密级_____

中国科学院大学
University of Chinese Academy of Sciences

博士学位论文

基于成对监督信息的半监督社区发现研究

作者姓名: 杨亮

指导教师: 操晓春 研究员

中国科学院信息工程研究所

学位类别: 工学博士

学科专业: 计算机软件与理论

研究所: 中国科学院信息工程研究所

2016 年 5 月

Typeset by LaTeX 2_ε at May 1, 2016

With package `CASthesis` v0.2 of CTeX.ORG

Research on the Semi-supervised Community Detection with Pairwise Prior Information

By
Liang Yang

A Dissertation Submitted to
The University of Chinese Academy of Sciences
In partial fulfillment of the requirement
For the degree of
Doctor of Computer Software and Theory

State Key Laboratory of Information Security
Institute of Information Engineering
Chinese Academy of Sciences

May, 2016

摘　　要

复杂网络中社区结构的检测对理解网络功能有着十分重要的意义, 被广泛用于恐怖组织识别、社交网络分析等实际问题。但随着网络结构的愈加复杂, 单纯依靠拓扑信息的社区检测很难获得令人满意的效果。近年来, 已提出了一些通过融合各类背景信息的半监督社区发现方法, 但这些方法在融合效率和自动确定社区个数等方面存在较大的不足，从而限制了他们的实际应用。为了使半监督社区发现方法在现实场景中有着更广的适用范围和更好的性能, 论文拟从如何提升半监督社区发现的高效性和自动性这两个方面开展研究，旨在开发能够高效利用监督信息进行自动化（同时检测社区结构和确定社区个数）精确社区发现的算法。具体的说，我们从被动高效监督信息的利用和主动待标注监督信息的选取两个方面提升监督信息的利用率，并通过将非负矩阵分解改进为能自动确定社区个数的模型使采用其作为主要部分的半监督社区发现算法能够实现社区检测的自动化。

本论文主要包括以下三个部分:

1. 高效的监督信息嵌入策略。这里，我们首先对一些被广泛使用的基于矩阵的社区发现算法（非负矩阵分析、谱聚类方法及其变种）给出一种全新的统一化解释。此类方法以表示网络拓扑结构的邻接矩阵作为输入，通过优化不同的目标函数获得节点在隐空间中的全新表达，并针对这些新的表达进行聚类操作。之后，基于这个统一化解释，我们提出一个统一的半监督社区发现框架。基于我们的统一解释，节点是根据隐空间中的全新表达进行聚类的。因此只有具有相似隐空间表达的节点才会被分到同一个社区中。如果我们有先验信息指出一对节点属于同一个社区，那么我们应该使这两个节点有着尽量相似的隐空间表达。这里，我们通过在原有待优化的目标函数中引入隐空间图正则来将先验信息与原有拓扑信息相融合。由于我们用目标函数中不同的项来刻画不同信息的作用，这个框架主要有以下两个方面的优点。一方面我们将半监督社区发现当做一个全新的问题进行处理，而不是简单的将其当做传统社区发现算法的前处理过程。另一方面它可以根据先验信息的可信度来平衡拓扑结构信息和先验信息对最终结

果的影响。此外，虽然我们引入了监督信息，但是我们提出的框架并没有大幅增加原有算法的时间复杂度。

2. 高效的主动链接选择策略。受到主动学习的启发，为了减少对监督信息的需求量，我们提出了一种可以主动选择对性能影响最为明显的链接供人工标注的链接选择框架。通过对现有的社区发现算法的研究，我们发现在传统社区发现算法得到的网络的社区结构中，社区边界上的点是最不确定且最容易被分错的节点。而且要确定一个社区，最主要的就是要确定其明确的边界。因此我们的主要想法是通过传统的社区发现方法（以非负矩阵分解为例）先找出网络中处于社区边界上的节点，并通过人工标注将这些节点与其所属的社区核心节点相连接（连接策略），同时通过进一步分析所获得的监督信息将最有可能是社区间的链接断开（断链策略）。通过以上的两步对网络拓扑结构的操作我们可以比添加随机成对监督信息更加高效地使邻接矩阵的块结构变得清晰。通过迭代上述过程来改善社区发现的算法（以非负矩阵分解为例）的性能。进一步，由于非负矩阵分解算法是初值依赖的，并且在主动链接选择过程中需要迭代的运行多次不同的非负矩阵分解，整个过程需要较长的时间。为了解决这个问题，我们提出了一种加速的策略。由于连续两次迭代过程中的邻接矩阵变化不大，因此我们将上一步非负矩阵分解的结果作为下一步的初始值。这就使得最终的运行速度是原始算法的二十分之一。另外，我们的算法的时间复杂度近线性与网络的规模，并且可以容易的被并行化和分布式。因此我们的算法适用于较大规模的网络。

3. 自动化社区发现算法（模型选择）。我们提出了一种自动加权低秩约束的非负矩阵分解算法来进行自动的社区发现（即同时确定社区结构和社区个数）。一种自然的想法是初始化较多的社区数目，再逐步的将这些社区进行合并。主要的想法是首先将分解后的矩阵列数初始化为远大于社区个数，并通过加权的低秩约束将其中的一些列压缩为零。具体的说，我们首先证明通常低秩正则项，即核模，等价于同时地同等程度的将分解后的矩阵的每列压缩到零。因此，为了使一部分列被压缩到零，我们就需要加入一个自适应加权的低秩约束。这就等价于在不同的列加入不同的压缩权重。如果权重较大，相应的列更容易被压缩到零，反之亦然。那么剩下的问题就是如何确定这些权重。这里我们并不是通过人工的方式设定这些参数，而

是在这些参数的分布上加入了两个约束。1）不能每个权重都很小，否则加权的低秩约束就起不到应有的作用。2）只能保留一部分的权重很大，因为我们只希望将部分列压缩到零。最后的优化函数是组合了非负矩阵分解的重构误差、加权低秩约束和在权重上的约束。通过最小化目标函数，我们可以使分解后的矩阵具有列稀疏的性质。最终非零列的个数即为社区的个数，且可以使用非零列将节点分到不同的社区中。我们的算法有两大优势。1）我们可以在不增加非负矩阵分解算法复杂度的同时确定社区个数。2）系统中之后一个容易调节的参数。通常情况下这个参数并不需要调节，如果要将网络划分为具有层次结构的社区，则可以适当的逐步增大这个参数。

