat{p}_y, \hat{d}_y)$
Cora	0.3338	0.3279±0.006
Citeseer	0.3120	0.3011±0.007

of noisily labeled pairs n as presented in the paper. We sample the equal number of pairs in the same class and pairs in different classes. The noisy label for these pairs are obtained by using the same random flipping process. We fix the noisy level to $p\% = 20\%$. We compare the clustering performance of the proposed approach with noisy labels to the same approach but with perfect labels by increasing the total number of pairwise constraints from $200, 1000, 2000, 10,000$ to $20,000$. The results in Figure 7.3(a) and Figure 7.3(b) show that the difference decreases as the number of constraints increases. We also noticed that the iterative method (e.g., Newton method) for solving the related optimization problem converges very quickly in one hundred iterations on the two data sets with 20,000 paris.

Finally, we mention that the two approaches proposed in this Chapter and last Chapter yield similar empirical results.

7.4 Conclusions

In this chapter, we have presented an alternative approach for estimating the sufficient statistics in generalized maximum entropy model under different assumptions from Chapter 6. Our theoretical analysis shows that the model trained from the noisy link information converges to the model trained from the perfect link information. Extensive experimental results verify the efficacy of the proposed approach.

CHAPTER 8

Conclusions and Future Work

In previous chapters, we have presented a link-based model for community detection, a discriminative approach for combining link and content information for community detection for networked data, and a probabilistic link prediction model for clustering the noisy M&C linked data. We summarize the contributions of the thesis as follows.

- In Chapter 4, we present a probabilistic link model for detecting communities for directed networks. We introduce popularity and productivity to model the differences of nodes in receiving links and the differences of nodes in producing links, respectively. These two factors can explain the noisy connections in terms of community detection, because the connections between nodes may be not due to their common or similar community, but rather because of their high popularities tending to receive links or high productivities tending to produce links. These two factors can also explain the preferential attachment phenomenon in the real world, i.e., the power law degree distribution. Using these two factors, together with community memberships, we define a generative model to generate the links. The proposed link model is an unified framework which can include several previous models in degenerated cases, and also can derive us new link models. It is the first time that the power law degree distribution is modeled for community detection.

- In Chapter 5, we present a discriminative approach for combining link and content information for community detection. Different from previous approaches that usually put together a generative link model and a generative content model and therefore are vulnerable to irrelevant or noisy attributes, the proposed approach uses a discriminative model on the content to fit the community memberships, which thus has discriminative power to identify relevant attributes from irrelevant ones. The proposed approach for combining our link model and our content model yield significantly better empirical performance for community detection. It is the first discriminative approach for combining link and content for community detection.

- In Chapter 6 and Chapter 7, we present approaches for clustering noisy must-and-cannot linked data. To handle the noisy M&C links, we formulate the problem into learning from noisily labeled data. We propose a generalized maximum entropy model to learn the conditional model that given a pair of data how likely they are connected by a must link (i.e., how likely they belong to the same cluster). The critical problem of learning the conditional model is to compute the sufficient statistics that depend on the true labels which are unknown. We propose two different approaches under different assumptions to estimate the sufficient statistics and we prove the convergence results , i.e., the model learned from the noisy labels converges to the model learned from the true labels under appropriate assumptions about the data and the noise. It is the first work that proves the convergence for learning from noisily labeled data.

The studies conducted during the course of this dissertation point to several directions for future research:

- **Algorithms for Detecting Communities and Their Evolutions in Dynamic Networks** Since networks are dynamic in the real world, e.g., nodes could leave the network and new nodes could join in the network, nodes could change their communities, and communities could disappear and new communities would emerge, it is important to model the dynamic behavior of communities in networks. For future work, we can extend the proposed link models or previous link models to a dynamic version in order to model the dynamic changes in networks. The key challenges are (i) how to model the evolution of communities sequentially, and (ii) how to handle the deletion, insertion of nodes.

- **Algorithms and Theory for Learning from Noisily Labeled Data** The problem of noisy M&C linked data was abstracted into the problem of learning from noisily labeled data in Chapter 6&7, which is an important problem in machine learning since the labels could be noisy due to human error or uncertainty in labeling. For future work along the line, we can think about several directions: (1) how to relax the independence assumption made by the approaches in Chapter 6&7, while still maintain some theoretical guarantee about the learned model; (2) how to extend SVM to handle the noise labels with an efficient optimization algorithm to learn the parameter; (3) how to extend the problem to an online setup, where at each trial an noisy label is given to the learner together with the data, after making prediction on the data, the learner decides to make a query or not for the true label of the data, for which the learner usually needs to pay a cost. The goal of the learner is to minimize the errors he makes on the received examples under certain constraint imposed on the cost for querying the true labels.

- **Applications** The proposed link models for community detection provide a solution for link prediction. The goal of link prediction is to predict whether two entities could link to each other (e.g., whether a user could be followed by other users or whether a product could be purchased by customers). The proposed approach can predict the links based on the communities of the entities and their popularities, productivities and attributes. It would be interesting to compare the community-based link models with traditional approaches for link prediction.

  The proposed approach for learning from noisily labeled data can be applied to semi-supervised crowd clustering, where pairwise constraints are usually collected to facilitate the clustering and subject to errors because of biased or adversarial human annotators. The technique of estimating the sufficient statistics from noisy labels can be also applied to estimating the sufficient statistics for a target class from that for auxiliary classes based on the relation between the target class and the auxiliary classes parameterized by the conditional probabilities that given a data point belongs to the target class how likely it also belongs to an auxiliary class. Using this technique, we can learn a model for predicting the target class without any training examples that are labeled by the target class, which could be useful when the target class label is difficult or expensive to collect.