在人工及真实网络上的性能充分说明了，相比于之前的方法，我们的算法在嵌入监督信息的性能和效率方面方面都有着显著的提升，并在自动性方面有了明显的改善。这些都使半监督社区发现有了更广阔的应用场景。

关键词：社区发现，半监督学习，主动学习，模型选择，成对约束。

Abstract

Community detection in complex networks is of significant importance for comprehending network functions, which has been widely used in many areas, such as terrorist organization recognition, social network analysis, etc. As network structures become complicated, approaches based solely on network topology cannot yield satisfactory results. Thus, some semi-supervised community detection algorithms have been proposed to alleviate this problem in the last few years. Most of them, however, are deficient in encoding efficiency and determination the number of communities, which limits their practical application. To make semi-supervised community detection can be applicable to more real areas and achieve better performance, this thesis conducts research on how to improve their efficiency and automaticity, i.e., detect the community structures and determine the number of communities simultaneously. We aim to develop a group of algorithms which can efficiently make use of supervised information and automatically detect the community structures. Specifically, we improve the efficiency of the supervised information from both high efficient supervised information embedding strategy and active supervised information selection strategy. Furthermore, we enhance the automaticity of semi-supervised community detection by transforming the original nonnegative matrix factorization, which is the key component of many semi-supervised community detection algorithms, into a new model which can automatically determine the number of communities.

This thesis consists of the following three components:

1. High Efficient Supervised Information Encoding Strategy. We first present a unified interpretation under which a group of widely-used matrix based community detection algorithms can be analyzed, including nonnegative matrix factorization, spectral clustering and their variants. The inputs of these community detection algorithms are the adjacency matrices which represent the network topology information. They first obtain the new

property representations in the latent space for all nodes by optimizing different objective functions, and then cluster nodes based on their latent space representations. And then based on this interpretation we propose a unified semi-supervised community detection framework for these methods. Under our interpretation, nodes are clustered based on the new property representation. It is believed that the property representations of intra-community nodes are more similar than that of inter-community nodes. If we have priors that some nodes belong to the same community, we should make them have the similar latent space representations. Here, we introduce a latent space graph regularization term into the original objective functions to incorporate the prior information with the topology information. Since we characterize different constraints using different terms in the objective function, the proposed framework has the following two advantages. First, we treat semi-supervised community detection as a brand new problem instead of handling it as a preprocess of traditional community detection problem. Second, the proposed framework is flexible to balance the factors between the topology information and prior information according to reliability of priors. Besides, though we introduce the graph regularization term into the objective function, the proposed framework does not significantly increase the computational complexity.

2. High Efficient Active Link Selection Strategy. Motivated by the active learning, in order to reduce the demand for the supervised information, we propose an active link selection framework which actively selects some links that significantly impact the performance for human labelling. Through the study of existing community detection methods, we find out that the nodes near the community boundaries are the most uncertain and informative ones and are critical for accurate community detection. The key task to detect a community is to determine its boundary. The main idea is that first we find the uncertain nodes near the community boundaries using the traditionally community detection algorithms such as nonnegative matrix factorization. Then, we connect uncertain nodes to their community

hubs by human labelling (Connection Strategy) and disconnect the inter-community edges by further analyzing the obtained supervised information (Disconnection Strategy). By these two steps, we can sharpen the block structure of adjacency matrix more efficiently than randomly labeling links as the existing methods did. The performance of community detection can be improved by repeating the above steps. Furthermore, since the nonnegative matrix factorization algorithms are dependent on the initialization and need repeat on many different initializations, iteratively conducting many different nonnegative matrix factorizations will be time-consuming. To solve this problem, we present a speedup scheme. We take the result of previous nonnegative matrix factorization as the initialization of the next one, since the adjacency matrices of the two consecutive iterations are similar. The running time of the speedup scheme is shortened to one-twentieth of the original one. In addition, our framework is near linear with the size of the network, and can easily be parallelized and distributed Thus, it has the potential to be applied to real large-scale problems.

3. Autonomous Community Detection (Model Selection). We propose a nonnegative matrix factorization with adaptively weighted low-rank constraint to autonomously detect community structure. A natural idea is to initialize a large number of communities and gradually merge some of them. The main idea is that we first initialize the number of columns of factorized matrix be much larger than the real number of communities and suppress some of them to zero using the weighted low-rank constraint instead of directly pre-defining the number of columns of obtained matrix. Specifically, firstly of all, we prove that the normal low-rank regularization term, i.e., the nuclear norm, is equivalent to suppressing each column of the obtained matrix to be zero simultaneously and equally. Thus, to make only part of the columns to be suppressed to zero, we add an adaptively weighted low-rank constraint. It is equivalent to adding different weights to different columns of the factorized matrix. If the weight is large, the corresponding column is much easier to be suppressed, and vice versa. The remaining

problem is how to set these weights. Here, instead of setting them artificially, we only add two constraints to them: 1) Each of the weights can not be too small, otherwise the weighted low-rank constraint is invalid. 2) Not all the weights are too large, since we only want part of the columns to be suppressed to zero. The final objective function is the combination of the reconstruction error of the NMF, weighted low-rank constraint and the constraints on the weights. By minimizing the objective function we can obtain the factorized matrices with some zero columns. The number of the non-zero columns is that of communities, and we use the non-zero columns to classify nodes into different communities as previous work did. There are two main merits of our approach: 1) It can automatically determine the number of communities without increasing the complexity of NMF. 2) There is only one parameter that needs to be simply tuned. The satisfactory results can always be obtained by setting that parameter to 1, while we can appropriately and gradually improve that parameter to detect hierarchical community.

Keywords: community detection, semi-supervised learning, active learning, model selection, pairwise constraint.