APPENDICES

APPENDIX A

Proofs

Proof to Theorem 7

Proof. Under the i.i.d. assumption, it is straightforward to show that

$$\mathrm{E}[a_y^p[\kappa_j]] = \mathrm{E}[a_y^\delta[\kappa_j]] = \mathrm{E}_y^\delta[\kappa_j]$$

Following the McDiarmid's inequality, for any $\epsilon > 0$, we have

$$\Pr\left(\left|a_y^p[\kappa_j] - \mathrm{E}_y^\delta[\kappa_j]\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$$

$$\Pr\left(\left|a_y^\delta[\kappa_j] - \mathrm{E}_y^\delta[\kappa_j]\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$$

Using the following inequality and the union bound,

$$\left|a_y^\delta[\kappa_j] - a_y^p[\kappa_j]\right| \leq \left|a_y^\delta[\kappa_j] - \mathrm{E}_y^\delta[\kappa_j]\right| + \left|a_y^p[\kappa_j] - \mathrm{E}_y^\delta[\kappa_j]\right|$$

we can complete the theorem using union bound. □

Proof to Theorem 8

Proof. Using the assumption (1.a) and (1.b), we have

$$\widehat{\mathrm{E}}_y^\delta[\kappa_j] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2}\mathrm{E}_{\widehat{\mathrm{Y}}|\mathrm{X}^1,\mathrm{X}^2}[\delta(\widehat{\mathrm{Y}}, y)\kappa_j(X^1, X^2)] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2}[\Pr(\widehat{\mathrm{Y}} = y|\mathrm{X}^1, \mathrm{X}^2)\kappa_j(\mathrm{X}^1, \mathrm{X}^2)]$$

$$= \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2}\left[c_y \Pr(\mathrm{Y} = y|\mathrm{X}^1, \mathrm{X}^2)\kappa_j(\mathrm{X}^1, \mathrm{X}^2)\right] + \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2}\left[(1 - c_{\bar{y}}) \Pr(\mathrm{Y} = \bar{y}|\mathrm{X}^1, \mathrm{X}^2)\kappa_j(\mathrm{X}^1, \mathrm{X}^2)\right]$$

$$= c_y \mathrm{E}_y^\delta[\kappa_j] + (1 - c_{\bar{y}})\mathrm{E}_{\bar{y}}^\delta[\kappa_j] = (c_y + c_{\bar{y}} - 1)\mathrm{E}_y^\delta[\kappa_j] + (1 - c_{\bar{y}})\mathrm{E}[\kappa_j(\mathrm{X}^1, \mathrm{X}^2)]$$

where we use the fact $\mathrm{E}_y^\delta[\kappa_j] + \mathrm{E}_{\bar{y}}^\delta[\kappa_j] = \mathrm{E}[\kappa_j(\mathrm{X}^1, \mathrm{X}^2)]$. Let us define

$$\widehat{c}_y^\delta[\kappa_j] = (c_y + c_{\bar{y}} - 1)a_y^\delta[\kappa_j] + (1 - c_{\bar{y}})\frac{1}{n}\sum_i \kappa_j(\mathbf{x}_i^1, \mathbf{x}_i^2)$$

Under the i.i.d. assumption, it is straightforward to show that

$$\mathrm{E}[\widehat{a}_y^\delta[\kappa_j]] = \mathrm{E}[\widehat{c}_y^\delta[\kappa_j]] = \widehat{\mathrm{E}}_y^\delta[\kappa_j]$$

Following the McDiarmid's inequality, for any $\epsilon > 0$, we have

$$\begin{aligned}
\Pr\left(\left|\widehat{a}_y^\delta[\kappa_j] - \widehat{\mathrm{E}}_y^\delta[\kappa_j]\right| \geq \epsilon\right) &\leq 2\exp\left(-\frac{\epsilon^2 n}{2R^2}\right) \\
\Pr\left(\left|\widehat{c}_y^\delta[\kappa_j] - \widehat{\mathrm{E}}_y^\delta[\kappa_j]\right| \geq \epsilon\right) &\leq 2\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)
\end{aligned}$$

Then we have

$$\Pr\left(\left|\widehat{c}_y^\delta[\kappa_j] - \widehat{a}_y^\delta[\kappa_j]\right| \geq \epsilon\right) \leq 4\exp\left(-\frac{\epsilon^2 n}{8R^2}\right)$$

Dividing both sides of $\left|\widehat{c}_y^\delta[\kappa_j] - \widehat{a}_y^\delta[\kappa_j]\right| \geq \epsilon$ by $|c_y + c_{\bar{y}} - 1|$, we have

$$\Pr\left(\left|a_y^\delta[\kappa_j] - \widehat{b}_y^\delta[\kappa_j]\right| \geq \frac{\epsilon}{|c_y + c_{\bar{y}} - 1|}\right) \leq 4\exp\left(-\frac{\epsilon^2 n}{8R^2}\right)$$

Replacing ϵ with $|(c_y + c_{\bar{y}} - 1)|\epsilon$, we complete the proof using the union bound. □

Proof to Theorem 11

Proof. Let

$$\begin{aligned}
L(\mathbf{w}) &= \frac{1}{n}\sum_i \ln \sum_y \exp\left(\frac{1}{2} y \mathbf{w}^\top \mathbf{k}_i\right) - \frac{1}{2}\mathbf{w}^\top \mathbf{v} + \frac{\lambda}{2}\|\mathbf{w}\|_2^2 \\
&= g(\mathbf{w}) - \frac{1}{2}\mathbf{w}^\top \mathbf{v} + \frac{\lambda}{2}\|\mathbf{w}\|_2^2
\end{aligned}$$

where $g(\mathbf{w})$ is the sum of log-exponential function of $\mathbf{w}$, which is convex in $\mathbf{w}$. Assume $\mathbf{w}^*$ is the optimal solution to minimizing $L(\mathbf{w})$, $\widehat{\mathbf{w}}^*$ is the optimal solution to minimizing $L(\mathbf{w})$ with $\mathbf{v}$ replaced with $\widehat{\mathbf{v}}$. First, we have

$$\begin{aligned}
L(\widehat{\mathbf{w}}^*) &\geq L(\mathbf{w}^*) + \nabla L(\mathbf{w}^*)^\top (\widehat{\mathbf{w}}^* - \mathbf{w}^*) + \frac{\lambda}{2}\|\widehat{\mathbf{w}}^* - \mathbf{w}^*\|_2^2 \\
&\geq L(\mathbf{w}^*) + \frac{\lambda}{2}\|\widehat{\mathbf{w}}^* - \mathbf{w}^*\|_2^2
\end{aligned}$$

where the first inequality follows that $L(\cdot)$ is a λ-strongly convex function, and the second inequality follows the optimality criterion that $\nabla L(\mathbf{w}^*)^\top(\widehat{\mathbf{w}}^* - \mathbf{w}^*) \geq 0$. Second,

$$\begin{aligned}
L(\widehat{\mathbf{w}}^*) &= g(\widehat{\mathbf{w}}^*) - \frac{1}{2}\mathbf{v}^\top\widehat{\mathbf{w}}^* + \frac{\lambda}{2}\|\widehat{\mathbf{w}}^*\|_2^2 \\
&= g(\widehat{\mathbf{w}}^*) - \frac{1}{2}\widehat{\mathbf{v}}^\top\widehat{\mathbf{w}}^* + \frac{\lambda}{2}\|\widehat{\mathbf{w}}^*\|_2^2 + \frac{1}{2}(\widehat{\mathbf{v}} - \mathbf{v})^\top\widehat{\mathbf{w}}^* \\
&\leq g(\mathbf{w}^*) - \frac{1}{2}\widehat{\mathbf{v}}^\top\mathbf{w}^* + \frac{\lambda}{2}\|\mathbf{w}^*\|_2^2 + \frac{1}{2}(\widehat{\mathbf{v}} - \mathbf{v})^\top\widehat{\mathbf{w}}^* \\
&\leq g(\mathbf{w}^*) - \frac{1}{2}\mathbf{v}^\top\mathbf{w}^* + \frac{\lambda}{2}\|\mathbf{w}^*\|_2^2 + \frac{1}{2}(\widehat{\mathbf{w}}^* - \mathbf{w}^*)^\top(\widehat{\mathbf{v}} - \mathbf{v}) \\
&\leq L(\mathbf{w}^*) + \frac{1}{2}\|\mathbf{w}^* - \widehat{\mathbf{w}}^*\|_2\|\mathbf{v} - \widehat{\mathbf{v}}\|_2
\end{aligned}$$

Coming the above two bounds for $L(\widehat{\mathbf{w}}^*)$ together, we have

$$\frac{\lambda}{2}\|\mathbf{w}^* - \widehat{\mathbf{w}}^*\|_2^2 \leq \frac{1}{2}\|\mathbf{w}^* - \widehat{\mathbf{w}}^*\|_2\|\mathbf{v} - \widehat{\mathbf{v}}\|_2$$

i.e.,

$$\|\mathbf{w}^* - \widehat{\mathbf{w}}^*\|_2 \leq \frac{1}{\lambda}\|\mathbf{v} - \widehat{\mathbf{v}}\|_2$$

Let $\mathbf{v} = a^o[\mathbf{k}]$, and $\mathbf{w}_a^*$ be the corresponding optimal solution, and $\widehat{\mathbf{v}} = b^o[\mathbf{k}]$, and $\mathbf{w}_b^*$ be the corresponding optimal solution, then we have

$$\|\mathbf{w}_a^* - \mathbf{w}_b^*\|_2 \leq \frac{1}{\lambda}\|a^o[\mathbf{k}] - b^o[\mathbf{k}]\|_2$$

□

Proof of Theorem 12

Proof. Since $1/(1+\exp(-s))$ is a lipschitz continuous function with lipschitz continuity of 1, we have

$$\begin{aligned}
&|\widehat{p}(y|\mathbf{x}^1, \mathbf{x}^2) - p(y|\mathbf{x}^1, \mathbf{x}^2)| \leq \left|(\mathbf{w}_a^* - \mathbf{w}_b^*)^\top\mathbf{k}(\mathbf{x}^1, \mathbf{x}^2)\right| \\
&\leq \sqrt{m}\|\mathbf{w}_a^* - \mathbf{w}_b^*\|_2 R \leq \frac{\sqrt{m}R}{\lambda}\|a^o[\mathbf{k}] - b^o[\mathbf{k}]\|_2 \leq \frac{\sqrt{m}R}{\lambda}\sqrt{\frac{8mR^2}{(c_+ + c_- - 1)^2 n}\ln\left(\frac{4m}{\delta}\right)} \\
&= \frac{mR^2}{c\lambda}\sqrt{\frac{8}{n}\left(\frac{4m}{\delta}\right)}
\end{aligned}$$

□

Proof of Theorem 13

Proof. We define

$$\widetilde{b}^o[\mathbf{k}] = \frac{\widehat{a}^o[\mathbf{k}]}{\widehat{c}} - \frac{1-\widehat{c}_{\bar{y}}}{\widehat{c}} a[\mathbf{k}]$$

where $\widehat{c} = \widehat{c}_+ + \widehat{c}_- - 1$. Let $\widetilde{p}(y|\mathbf{x}^1, \mathbf{x}^2)$ be the classification model learned from the noisy side information using corrupted $\widehat{c}_+$ and $\widehat{c}_-$, and $\widehat{p}(y|\mathbf{x}^1, \mathbf{x}^2)$ be the classification model learned from the noisy side information using perfect c_+ and c_-. Using the analysis in the proof of Theorem 12, we have

$$\begin{aligned}
&\left|\widetilde{p}(y|\mathbf{x}^1,\mathbf{x}^2) - \widehat{p}(y|\mathbf{x}^1,\mathbf{x}^2)\right| \le \frac{\sqrt{m}R}{\lambda}\left\|b^o[\mathbf{k}] - \widetilde{b}^o[\mathbf{k}]\right\|_2 \\
&\le \frac{\sqrt{m}R}{\lambda}\left\|\frac{\widehat{a}^o[\mathbf{k}]}{\widehat{c}} - \frac{\widehat{a}^o[\mathbf{k}]}{c} + \frac{1-c_{\bar{y}}}{c}a[\mathbf{k}] - \frac{1-\widehat{c}_{\bar{y}}}{\widehat{c}}a[\mathbf{k}]\right\|_2 \\
&\le \frac{\sqrt{m}R}{\lambda}\left(\sqrt{m}R\left|\frac{1}{c} - \frac{1}{\widehat{c}}\right| + \sqrt{m}R\left|\frac{1-c_{\bar{y}}}{c} - \frac{1-\widehat{c}_{\bar{y}}}{\widehat{c}}\right|\right) \\
&\le \frac{mR^2}{\lambda}\left(\frac{4\Delta}{\rho^2} + \frac{2c\Delta + 4\Delta}{\rho^2}\right)
\end{aligned}$$

where we use $\|\widehat{a}^o[\mathbf{k}]\|_2 \le \sqrt{m}R$, $\|a[\mathbf{k}]\|_2 \le \sqrt{m}R$, $|c - \widehat{c}| \le 2\Delta$, $\widehat{c} \ge \rho/2$. Using the fact

$$\left|\widetilde{p}(y|\mathbf{x}^1,\mathbf{x}^2) - p(y|\mathbf{x}^1,\mathbf{x}^2)\right| \le \left|\widetilde{p}(y|\mathbf{x}^1,\mathbf{x}^2) - \widehat{p}(y|\mathbf{x}^1,\mathbf{x}^2)\right| + \left|\widehat{p}(y|\mathbf{x}^1,\mathbf{x}^2) - p(y|\mathbf{x}^1,\mathbf{x}^2)\right|$$

and the result in Theorem 12, we have the theorem. □

Proof to Lemma 15

Proof. First we note that

$$\mathrm{E}^o[k_j] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2,\mathrm{Y}}[\mathrm{Y}k_j(\mathrm{X}^1,\mathrm{X}^2)]$$