目录

摘要 i

Abstract v

目录 ix

第一章 引言 1

1.1 研究背景及意义 1

1.2 研究内容 3

1.3 论文结构 4

第二章 预备知识 5

2.1 符号定义 5

2.2 基于随机块模型的社区发现算法 5

2.3 基于最大模块性的社区发现算法 7

2.4 半监督社区发现算法概述 9

2.4.1 基础独立节点信息的半监督社区发现算法 9

2.4.2 基于成对关系信息的半监督社区发现算法 11

2.5 性能衡量标准 12

2.6 常用数据集 13

2.6.1 真实网络数据集 13

2.6.2 人工网络数据集 15

第三章 统一的半监督社区发现 19

3.1 引言 19

3.2 统一化解释 20

3.3 基于隐空间图正则的统一半监督框架 ····· 21
3.3.1 模型概述 ····· 21
3.3.2 优化算法 ····· 24
3.3.3 时间复杂度分析 ····· 26
3.3.4 模型选择 ····· 28
3.4 实验结果与分析 ····· 28
3.4.1 人工网络 ····· 29
3.4.2 错误先验影响 ····· 33
3.4.3 参数设置 ····· 35
3.4.4 真实网络 ····· 38
3.4.5 用例研究 ····· 40
3.5 小结 ····· 42

第四章 主动半监督社区发现 ····· 43
4.1 引言 ····· 43
4.2 主动边选择策略 ····· 44
4.2.1 模型概述 ····· 44
4.2.2 选择策略 ····· 45
4.2.3 加速策略 ····· 49
4.2.4 时间复杂度分析 ····· 49
4.3 实验结果与分析 ····· 50
4.3.1 参数设定 ····· 50
4.3.2 人工网络 ····· 51
4.3.3 真实网络量化比较 ····· 53
4.3.4 真实网络用例研究 ····· 57
4.4 小结 ····· 62

第五章 自动化社区发现 ····· 65
5.1 引言 ····· 65

5.1.1 相关工作 ····· 66
5.2 低秩矩阵近似 ····· 66
5.2.1 已知秩的低秩近似 ····· 67
5.2.2 未知秩的低秩近似 ····· 68
5.3 自动加权的低秩矩阵近似 ····· 69
5.3.1 等权低秩约束 ····· 69
5.3.2 加权低秩约束 ····· 69
5.3.3 自适应权重设定 ····· 71
5.3.4 目标函数与优化 ····· 74
5.3.5 参数设定与含义 ····· 75
5.3.6 自动社区发现 ····· 76
5.3.7 时间复杂度分析 ····· 76
5.4 实验结果与分析 ····· 78
5.4.1 人工网络 ····· 78
5.4.2 参数设定 ····· 80
5.4.3 算法验证 ····· 83
5.4.4 真实网络 ····· 84
5.5 小结 ····· 87

第六章 总结与展望 ····· 89
6.1 总结 ····· 89
6.2 展望 ····· 90

参考文献 ····· 93

发表文章目录 ····· 107

简历 ····· 109

致谢 ····· 111

表格

2.1 本文中用到的真实网络。 ····· 15

3.1 基于我们提出框架的 4 种不同方法在 8 个真实网络上的性能。··· 39

5.1 在真实网络上各种方法基于归一化互斥信息（NMI）的性能及检测到的社区个数。 ····· 85

插图

3.1 对包含随机快模型及最大模块性模型在内的一些经典的社区发现算法的统一解释。 ····· 21

3.2 能够将成对先验信息与拓扑结构信息相融合的统一半监督社区发现框架。 ····· 22

3.3 在 GN 网络上，原始社区发现算法的性能随着社区间边数目 Z_{out} 的变化情况。 ····· 29

3.4 在 GN 网络上，我们框架的性能随着加入先验信息的比例的变化情况。 ····· 30

3.5 在社区结构相对清晰（$\mu = 0.7$）的 LFR 网络上，我们框架的性能随着加入先验信息的比例的变化情况。 ····· 31

3.6 在社区结构相对模糊（$\mu = 0.8$）LFR 网络上，我们框架的性能随着加入先验信息的比例的变化情况。 ····· 32

3.7 与其他半监督社区发现算法的性能比较。 ····· 33

3.8 错误先验信息对性能的影响。 ····· 34

3.9 在 GN 网络上，基于我们提出的框架的各类算法的性能随着平衡参数 λ 的变化情况。 ····· 35

3.10 在混合参数 $\mu = 0.8$ 的 LFR 网络上，基于我们提出的框架的各类算法的性能随着平衡参数 λ 的变化情况。 ····· 37

3.11 在社区规模较大的 LFR 网络上，基于我们框架的 SNMF 算法的性能的变化曲线。三幅子图分别对应混合参数 $\mu = 0.7$，0.75 和 0.8 的网络。 ····· 38

3.12 一个关于美国政治书的网络的实例研究。 ····· 41

4.1 主动边选择框架的流程图。 ····· 45

4.2 基于节点隶属度分布的熵确定节点在社区中的角色。其中熵较大的认为是出于社区边界，熵较小的认为是出于社区核心。 ····· 46

4.3 主动边选择策略的可视化解释。(a) 网络拓扑结构及被选择的链接。(b) 连接策略。(c) 断连策略。 ····· 47
4.4 在 $Z_{out} = 8$ 的 GN 网络上参数 n_{select} 对性能提升的影响。其中每条曲线代表选取某一 n_{select} 时性能随着先验信息增加的增长速度。 ····· 50
4.5 在 $Z_{out} = 8$ 的 GN 网络上，邻接矩阵的变化情况。 ····· 51
4.6 在人工网络上主动链接选择框架的性能。 ····· 53
4.7 在 6 个真实网络上主动链接选择框架的性能。 ····· 54
4.8 在 12 个美国大学 Facebook 网络上主动链接选择框架的性能。 ··· 56
4.9 在使用我们的主动链接选择策略前后的海豚网络的网络拓扑结构和检测到的社区结构。 ····· 57
4.10 在使用我们的主动链接选择策略前后的美国高校足球俱乐部网络的拓扑结构和检测到的社区结构。 ····· 59
4.11 在使用我们的主动链接选择策略前后的美国高校网络的网络拓扑结构和检测到的社区结构。 ····· 61
4.12 在使用我们的主动链接选择策略前后的美国政治书网络的网络拓扑结构和检测到的社区结构。 ····· 63

5.1 三种非负矩阵分解方法的比较。 ····· 67
5.2 损失函数 $g_1(x)$ 在不同底数 b 的时候曲线。当 b 较大时，曲线快速下降到 0 并且有一个较平的尾部，反之当 b 较小时，起初下降的较慢且尾部较陡峭。 ····· 72
5.3 损失函数 $f(x)$ 在不同底数 b 时的曲线。较大的 b 使 $f(x)$ 在较小的 x 处达到最小值，反之则在较大的时候达到最小值。因此为了使 $f(x)$ 在 $x = 1$ 时达到最小值，我们令 $b = e$。 ····· 73
5.4 NMF-AWL 和 4 个现有的自动社区发现算法在 GN 网咯上的性能对比。 ····· 79
5.5 NMF-AWL 和 4 个现有的自动社区发现算法在 LFR 网络上的性能对比。衡量标准及符号标记与图 5.4 中一致。 ····· 80
5.6 在不同混合参数 μ 的 LFR 网络上，参数 α 和 b 的作用。我们给出不同的 α 和 b 的组合对基于 NMI 的社区检测结果的影响。 ··· 81

5.7 在混合参数较大的 LFR 网络上（μ 从 0.4 到 0.6），对图 5.6 中的 3 维柱状图在 $b = e$ 时的二位横断面。 ····· 82
5.8 初始矩阵列数 p 占比对最终结果的影响。(a) 对最终社区检测精度的影响，(b) 对最终检测到的社区个数的影响。横坐标代表初始列数的比例。 ····· 82
5.9 随着迭代过程，分解后矩阵的列的模及相应的 σ 的变化过程。 ··· 83
5.10 参数 α 在真实网络分层社区划分上的作用。 ····· 86
5.11 参数 α 在美国政治书网络分层社区划分上的作用。 ····· 87

第一章　引言

1.1　研究背景及意义

现实世界中的很多复杂系统都可以通过复杂网络来进行建模，如社会网络 [99]、生物网络 [53]、科技网络 [40] 等。通过研究，人们发现社区结构（或模块性）[67, 68] 是复杂网络最为普遍和最为重要的属性之一。通过检测复杂网络中的社区结构可以帮助人们深入理解网络的基本结构和功能，挖掘网络中蕴含的深层次规律并预测网络进一步的特征。该问题的研究不仅具有十分深远的理论意义，而且对国民经济和现代化建设有着重要的现实意义。社区发现技术已经被广泛的应用于恐怖组织识别、蛋白质功能预测和新药物研发、搜索引擎优化与推荐系统设计中。

近年来，学术界提出了许多的社区检测方法 [26, 33, 41, 56, 66, 100]。其中的绝大多数都是单纯的将网络的拓扑结构作为输入。虽然他们中的很多在现实生活中都取得了不错的表现，但是此类方法通常只能处理社区结构比较清晰的网络。当网络中社区结构变得模糊时，这些方法通常都无法获得令人满意的性能。这种检测的困难性主要是由两个方面的原因引起的。一方面，由于网络中的边是非常稀疏 [18, 91] 而且具有大量噪音的，因此单纯依赖拓扑结构来进行社区划分是不可靠的。另一方面，网络中的社区通常具有非常复杂的结构，如重叠社区结构 [2, 70, 101, 103] 和层级社区结构 [35, 71, 77, 82]，从而需要一些先验知识来辅助精准的社区发现 [4]。此外，近期的关于社区可检测性的研究表明，社区间边数和社区内部边数的差值存在一个相变阈值，即当这个差值超过某个阈值的时候，任何方法都无法正确检测社区的结构 [21, 62, 74–76]。

在很多的真实场景中，除了复杂网络的拓扑信息外，我们还可以获得一些关于节点或者链接的监督信息 [73, 78, 105–107]。譬如在社会化网络中（如微博和人人网），根据账户的个人信息及其行为记录，可以比较容易地获得某个账户属于哪个社区，某个账户一定不属于哪个社区，或某两个账户是否在同一个社区这样的信息。这些监督信息都可以用以辅助原始的基于拓扑结构的无监督社区发现。

针对不同种类的监督信息，已经提了很多不同的半监督社区发现方法。针

对基于节点监督信息的情况，一种最为自然和直观的方法就是假设相邻节点间具有相似的标签并采取标签传播 [39, 47, 122] 等算法将监督信息和原始拓扑结构相融合，从而提升社区检测的准确性。而如何将链接的监督信息与原始的拓扑结构相融合成为了近两年的研究热点 [42, 114, 115]。所谓基于链接的信息，就是判断任意两个节点是否属于同一个社区。如果属于同一个社区，他们之间就具有“必连”关系；如果不属于，则存在“必不连”关系。一种最为直观的方法就是将监督信息适当的转化为拓扑结构信息，即如果两个节点之间有“必连”关系，则在他们之间增加一条特定权重的边；如果两个节点之间有“必不连”关系，则断开他们之间可能已经存在的边。

然而这些直观方法并不是完美的，他们所存在的最大问题就是无法保证监督信息的利用的高效性。所谓高效性：一方面是指半监督框架可以充分地利用已经获得的监督信息。系统获得的监督信息相对于原始的拓扑结构信息虽然数量上更少，但是信息量相对较大。例如在拓扑结构中两个点之间的边只表明他们之间存在相互关系；而我们获得的“成对”监督信息指出如果两个节点是“必连”关系，这两个节点就属于同一个社区。可见监督信息具有较强的约束力，而在现有方法中这种强约束通常被忽略。另一方面，高效性是指系统可以根据拓扑结构主动地选择最具有信息量、最不确定的节点或链接，供人工标注。这种主动选择获取的监督信息由于具有针对性，因此更能大幅提升系统准确度。由于监督信息的获取是非常困难且昂贵的，如何能高效地获取并利用监督信息，对于半监督社区发现是至关重要的。如果不能充分利用这些监督信息，则为了提升性能就要收集大量的监督信息。这就极大地削弱了半监督社区发现算法的实际意义。

现有半监督社区发现算法存在的另外一个重要问题就是在检测社区结构前需要预先精确设定复杂网络中所含社区的个数。也就是说不能实现完全自动化的社区发现。而在实际应用中，社区个数通常很难提前确定，尤其是针对半监督社区发现算法主要针对的社区结构相对模糊的网络。半监督社区发现算法无法实现完全自动化也限制了半监督社区发现算法被广泛的应用到具体的实际问题中。

因此，半监督社区发现算法嵌入监督信息的效率及能否实现自动的社区发现成为半监督社区发现算法领域两个重要的研究课题，它们直接决定了半监督社区发现算法的适用场景和应用范围。

1.2　研究内容

根据此前的讨论可知，半监督社区发现算法嵌入监督信息的效率及算法的自动化程度极大的影响了算法的适用场景和应用范围。故此毕业论文将主要研究如何提升半监督社区发现算法嵌入监督信息的效率和算法自身的自动性。我们首先将从被动高效利用和主动积极获取两个角度来提升框架对监督信息的利用效率。之后，我们还将通过使常见的半监督社区发现算法的核心部分实现自动化来改善整体半监督社区发现算法的自动性。具体的说，我们将主要研究以下三方面的内容：