Under the i.i.d. assumption, we have

$$\mathrm{E}[a^o[k_j]] = \mathrm{E}_{\mathrm{X}^1,\mathrm{X}^2,\mathrm{Y}}[\mathrm{Y}k_j(\mathrm{X}^1,\mathrm{X}^2)] = \mathrm{E}^o[k_j]$$

Following the McDiarmid's inequality, for any $\epsilon > 0$, we have

$$\Pr\left(\left|a^o[k_j] - \mathrm{E}^o[k_j]\right| \geq \epsilon\right) \leq 2\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$$

Using the union bound, we have

$$\Pr\left(\|a^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2^2 \leq m\epsilon^2\right) \geq 1 - 2m\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$$

Let $\delta = 2m\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$, i.e. $\epsilon = \sqrt{\frac{2R^2}{n}\ln\left(\frac{2m}{\delta}\right)}$, we can complete the proof. □

Proof to Lemma 16

Proof. First, under the independence assumption, we can have a similar solution for $\mathrm{E}^c_y[\mathbf{k}]$ as $b^c_y[\mathbf{k}]$ in Equation 7.7, i.e.

$$\begin{pmatrix} \mathrm{E}^c_+[\mathbf{k}]^\top \\ \mathrm{E}^c_-[\mathbf{k}]^\top \end{pmatrix} = (\mathbf{B}^\top\mathbf{B})^{-1}\left(\frac{\mathbf{p}\mathrm{E}[\mathbf{k}]^\top}{\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{p}} + \left[\mathbf{I} - \frac{\mathbf{p}\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}}{\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{p}}\right]\mathbf{B}^\top\bar{\mathbf{A}}\right)$$

where $\bar{\mathbf{A}} = \begin{pmatrix} \widehat{\mathrm{E}}^c_+[\mathbf{k}]^\top \\ \widehat{\mathrm{E}}^c_-[\mathbf{k}]^\top \end{pmatrix}$. To bound $\|b^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2$, we have

$$\begin{aligned}
&\|b^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2^2 \leq 2\left\|\begin{pmatrix} b^c_+[\mathbf{k}]^\top \\ b^c_-[\mathbf{k}]^\top \end{pmatrix} - \begin{pmatrix} \mathrm{E}^c_+[\mathbf{k}]^\top \\ \mathrm{E}^c_-[\mathbf{k}]^\top \end{pmatrix}\right\|_F^2 \\
&\leq 4\left\|(\mathbf{B}^\top\mathbf{B})^{-1}\frac{\mathbf{p}(\mathrm{E}[\mathbf{k}] - a[\mathbf{k}])^\top}{\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{p}}\right\|_F^2 + 4\left\|(\mathbf{B}^\top\mathbf{B})^{-1}\left[\mathbf{I} - \frac{\mathbf{p}\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}}{\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{p}}\right]\mathbf{B}^\top(\bar{\mathbf{A}} - \widehat{\mathbf{A}})\right\|_F^2 \\
&\leq 4\frac{\sigma_{\min}^{-2}}{\sigma_{\max}^{-2}\|\mathbf{p}\|_2^2}\|\mathrm{E}[\mathbf{k}] - a[\mathbf{k}]\|_2^2 + 8\|(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{B}^\top\|_F^2\|\bar{\mathbf{A}} - \widehat{\mathbf{A}}\|_F^2 \\
&+ 8\left\|\frac{(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{p}\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}}{\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{p}}\mathbf{B}^\top\right\|_F^2\|\bar{\mathbf{A}} - \widehat{\mathbf{A}}\|_F^2 \\
&\leq 4\frac{\kappa^2}{\|\mathbf{p}\|_2^2}\|\mathrm{E}[\mathbf{k}] - a[\mathbf{k}]\|_2^2 + 16(\sigma_{\min}^{-1} + \frac{\sigma_{\min}^{-4}}{\sigma_{\max}^{-2}}\sigma_{\max})\|\bar{\mathbf{A}} - \widehat{\mathbf{A}}\|_F^2 \\
&\leq \frac{4\kappa^2}{\|\mathbf{p}\|_2^2}\|\mathrm{E}[\mathbf{k}] - a[\mathbf{k}]\|_2^2 + \frac{16(1+\kappa^3)}{\sigma_{\min}}\|\bar{\mathbf{A}} - \widehat{\mathbf{A}}\|_F^2
\end{aligned}$$

where we use the fact $\|\mathbf{p}\|_2^2\sigma_{\max}^{-1} \le \|\mathbf{p}^\top(\mathbf{B}^\top\mathbf{B})^{-1}\mathbf{p}\| \le \|\mathbf{p}\|_2^2\sigma_{\min}^{-1}$, $\|\mathbf{B}\|_F^2 \le 2\sigma_{\max}$ and $\|\mathbf{B}^{-1}\|_F^2 \le 2\sigma_{\min}^{-1}$ Next, we show bound for $\|\mathrm{E}[\mathbf{k}]-a[\mathbf{k}]\|_2^2$ and $\|\bar{\mathbf{A}}-\widehat{\mathbf{A}}\|_F^2$. For $\|\mathrm{E}[\mathbf{k}]-a[\mathbf{k}]\|_2^2$, similar to Lemma 15, the following inequality holds for any $\epsilon > 0$,

$$\Pr(\|\mathrm{E}[\mathbf{k}] - a[\mathbf{k}]\|_2^2 \le m\epsilon^2) \ge 1 - 2m\exp\left(\frac{\epsilon^2 n}{2R^2}\right)$$

For $\|\bar{\mathbf{A}} - \widehat{\mathbf{A}}\|_F^2$, we need first bound each element in the matrix, i.e. $|\widehat{\mathrm{E}}_y^c[k_j] - \widehat{a}_y^c[k_j]|$, it is easy to show that

$$\Pr(|\widehat{\mathrm{E}}_y^c[k_j] - \widehat{a}_y^c[k_j]| \le \frac{\epsilon n}{n_y}) \ge 1 - 2\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$$