1) 高效的监督信息融合策略。和传统的拓扑结构信息相比，虽然监督信息的数量较少，但是其中蕴含了丰富的信息。例如在复杂网络中，两个点之的边只说明这两个点之间有相互关系，不能说明他们属于一个社区。而在“成对”的监督信息中，当两个节点之间有“必连”关系是，两个点肯定属于同一社区。如何高效地利用这些具有强约束的监督信息，来提升原有基于拓扑结构的社区发现算法的性能，是我们研究的一个重要方面。

2) 高效的边候选策略。为了进一步减少对监督信息的需求量，相对于被动使用随机获得的网络中成对的监督信息，一种更为有效的方法就是根据现有的拓扑结构信息，主动的选择最有可能被分错或者对整体性能有较大影响的链接供人工标注。这种主动学习的方法在监督学习中已经被广泛应用。但针对于社区发现这种无监督学习，如何能够根据拓扑结构来选择节点或者链接是一个非常有挑战性的问题，同时如何能够高效地选择出这些边，将对整体性能的提升有着重要的意义。

3) 有效的模型选择策略。现有的半监督社团发现方法通常都是预先要检测的社区个数是已知的，而监督信息只是用来提升社区发现的精度。但在实际应用中，社区数目是很难提前获得的。因此，如何能够完全自动化的发现社区，即同时检测社区个数和结构，依然是社区发现中的重要课题。大多数现有的半监督社区发现算法都是以非负矩阵分解作为其核心组成部分，因此如果能实现非负矩阵分解的自动化，则能使绝大多数半监督社区发现算法实现自动化。此外随着监督信息的加入，社区结构和社区数目的检测都将获益。因此，我们希望通过实现基于非负矩阵分解的社区发现算法的自动化，使随着成对监督信息的加入，不仅社区结构检测的精度明显提升，社区数目的确定也变得容易。

总之，论文希望通过对半监督社区发现中的高效监督信息融合策略、高效

边候选策略和自动模型选择策略的研究，大幅提升嵌入信息对社区发现算法性能的增长的影响，并在提升性能的同时自动检测社区个数，实现真正的自动检测。

1.3 论文结构

根据论文整体要解决的基于成对监督信息的半监督社区发现问题，以及具体要解决的高效监督信息融合策略、高效边候选策略和自动模型选择策略这三个子问题，论文的余下部分将如此组织：在第二章我们主要对社区发现问题进行概述，包括介绍论文中将用到的符号，现有的经典社区发现算法和半监督社区发现算法、验证算法所使用的度量标准及常用的数据集。在第三章，我们主要针对第一个关键问题，即如何高效的融入成对监督信息，开展讨论，并提出了基于隐空间图正则的半监督社区发现算法。在第四章，我们主要针对第二个关键问题，即如何高效的选择候选链接供人工标注，开展研究，并提出了融合连接策略和断连策略的主动链接选择框架。在第五章，我们将主要针对第三个关键问题，即如何设计有效的模型选择策略，开展工作，并提出了一个自动加权低秩约束的非负矩阵分解算法。最后，在第六章，我们将整体总结本论文的工作，并指出今后的研究方向及要进一步解决的问题。

第二章　预备知识

本章我们首先给出论文中常用符号的定义，其次介绍两类被广泛使用的社区发现算法，即随机块模型和最大模块性模型。这两类方法将是我们研究半监督社区发现的基础。之后我们对现有的半监督社区发现方法进行全面的综述。最后，我们对后面各章实验部分将用到的性能度量标准和常用的数据集进行简要的介绍。

2.1　符号定义

一个网络可以用一个图 $G=(V,E)$ 来建模。这里 V 是节点的结合，E 是边的集合，其中每条边链接 V 中的两个顶点。为了简单起见，我们假设 G 是一个含有 N 个节点的无向无权图。我们用一个非负对称二值矩阵 $\mathbf{A}=[a_{ij}]\in\mathbb{R}_+^{N\times N}$ 来代表图 G 的邻接矩阵，其中元素 a_{ij} 表示节点 i 和 j 之间是否存在一条边。当且仅当他们之间存在边时 $a_{ij}=1$，否则 $a_{ij}=0$。并且，对于所有的 $1\le i\le N$，我们令 $a_{ii}=0$。此外，我们记 $d_i=\sum_{j=1}^N a_{ij}$ 为节点 i 的度，$M=\frac{1}{2}\sum_{i=1}^N d_i$ 为图中链接的总数。进一步，在未说明的情况下，我们假设已知网络中有 K 个社区。社区检测问题是根据网络拓扑结构 $\mathbf{A}$ 将网络中的这些节点分为 K 个不同的组 $\{V_j\}_{j=1}^K$，且满足 $\cup_{j=1}^K V_j=V$。由于本文只对非重叠社区发现算法进行讨论，划分的社区对于所有的 $i\ne j$ 满足 $V_i\cap V_j=\emptyset$。

2.2　基于随机块模型的社区发现算法

随机块模型 [3, 31] 主要从网络数据的生成过程这个角度出发对社区结构进行建模，即所谓的生成模型。首先，我们将节点 i 和 j 之间存在边的概率（即是否存在链接）a_{ij} 视为观测变量。我们假设节点 i 和 j 之间存在一条属于社区 k 的链接是由这两个节点分别生成属于社区 k 的入边和出边决定的。我们定义两个隐变量 $\mathbf{W}=[w_{ik}]\in\mathbb{R}_+^{N\times K}$ 和 $\mathbf{H}=[h_{jk}]\in\mathbb{R}_+^{N\times K}$，其中 w_{ik} 和 h_{ik} 分别代表节点 i 生成属于社区 k 的出边和入边的概率。因此节点 i 和 j 之间存在一条属于社区 k 的链接的概率是 $w_{ik}h_{jk}$。由于总共有 K 个社区，因此节点 i 和 j 之间

存在链接的概率是:

$$\hat{a}_{ij} = \sum_{k=1}^{K} w_{ik} h_{jk}. \tag{2.1}$$

其中 $\hat{a}_{ij}$ 是根据我们生成过程所得到的网络的节点 i 和 j 存在链接的概率。通过将生成的网络数据 $\hat{a}_{ij}$ 与观测到的真实网络数据 a_{ij} 相拟合，

$$a_{ij} \approx \hat{a}_{ij} = \sum_{k=1}^{K} w_{ik} h_{jk}. \tag{2.2}$$

我们即可以推断出我们模型中的两个隐变量 $\mathbf{W}$ 和 $\mathbf{H}$，用矩阵的形式可以表示为:

$$\mathbf{A} \approx \widehat{\mathbf{A}} = \mathbf{W}\mathbf{H}^T \tag{2.3}$$

可见推断隐变量 $\mathbf{W}$ 或 $\mathbf{H}$ 的过程就是用这两个隐变量重构邻接矩阵 $\mathbf{A}$ 的过程。我们假设一个节点生成属于社区 k 的边的概率等价于该节点属于社区 k 的概率，即 w_{ik} 和 h_{ik} 分别代表节点 i 属于出社区 k 和入社区 k 的概率。$\mathbf{W}$ 或 $\mathbf{H}$ 的第 i 行分别代表节点 i 属于各个社区的概率分布。因此，我们就可以根据 $\mathbf{W}$ 或 $\mathbf{H}$ 中第 i 行最大元素的下标，推断出节点 i 所属的社区。这样我们就将社区发现问题转化为了非负矩阵分解（Nonnegative Matrix Factorization（NMF））问题。由于我们这里考虑的是无向网络，即邻接矩阵 $\mathbf{A}$ 是对称的，分解后的 $\mathbf{W}$ 和 $\mathbf{H}$ 通常只相差一个常数。因此不失一般性，我们这里使用分解后的矩阵 $\mathbf{H}$ 作为社区划分的依据。

有两种常用的量化矩阵分解结果的目标函数（损失函数）。第一个是基于平方损失函数，它等价于两个矩阵差的 Frobenius 模的平方:

$$\mathcal{L}_{LSE}(\mathbf{A}, \mathbf{W}\mathbf{H}^T) = ||\mathbf{A} - \mathbf{W}\mathbf{H}^T||_F^2 \;=\; \sum_{i,j} \Big(a_{ij} - \sum_{k=1}^{K} w_{ik} h_{jk} \Big)^2. \tag{2.4}$$

第二个是基于两个矩阵的 Kullback-Leibleer 散度 (KL-散度)

$$\begin{aligned} \mathcal{L}_{KL}(\mathbf{A}, \mathbf{W}\mathbf{H}^T) &= KL(\mathbf{A}||\mathbf{W}\mathbf{H}^T) \\ &= \sum_{i,j} \Big(a_{ij} \log \Big(\frac{a_{ij}}{\hat{a}_{ij}} \Big) - a_{ij} + \hat{a}_{ij} \Big). \end{aligned} \tag{2.5}$$

在不同的网络及应用中，可以选择不同的损失函数。

一种非负矩阵分解的变种就是对称非负矩阵分解 [92, 93]，它将对称性约束加入到了非负矩阵分解中。由于我们考虑的是邻接矩阵为对称矩阵的无向网络，因此对称非负矩阵分解可以如下表示：

$$\mathcal{L}_{SYM}(\mathbf{A}, \mathbf{H}\mathbf{H}^T) = ||\mathbf{A} - \mathbf{H}\mathbf{H}^T||_F^2 = \sum_{i,j}\Big(a_{ij} - \sum_{k=1}^{K} h_{ik}h_{jk}\Big)^2 \tag{2.6}$$

由于 $\mathcal{L}_{LSE}$, $\mathcal{L}_{KL}$ 和 $\mathcal{L}_{SYM}$ 对于两个变量 $\mathbf{W}$ 和 $\mathbf{H}$ 不是联合凸的，故而并不容易设计能找到这些损失函数全局最优解的优化算法。但是由于这些损失函数对单一的 $\mathbf{W}$ 或 $\mathbf{H}$ 是凸的，Lee 和 Seung [38] 提出了一种求解 $\mathcal{L}_{LSE}$ 的迭代更新算法：

$$w_{ik} \leftarrow w_{ik}\frac{(\mathbf{A}\mathbf{H})_{ik}}{(\mathbf{W}\mathbf{H}^T\mathbf{H})_{ik}}, \quad h_{jk} \leftarrow h_{jk}\frac{(\mathbf{A}^T\mathbf{W})_{jk}}{(\mathbf{H}\mathbf{W}^T\mathbf{W})_{jk}}. \tag{2.7}$$

类似的，公式 (5.12) 中的 $\mathcal{L}_{KL}$ 和公式 (2.6) 中的 $\mathcal{L}_{SYM}$ 可以用如下的两个更新规则进行求解：

$$\begin{aligned} w_{ik} &\leftarrow w_{ik}\frac{\sum_j (a_{ij}h_{jk}/\sum_k w_{ik}h_{jk})}{\sum_j h_{jk}}, \\ h_{jk} &\leftarrow h_{jk}\frac{\sum_i (a_{ij}w_{ik}/\sum_k w_{ik}h_{jk})}{\sum_i w_{ik}}, \end{aligned} \tag{2.8}$$

$$h_{ik} \leftarrow h_{ik}\Big(\frac{1}{2} + \frac{(\mathbf{A}\mathbf{H})_{ik}}{(2\mathbf{H}\mathbf{H}^T\mathbf{H})_{ik}}\Big). \tag{2.9}$$

除此之外，从不同的随机块模型出发，人们还提出了很多不同的非负矩阵分解的算法 [96, 111]，如非负对阵矩阵三分解 [16]、二值非负矩阵分解 [116]、有界非负矩阵分解 [112]。除了被成功的应用于社区发现外，非负矩阵分解还被广泛的应用于计算机视觉 [37, 117]、信息检索 [102, 123]、音乐处理 [9, 85] 等领域，并针对不同的应用提出了很多的变种及扩展 [97]，同时也提出了一些半监督和约束的非负矩阵分解算法 [17, 52, 94]。

2.3　基于最大模块性的社区发现算法

与源于生成模型的随机块模型不同，基于最大模块性模型的社区发现算法是一种判别模型。它通过刻画检测出的社区结构应具有模块性的特点，并通过最大化模块性函数来实现社区发现。如前所述，社区结构是网络中一些内部紧密联

系的子网络，故由此检测出的社区结构应该能够最大化社区中的边数与在等价网络中随机放置所得到的边数的差。因此我们就将这个需要最大化的量称作社区结构的模块性（modularity）。从而社区发现的过程，即是寻找能达到最大化模块性的网络划分。Newman 等人提出了最为流行的模块性的定义 [67, 68]。