Then with the union bound,

$$\Pr\left(\|\bar{\mathbf{A}} - \widehat{\mathbf{A}}\|_F^2 \le \frac{2}{\rho^2}m\epsilon^2\right) \ge 1 - 4m\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$$

where we use $\min(n_+/n, n_-/n) \ge \rho$. Then we have

$$\Pr\left(\|b^o[\mathbf{k}] - \mathrm{E}^o[\mathbf{k}]\|_2^2 \le \frac{4\kappa^2}{\|\mathbf{p}\|_2^2}m\epsilon^2 + \frac{32(1+\kappa^3)}{\rho^2\sigma_{\min}}m\epsilon^2\right) \ge 1 - 6m\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$$

Let $\delta = 6m\exp\left(-\frac{\epsilon^2 n}{2R^2}\right)$, i.e. $\epsilon = \sqrt{\frac{2R^2}{n}\ln\left(\frac{6m}{\delta}\right)}$, and note that $\sigma_{\max}\sigma_{\min} = \det(\mathbf{B})^2 = d^2$, $\sigma_{\max} \le \|\mathbf{B}^\top\mathbf{B}\|_1 = \hat{d}$, we have the result in Lemma 3. □

Proof to Theorem 18

Proof. We first show $\|\mathbf{w}_a^* - \mathbf{w}^*\|_2^2$ can be well bounded with a large probability. By combining this bound with Theorem 17, we have the result in Theorem 18. In the following derivation, we use $\|\mathbf{k}(\mathrm{X}^1, \mathrm{X}^2)\|_2 \le \sqrt{m}R$. We denote by

$$\widehat{\mathcal{L}}(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \frac{1}{n}\sum_{i=1}^{n}\ln(1 + \exp(-y_i\mathbf{w}^\top\mathbf{k}_i)$$

$$\mathcal{L}(\mathbf{w}) = \frac{\lambda}{2}\|\mathbf{w}\|_2^2 + \mathrm{E}[\ln(1 + \exp(-\mathrm{Y}\mathbf{w}^\top\mathbf{k}(\mathrm{X}^1, \mathrm{X}^2)))]$$

We first bound $\mathcal{L}(\mathbf{w}_a^*)$ as follows

$$\mathcal{L}(\mathbf{w}_a^*) \leq \widehat{\mathcal{L}}(\mathbf{w}_a^*) + \max_{\|\mathbf{w}\|_2 \leq \rho} \mathcal{L}(\mathbf{w}) - \widehat{\mathcal{L}}(\mathbf{w})$$

where $\rho = \sqrt{\frac{2\ln 2}{\lambda}}$ is due to $\mathcal{L}(\mathbf{w}^*) \leq \ln 2$. It is easy to show that, with probability at least $1-\delta$, we have

$$\max_{\|\mathbf{w}\|_2 \leq \rho} \mathcal{L}(\mathbf{w}) - \widehat{\mathcal{L}}(\mathbf{w}) \leq \mathrm{R}_n[\mathcal{F}_{\mathbf{w}}] + \rho\sqrt{m}R\sqrt{\frac{\ln(1/\delta)}{2n}} \leq \frac{2\rho\sqrt{m}R}{\sqrt{n}} + \rho\sqrt{m}R\sqrt{\frac{\ln(1/\delta)}{2n}}$$

where $\mathrm{R}_n[\mathcal{F}]$ is the Rademacher complexity of function class $\mathcal{F}$, and $\mathcal{F}_{\mathbf{w}} = \{f(\mathrm{X}^1, \mathrm{X}^2, \mathrm{Y}) = \ln(1 + \exp(-\mathrm{Y}\mathbf{w}^\top \mathbf{k}(\mathrm{X}^1, \mathrm{X}^2))) : \|\mathbf{w}\|_2 \leq \rho\}$. Using the concentration inequality of bounded difference, with probability at least $1-\delta$, we have

$$\widehat{\mathcal{L}}(\mathbf{w}^*) \leq \mathcal{L}(\mathbf{w}^*) + \rho\sqrt{m}R\sqrt{\frac{\ln(1/\delta)}{2n}}$$

Using the fact $\widehat{\mathcal{L}}(\mathbf{w}^*) \geq \widehat{\mathcal{L}}(\mathbf{w}_a^*)$, with probability $1-2\delta$, we have

$$\mathcal{L}(\mathbf{w}_a^*) - \mathcal{L}(\mathbf{w}^*) \leq \frac{2\rho\sqrt{m}R}{\sqrt{n}} + 2\rho\sqrt{m}R\sqrt{\frac{\ln(1/\delta)}{2n}}$$

Using the strong convexity of $\mathcal{L}(\mathbf{w})$, we have $\mathcal{L}(\mathbf{w}_a^*) - \mathcal{L}(\mathbf{w}^*) \geq \frac{\lambda}{2}\|\mathbf{w}_a^* - \mathbf{w}^*\|_2^2$, and therefore, with probability $1-\delta$, have

$$\|\mathbf{w}_a^* - \mathbf{w}^*\| \leq \sqrt{\frac{4\rho\sqrt{m}R}{\lambda\sqrt{n}} + \frac{4\rho\sqrt{m}R}{\lambda\sqrt{2n}}\sqrt{\ln\frac{2}{\delta}}}$$

We complete the proof by combining the above result with the result in Theorem 4. □

BIBLIOGRAPHY

BIBLIOGRAPHY

[1] Lada Adamic and Natalie Glance. The political blogosphere and the 2004 U.S. election: divided they blog. In *Proceedings of the 3rd International Workshop on Link Discovery*, 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 *Proceedings of the International Biometrics Society Annual Meeting*, 2006.

[3] R. Albert. Scale-free networks in cell biology. *Journal of Cell Science*, 118(21):4947–4957, 2005.

[4] R. Albert and A. L. Barabási. Statistical mechanics of complex networks. *Reviews of Modern Physics*, 74:47–97, 2002.

[5] Alex M. Andrew. Reinforcement learning: An introduction. *Robotica*, 17:229–235, March 1999.

[6] A. L. Barabási and R. Albert. Emergence of scaling in random networks. *Science*, 286(5439):509–512, 1999.

[7] Marc Barthélemy and Nunes L. A. Amaral. Small-World networks: Evidence for a crossover picture. *Physical Review Letters*, 82:3180–3183, 1999.

[8] Sugato Basu, Mikhail Bilenko, and Raymond J. Mooney. A probabilistic framework for semi-supervised clustering. In *Proceedings of the 10th Annual International ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 59–68, 2004.

[9] C. M. Bishop. *Pattern recognition and machine learning (information science and statistics)*. Springer-Verlag New York, Inc., 2006.

[10] David M. Blei, Andrew Y. Ng, Michael I. Jordan, and John Lafferty. Latent dirichlet allocation. *Journal of Machine Learning Research*, 2003.