为了简单起见，我们首先着手分析含有两个社区（V_1 和 V_2）的网络。当网络中有 M 条边，节点 i 和 j 的度分别为 d_i 和 d_j，那么当网络中的边完全随机放置时，节点 i 发出的任意一条边，这条边与 j 点相连的概率是 $\frac{d_j}{\sum_i d_i} = \frac{d_j}{2Ms}$。由于节点 i 有 d_i 条边，那么节点 i 和 j 之间的边数的期望是 $\frac{d_i d_j}{2M}$。则第 k（$k = 1$ 或 2）个社区的模块性可以写为

$$\sum_{i,j \in V_k} \left(a_{ij} - \frac{d_i d_j}{2M} \right) \tag{2.10}$$

如果节点 i 属于第一个社区，那么我们令 $h_i = 1$，否则 $h_i = -1$。通过令 $\delta(h_i, h_j) = \frac{1}{2}(h_i h_j + 1)$，我们有：如果节点 i 和 j 属于同一个社区，那么 $\delta(h_i, h_j) = 1$；如果他们属于不同的社区，那么 $\delta(h_i, h_j) = 0$。这样，我们就可以将整个网络的模块性写为

$$\begin{aligned} Q &= \frac{1}{2M} \sum_{k=1}^{2} \sum_{i,j \in V_k} \left(a_{ij} - \frac{d_i d_j}{2M} \right) \\ &= \frac{1}{2M} \sum_{i,j} \left(a_{ij} - \frac{d_i d_j}{2M} \right) \delta(h_i, h_j) \\ &= \frac{1}{4M} \sum_{i,j} \left(a_{ij} - \frac{d_i d_j}{2M} \right) (h_i h_j + 1) \\ &= \frac{1}{4M} \sum_{i,j} \left(a_{ij} - \frac{d_i d_j}{2M} \right) h_i h_j + const. \end{aligned} \tag{2.11}$$

其中，由于总共有 M 条边，且每条边被计算了两次，因此我们加入了归一化因子 $\frac{1}{2M}$。这里我们定义模块性矩阵 $\mathbf{B} = [b_{ij}] \in \mathbb{R}^{N \times N}$，它的每一个元素

$$b_{ij} = a_{ij} - \frac{d_i d_j}{2M} \tag{2.12}$$

并定义社区指示向量 $\mathbf{h} = [h_i] \in \mathbb{R}^N$，我们有 $\mathbf{h}^T \mathbf{h} = N$。因此我们可以将最大化模块性问题写成如下的形式：

$$\begin{aligned} \mathbf{h}^* = \underset{\mathbf{h}}{\operatorname{argmax}} \frac{1}{4m} \mathbf{h}^T \mathbf{B} \mathbf{h} \\ \text{s.t.}\, \mathbf{h}^T \mathbf{h} = N \end{aligned} \tag{2.13}$$

下面我们将最大化模块性问题推广到具有 $K > 2$ 个社区的网络中，通过定义隶属度指示矩阵 $\mathbf{H} = [h_{ik}] \in \{0,1\}^{N\times K}$，其中如果节点 i 属于第 k 个社区，则 $h_{ik} = 1$，否则 $h_{ik} = 0$，我们可以将公式 (2.13) 推广到：

$$\mathbf{H}^* = \underset{\mathbf{H}}{\operatorname{argmax}}\, \mathcal{L}_{MOD}(\mathbf{H},\mathbf{B}) = \underset{\mathbf{H}}{\operatorname{argmax}}\, \frac{1}{4m}\operatorname{Tr}(\mathbf{H}^T\mathbf{B}\mathbf{H}),$$
$$s.t.\ \operatorname{Tr}(\mathbf{H}^T\mathbf{H}) = N. \tag{2.14}$$

其中 Tr(.) 代表矩阵的迹。最大化公式（2.14）被证明是一个 NP-hard 问题 [11]，且为了求解这个问题提出了很多的近似优化算法。这里，我们将变量 h_{ik} 松弛到可以为任意的实数。根据 Rayleigh Quotient 定理，求解这个问题等价于寻找模块性矩阵 $\mathbf{B}$ 的最大的 K 个特征值对应的特征向量。获得的 $\mathbf{H}$ 的每一行可以看做是节点 i 的一个全新的表达，并可以使用如 k-均值的聚类算法进行聚类。最大化模块性问题（2.14）与很多的图分割问题密切相关，如 normalzied cuts 等。如 [80] 中所讨论的，除去模块性矩阵 $\mathbf{B}$ ($\mathcal{L}_{MOD}$), 其他的与网络拓扑结构相关的度量矩阵也可以用来替换公式（2.14）中的 $\mathbf{B}$ 来实现社区发现，如邻接矩阵 $\mathbf{A}$ ($\mathcal{L}_{ADJ}$), 标准 Laplacian 矩阵 $\mathbf{L}_S = \mathbf{D} - \mathbf{A}$ ($\mathcal{L}_{LAP}$) 和归一化的 Laplacian 矩阵 $\mathbf{L}_N = \mathbf{I} - \mathbf{D}^{-1}\mathbf{A}$ ($\mathcal{L}_{NLAP}$)，其中 $\mathbf{D}$ 是对角矩阵，且第 i 个对角元素 d_i 对应与节点 i 的度。

虽然，最大模块性模型在社区发现领域取得了很大的成功 [5]，但是很多最新的工作也指出了此模型存在的一些问题，如极端退化现象 [29] 和分辨率极限问题 [8, 27, 34]。

2.4　半监督社区发现算法概述

为了解决社区可检测性问题，尤其是在社区结构相对模糊的网络上社区发现性能提升问题，在过去几年中研究人员提出了一些半监督的社区发现算法。根据融入监督信息类别的不同，半监督社区发现方法大致可以分为两类。一类是基于独立节点信息的半监督社区发现算法；另一类是基于成对关系信息的半监督社区发现算法。