[11] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D-U. Hwang. Complex networks : Structure and dynamics. *Phys. Rep.*, 424(4-5):175–308, Fervier 2006.

[12] Deepayan Chakrabarti and Christos Faloutsos. Graph mining: Laws, generators, and algorithms. *ACM Computing Survey*, 38, June 2006.

[13] Hong Chang and Dit yan Yeung. Kernel-based metric adaptation with pairwise constraints. In *Proceedings of International Conference on Machine Learning and Cybernetics*, pages 721–730, 2005.

[14] P. Chen and S. Redner. Community structure of the physical review citation network. January 2010.

[15] Yun Chi, Xiaodan Song, Dengyong Zhou, Koji Hino, and Belle L. Tseng. On evolutionary spectral clustering. *ACM Transactions Knowledge Discovery Data*, 3:17:1–17:30, December 2009.

[16] David Cohn and Huan Chang. Learning to probabilistically identify authoritative documents. In *Proceedings of the 17th International Conference on Machine Learning*, 2000.

[17] David Cohn and Thomas Hofmann. The missing link - a probabilistic model of document content and hypertext connectivity. In *Proceedings of the 13th Neural Information Processing Systems*, 2001.

[18] Ian Davidson and S. S. Ravi. Clustering with constraints: Feasibility issues and the k-means algorithm. In *Proceedings of the 5th SIAM International Conference on Data Mining*, 2005.

[19] Ian Davidson and S. S. Ravi. Hierarchical clustering with constraints: Theory and practice. In *Proceedings of the 9th European Conference on Principles and Practice of Knowledge Discovery in Databases*, pages 59–70, 2005.

[20] Ian Davidson and Basu Sugato. A survey of clustering with instance level constraints. Technical report, 2007.

[21] Jason V. Davis, Brian Kulis, Prateek Jain, Suvrit Sra, and Inderjit S. Dhillon. Information-theoretic metric learning. In *Proceedings of the 24th Annual International Conference on Machine Learning*, pages 209–216, 2007.

[22] Luis M. De Campos, Juan M. Fernández-Luna, Juan F. Huete, Andrés R. Masegosa, and Alfonso E. Romero. Link-based text classification using bayesian networks. In *Proceedings of the Focused retrieval and evaluation, and 8th international conference on Initiative for the evaluation of XML retrieval*, INEX'09, pages 397–406, Berlin, Heidelberg, 2010. Springer-Verlag.

[23] de Solla. Networks of scientific papers. *Science*, 149(3683):510–515, July 1965.

[24] Ayhan Demiriz, Kristin Bennett, and Mark J. Embrechts. Semi-supervised clustering using genetic algorithms. In *Proceedings of Artificial Neural Networks in Engineering*, pages 809–814. ASME Press, 1999.

[25] Laura Dietz, Steffen Bickel, and Tobias Scheffer. Unsupervised prediction of citation influences. In *Proceedings of the 24th International Conference on Machine Learning*, 2007.

[26] Chris Ding, Tao Li, and Wei Peng. Nmf and plsi: equivalence and a hybrid algorithm. In *Proceedings of the 29th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pages 641–642, New York, NY, USA, 2006. ACM.

[27] Chris Ding, Tao Li, and Wei Peng. On the equivalence between non-negative matrix factorization and probabilistic latent semantic indexing. *Journal of Computational Statistics and Data Analysis*, 52:3913–3927, April 2008.

[28] Elena Erosheva, Stephen Fienberg, and John Lafferty. Mixed membership models of scientific publications. In *Proceedings of the National Academy of Sciences*, 2004.

[29] Zoubin Ghahramani. Unsupervised learning. In *Advanced Lectures on Machine Learning*, pages 72–112, 2003.

[30] M. Girvan and M. E. J. Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[31] Jacob Goldberger, Sam Roweis, Geoff Hinton, and Ruslan Salakhutdinov. Neighbourhood components analysis. In *Proceedings of the 18nd Annual Conference on Neural Information Processing Systems*, pages 513–520, 2004.

[32] Anna Goldenberg, Alice X. Zheng, Stephen E. Fienberg, and Edoardo M. Airoldi. A survey of statistical network models. *Found. Trends Mach. Learn.*, 2:129–233, February 2010.

[33] Manuel Gomez Rodriguez, Jure Leskovec, and Andreas Krause. Inferring networks of diffusion and influence. In *Proceedings of the 16th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '10, pages 1019–1028, New York, NY, USA, 2010. ACM.

[34] Steve Gregory. An algorithm to find overlapping community structure in networks. In *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, 2007.

[35] Amit Gruber, Michal Rosen-Zvi, and Yair Weiss. Latent topic models for hypertext. In *Proceedings of the 24th Annual Conference on Uncertainty in Artificial Intelligence*, null, 2008. null.

[36] Natali Gulbahce and Sune Lehmann. The art of community detection. *BioEssays*, 30(10):934–938, October 2008.

[37] J. A. Hartigan and M. A. Wong. A K-means clustering algorithm. *Applied Statistics*, 28:100–108, 1979.

[38] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer Series in Statistics. Springer New York Inc., New York, NY, USA, 2001.

[39] J. M. Hofman and C. H. Wiggins. A Bayesian approach to network modularity. *Physical Review Letters*, 100, 2008.

[40] Thomas Hofmann. Probabilistic latent semantic indexing. In *Proceedings of the 22nd International Conference on Research and Development in Information Retrieval*, 1999.

[41] Steven C. H. Hoi and Rong Jin. Active kernel learning. In *Proceedings of the 25th Annual International Conference on Machine Learning*, pages 400–407, 2008.

[42] Steven C. H. Hoi, Rong Jin, and Michael R. Lyu. Learning nonparametric kernel matrices from pairwise constraints. In *Proceedings of the 24th Annual International Conference on Machine Learning*, pages 361–368, 2007.

[43] Steven C. H. Hoi, Wei Liu, Michael R. Lyu, and Wei-Ying Ma. Learning distance metrics with contextual constraints for image retrieval. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, pages 2072–2078, 2006.

[44] Steven C. H. Hoi, Michael R. Lyu, and Edward Y. Chang. Learning the unified kernel machines for classification. In *Proceedings of the 12nd Annual International ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 187–196, 2006.

[45] P. Holland and S. Leinhardt. Local structure in social networks. *Socialogical Methodology*, 1976.

[46] Enliang Hu, Songcan Chen, Daoqiang Zhang, and Xuesong Yin. Semisupervised kernel matrix learning by kernel propagation. *Transactions Neural Networks*, 21:1831–1841, November 2010.

[47] Tony Jebara, Risi Kondor, Andrew Howard, Kristin Bennett, and Nicol Cesa-bianchi. Probability product kernels. *Journal of Machine Learning Research*, 5:819–844, 2004.

[48] C. Kemp, T. L. Griffiths, and J. B. Tenenbaum. Discovering latent classes in relational data. Technical Report AI Memo 2004-019, MIT Computer Science and Artificial Intelligence Laboratory, 2004.

[49] Jon Kleinberg. *Cascading Behavior in Networks: Algorithmic and Economic Issues.* Cambridge University Press, 2007.

[50] Brian Kulis, Mátyás Sustik, and Inderjit Dhillon. Learning low-rank kernel matrices. In *Proceedings of the 23rd Annual International Conference on Machine Learning*, pages 505–512, 2006.

[51] Brian Kulis, Mtys Sustik, and Inderjit Dhillon. Learning low-rank kernel matrices. In *Proceedings of the 23rd Annual International Conference on Machine Learning*, pages 505–512. Morgan Kaufmann, 2006.

[52] Gert R. G. Lanckriet, Nello Cristianini, Peter Bartlett, Laurent El Ghaoui, and Michael I. Jordan. Learning the kernel matrix with semidefinite programming. *Journal of Machine Learning Research*, 5:27–72, 2004.

[53] Martin H. C. Law, Alexander P. Topchy, and Anil K. Jain. Model-based clustering with probabilistic constraints. In *Proceedings of the 5th SIAM International Conference on Data Mining*, 2005.

[54] E. A. Leicht and M. E. J. Newman. Community structure in directed networks. *Physical Review Letters*, 100, 2008.

[55] L. Li, D. Alderson, J.C. Doyle, and W. Willinger. Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, 2(4):431–523, 2005.

[56] Zhenguo Li and Jianzhuang Liu. Constrained clustering by spectral kernel learning. In *Proceedings of IEEE International Conference on Computer Vision*, pages 421–427, 2009.

[57] Zhenguo Li, Jianzhuang Liu, and Xiaoou Tang. Pairwise constraint propagation by semidefinite programming for semi-supervised classification. In *Proceedings of the 25th Annual International Conference on Machine Learning*, pages 576–583, 2008.

[58] David Liben-Nowell and Jon Kleinberg. The link-prediction problem for social networks. *J. Am. Soc. Inf. Sci. Technol.*, 58:1019–1031, May 2007.

[59] Fredrick Lilijeros, Cristofer Edling, Luís Amaral, Eugene Stanley, and Yvonne Åberg. The web of human sexual contacts. *Nature*, 411:907–908, 2001.

[60] Yu-Ru Lin, Yun Chi, Shenghuo Zhu, Hari Sundaram, and Belle L. Tseng. Analyzing communities and their evolutions in dynamic social networks. *ACM Transactions Knowledge Discovery Data*, 3:8:1–8:31, April 2009.

[61] Yi Liu, Rong Jin, and Anil K. Jain. Boostcluster: boosting clustering by pairwise constraints. In *Proceedings of the 13rd Annual International ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 450–459, 2007.

[62] Qing Lu and Lise Getoor. Link-based classification. In Tom Fawcett and Nina Mishra, editors, *Proceedings of the 20th International Conference on Machine Learning*, pages 496–503. AAAI Press, Chicago, IL, USA, 2003.

[63] A. McCallum and K. Nigam. A comparisoin of event models for naive bayes text classification. In *In AAAI Workshop on Leaning for Text Categorization*, 1998.

[64] Andrew McCallum, Kamal Nigam, Jason Rennie, and Kristie Seymore. Automating the contruction of internet portals with machine learning. *Information Retrieval Journal*, 3, 2000.

[65] Stanley Milgram. The small world problem. *Psychology Today*, 2:60–67, 1967.

[66] J. M. Montoya and R. V. Solé. Small world patterns in food webs. pages 405–412, February.

[67] Peter J. Mucha, Thomas Richardson, Kevin Macon, Mason A. Porter, and Jukka-Pekka Onnela. Community structure in Time-Dependent, multiscale, and multiplex networks. *Science*, 328(5980):876–878, May 2010.

[68] Ramesh M. Nallapati, Amr Ahmed, Eric P. Xing, and William W. Cohen. Joint latent topic models for text and citations. In *Proceedings of the 14th ACM International Conference on Knowledge Discovery and Data Mining*, 2008.

[69] Amit A. Nanavati, Siva Gurumurthy, Gautam Das, Dipanjan Chakraborty, Koustuv Dasgupta, Sougata Mukherjea, and Anupam Joshi. On the Structural Properties of Massive Telecom Call Graphs: Findings and Implications. In *Proceedings of the 15th ACM International Conference on Information and Kowledge Management*, pages 435–444, November 2006.

[70] A. Nemirovski. Efficient methods in convex programming. 1994.

[71] M E Newman. Modularity and community structure in networks. *Proc Natl Acad Sci U S A*, 103(23):8577–8582, June 2006.

[72] M. E. J. Newman and M. Girvan. Finding and evaluating community structure in networks. *Physical Review E*, 69, 2003.

[73] Mark Newman. *Networks: An Introduction.* Oxford University Press, 2010.

[74] M.E.J. Newman. The structure and function of complex networks. *SIAM review*, 45(2):167–256, 2003.

[75] Dan Pelleg and Dorit Baras. K-means with large and noisy constraint sets. In *Proceedings of the 18th European Conference on Machine Learning*, pages 674–682, 2007.

[76] M. A. Porter, J.-P. Onnela, and P. J. Mucha. Communities in Networks. *ArXiv e-prints*, February 2009.

[77] Mason A. Porter, Peter J. Mucha, M. E. J. Newman, and A. J. Friend. Community structure in the united states house of representatives, February 2006.

[78] Novi Quadrianto, Alex J. Smola, Tiberio S. Caetano, and Quoc V. Le. Estimating labels from label proportions. In *Proceedings of the 25th Annual International Conference on Machine Learning*, pages 776–783, 2008.

[79] Wei Ren, Guiying Yan, Xiaoping Liao, and Lan Xiao. Simple probabilistic algorithm for detecting community structure. *Physical Review E*, 79, 2009.

[80] B. Schwikowski, P. Uetz, and S. Fields. A network of protein-protein interactions in yeast. *Nature biotechnology*, 18(12):1257–1261, December 2000.

[81] E. Segal, H. Wang, and D. Koller. Discovering molecular pathways from protein interaction and gene expression data. *Bioinformatics*, 19:i264–272, 2003.

[82] Noam Shental, Tomer Hertz, Daphna Weinshall, and Misha Pavel. Adjustment learning and relevant component analysis. In *Proceedings of the 17th European Conference on Computer Vision*, pages 776–792, 2002.

[83] J. Shi and J. Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22(8), 2000.

[84] Jianbo Shi and Jitendra Malik. Normalized cuts and image segmentation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 22:888–905, 1997.

[85] O. Sporns, D. Chialvo, M. Kaiser, and C. Hilgetag. Organization, development and function of complex brain networks. *Trends in Cognitive Sciences*, 8(9):418–425, September 2004.

[86] Gabor J. Szekely and Maria L. Rizzo. Hierarchical clustering via joint between-within distances: Extending ward's minimum variance method. *Journal of Classification*, 22(2):151–183, September 2005.

[87] Ben Taskar, Ming F. Wong, Pieter Abbeel, and Daphne Koller. Link prediction in relational data. In *Proceedings of the 17th Annual Conference on Neural Information Processing Systems*, 2003.

[88] Sebastian Thrun and Lorien Pratt. *Learning to learn: introduction and overview*, pages 3–17. Kluwer Academic Publishers, Norwell, MA, USA, 1998.

[89] Tomer Hertz Tomboy, Aharon Bar-hillel, and Daphna Weinshall. Boosting margin based distance functions for clustering. In *Proceedings of the 21st Annual International Conference on Machine Learning*, 2004.

[90] Jeffrey Travers and Stanley Milgram. An experimental study of the small world problem. *Sociometry*, 32:425–443, 1969.

[91] Vera van Noort, Berend Snel, and Martijn Huynen. The yeast coexpression network has a small-world, scale-free architecture and can be explained by a simple model. *EMBO Reports*, 5(3):280–284, March 2004.

[92] Vladimir N. Vapnik. *The nature of statistical learning theory.* Springer-Verlag New York, Inc., New York, NY, USA, 1995.

[93] A. Wagner and D. A. Fell. The small world inside large metabolic networks. *Proc R Soc Lond B Biol Sci*, 268(1478):1803–1810, September 2001.

[94] K. Wagstaff and C. Cardie. Clustering with instance-level constraints. In *Proceedings of the 17th Annual International Conference on Machine Learning*, pages 1103–1110, 2000.

[95] Kiri Wagstaff, Claire Cardie, Seth Rogers, and Stefan Schrödl. Constrained k-means clustering with background knowledge. In *Proceedings of the 18th Annual International Conference on Machine Learning*, 2001.

[96] Xuerui Wang, Natasha Mohanty, and Andrew McCallum. Group and topic discovery from relations and their attributes. In *Proceedings of the 19th Annual Conference on Neural Information Processing Systems*, 2005.

[97] Jr. Ward. Hierarchical grouping to optimize an objective function. *Journal of the American Statistical Association*, 58:236–244, 1963.

[98] S. Wasserman and K. Faust. *Social network analysis: Methods and applications.* Cambridge University Press, 1994.

[99] Stanley Wasserman, Garry Robins, and Douglas Steinley. Statistical models for networks: A brief review of some recent research. In Edoardo Airoldi, David M. Blei, Stephen E. Fienberg, Anna Goldenberg, Eric P. Xing, and Alice X. Zheng, editors, *Statistical Network Analysis: Models, Issues, and New Directions*, volume 4503 of *Lecture Notes in Computer Science*, chapter 4, pages 45–56. Springer Berlin Heidelberg, 2007.

[100] K. Weinberger, J. Blitzer, and L. Saul. Distance metric learning for large margin nearest neighbor classification. In *Proceedings of the 20nd Annual Conference on Neural Information Processing Systems*, pages 207–244, 2006.

[101] Lei Wu, Rong Jin, Steven C.H. Hoi, Jinfeng Zhuang, and Nenghai Yu. Simplenpkl: Simple non-parametric kernel learning. In *Proceedings of the 23rd Annual Conference on Neural Information Processing Systems*, 2009.

[102] Eric P. Xing, Andrew Y. Ng, Michael I. Jordan, and Stuart Russell. Distance metric learning, with application to clustering with side-information. In *Proceedings of the*

17th Annual Conference on Neural Information Processing Systems, pages 505–512, 2003.

[103] Liu Yang and Rong Jin. Distance metric learning: A comprehensive survey. Technical report, 2006.

[104] Liu Yang, Rong Jin, Rahul Sukthankar, and Yi Liu. An efficient algorithm for local distance metric learning. In *Proceedings of the 21st National Conference on Aartifical Intelligence*, pages 543–548, 2006.

[105] Tianbao Yang, Yun Chi, Shenghuo Zhu, Yihong Gong, and Rong Jin. Directed network community detection: A popularity and productivity link model. In *Proceedings of the SIAM International Conference on Data Mining*, pages 742–753, 2010.

[106] Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. Combining link and content for community detection: A discriminative approach. In *Proceedings of the 15th ACM International Conference on Knowledge Discovery and Data Mining*, 2009.

[107] Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th Annual International ACM SIGKDD Conference on Knowledge Discovery and Data Mining*, pages 927–936, 2009.

[108] X.-S. Yang. Small-world networks in geophysics. *grl*, 28:2549–2552, July 2001.

[109] K. Yu, S. Yu, and V. Tresp. Soft clustering on graphs. In *Proceedings of 19th Advances in Neural Information Processing Systems*, 2005.

[110] Shi Yu, Bart De Moor, and Yves Moreau. Clustering by heterogeneous data fusion: framework and applications. *Worpshop of the 23rd Annual Conference on Neural Information Processing Systems*, 2009.

[111] Y. Zhang, A. Friend, A. Traud, M. Porter, J. Fowler, and P. Mucha. Community structure in congressional cosponsorship networks. *Physica A: Statistical Mechanics and its Applications*, 387(7):1705–1712, March 2008.

[112] Dengyong Zhou, Jiayuan Huang, and Bernhard Schölkopf. Learning from labeled and unlabeled data on a directed graph. In *Proceedings of the 22nd International Conference on Machine Learning*, 2005.

[113] Shenghuo Zhu, Kai Yu, Yun Chi, and Yihong Gong. Combining content and link for classification using matrix factorization. In *Proceedings of the 30th annual international ACM SIGIR conference on Research and development in information retrieval*, pages 487–494, New York, NY, USA, 2007. ACM Press.

[114] X. Zhu. *Semi-supervised learning with graphs*. PhD thesis, Carnegie Mellon University, 2005.