2.4.1　基础独立节点信息的半监督社区发现算法

社区检测算法是将网络中的节点划分到不同的社区中，也就是说根据相关信息估计每个节点所属社区的标签。因此最为直接的监督信息就是与社区发现任务

相关的每个节点所属的社区信息。一些半监督社区发现算法从社区可检测性或者标签传播的角度出发考虑如何嵌入这类的监督信息。

在无监督社区发现中，有关社区可检测性的研究指出，社区间链接数与社区被链接数之间的差存在一个相变阈值，超过这个阈值社区结构就不能被检测到。从社区可检测性角度出发，Allahverdyan 等人研究先验信息对可检测性阈值的影响 [4]。作者从统计物理的角度出发，通过分析一些在随机二等分图中自旋状态已知并被冻结的的 spin-glass 模型，给出了关于半监督社区发现可检测性的理论分析。其指出对于无权网络，少量的先验信息都会破坏原始社区可检测性的属性，使可检测性阈值向着最小可能值漂移。Eaton 和 Mansbach 通过说明 spin-glass 模型是模块性函数的一个一般化形式（即无监督信息的形式）来将监督信息融入到模块化函数中 [25]。除了独立节点监督信息外，此工作还考虑如何同时融入其他形式的监督信息。Nakata 和 Murata [63] 提出了一种类似于无监督社区发现中的 louvain 算法的优化算法来提升算法 [25] 的速度。之后，为了解决前面方法由于近似计算无法得到最优解的问题，Zhang 等人 [110] 通过使用腔法（cavity method）和置信度传播（belief propagation）研究半监督社区发现。这种方法被认为是在稀疏网络上当节点个数趋近于无穷时贝叶斯最优的。从标签传播 [122] 的角度出发，算法通常假设相邻的节点之间具有相近的标签，并根据网络拓扑结构根据已知节点的标签逐渐推断其他节点的标签。Liu 等人 [47] 首次将标签传播算法用到半监督社区发现中。之后，Liu 等人 [49] 又从离散势能理论的角度验证了标签传播算法在社区发现中的应用，并将其公式化为一个最小化哈密尔顿函数的问题。此外还存在一些其他的一些基于单独节点的半监督社区发现算法，如 Wang 等人 [98] 通过约束非负矩阵分解中分解后矩阵中元素与对应社区的关系来将独立节点信息融入到基于非负矩阵分解的社区发现中。Liu 等人 [48] 考虑如果高效的将负信息引入到半监督社区发现中，所谓负信息是指节点一定不属于哪些社区的相关信息。

除了被动的接受独立节点的标签，也有一些算法开始借鉴机器学习中的主动学习来通过算法选取节点供人工标注，并将标注的信息融合到后面的学习过程中。算法选取的节点的标准可能是选取最不确定的节点，可以是对其他节点影响最大的节点，还可以是最小冗余度的节点。Leng 等人 [39] 考虑如何在标签传播的半监督社区发现算法中设计主动选择策略来提升性能。Moore 等人 [59] 基于随机块模型，从信息论度量角度出发选取能够给出关于网络中其他节点最多信息的节点进行标注，设计同时适用于同配性网络和异配性网络的主动节点

分类算法。Shi 等人 [83] 考虑在主动学习中一次批量选取多个节点时，节点之间应具有的关系，并同时考虑了多种节点选择标准对最终性能的影响。

2.4.2 基于成对关系信息的半监督社区发现算法

虽然基于独立节点监督信息是最为直接的，但是这些信息的获取通常较为困难，且在精确度上存在一定的误差。而相比于具体每个节点所属的社区，人工标注节点之间的相互关系则更为容易。通常根据节点的元数据可以初步判定两个节点之间的关系，即这两个节点是否属于同一个社区，我们将这种关系称为成对关系信息。如果两个节点属于同一个社区，我们就说他们之间具有“必连”（must-link）关系，如果两个节点不属于同一个社区，我们则说他们之间具有“必不连”（cannot-link）关系。由于网络的拓扑结构主要是由边构成的，而边通常也是说明两个节点之间相互关系的，因此这种成对的关系信息是与网络拓扑结构最为贴近的一种约束信息。

这种成对的关系信息具有信息量少、约束性强及普遍统一性三个特点。首先，由于监督信息通常是需要人工标注的，因此很难获取如网络拓扑结构那样大规模的数据。其次，这种成对约束要比网络拓扑结构中连接两个节点的链接的约束要强。网络拓扑结构中的链接只意味着相应的两个节点之间存在关系，并不意味着他们属于同一个社区。而必连约束中的两个节点则必定属于同一社区。最后，也是最重要的，几乎所有的监督信息都可以转化为成对的关系信息。如2.4.1节中所述的独立节点信息。假如我们已知节点 i 和 j 属于社区 A，节点 p 和 q 属于社区 B，那么我们确定节点 i 和 j 之间，节点 p 和 q 之间存在必连关系，且节点 i 和 q，节点 j 和 q，节点 i 和 p，节点 j 和 p 之间存在必不连关系。因此我们可以将获得的各类监督信息转化为成对的监督信息并使用基于成对关系信息的半监督社区发现算法进行融合。因此研究基于成对关系信息的半监督社区发现算法更有一般性的意义。

Steeg 等人首次从统计力学的角度出发检验成对监督信息对半监督社区发现性能的影响 [91]。其中考虑了由两个同等大小社区组成的随机网络的情况，并证明社区发现的目标函数可以被映射到一个随机二等分图上的 Ising 模型。通过证明最大化模块性密度函数 [44] 和对称非负矩阵分解的等价性，Ma 等人将成对的必连约束和必不连约束嵌入到网络拓扑结构的邻接矩阵中，然后通过进行对称非负矩阵分解求得相应的社区 [55]。Zhang 通过直接改变网络的邻接矩

阵将方法 [55] 推广到包括 Infomap 和谱聚类在内的其他社区发现方法上 [114]。这种方法相当于在具有必连约束的两个节点之间增加一条链接并去掉具有必不连约束节点之间的链接。这种方法的最大问题是忽略了我们之前提到的成对信息的强约束性。之后，Zhang 等人 [115] 通过在方法 [114] 之前增加了一步逻辑推断来更加充分的利用已经获得的成对监督信息。这个逻辑推断的过程主要是扩展了现有的成对监督信息。如果节点 i 和 j 之间存在必连关系，节点 j 和 k 之间也存在必连关系，那么节点 i 和 k 之间也存在必连关系。如果节点 i 和 j 之间存在必连关系，节点 j 和 k 之间也存在不必连关系，那么节点 i 和 k 之间也存在必不连关系。Shi 等人 [84] 通过在非负矩阵分解待优化的目标函数中同时加入对必连和必不连的约束来更加有效的利用成对监督信息。Li 等人 [42] 通过外部优化算法解决在成对监督信息中的冲突问题。Cheng 等人 [19] 通过一种启发式的算法来将必连关系和必不连关系与网络拓扑结构相结合。

2.5 性能衡量标准

由于常规的社区发现算法及基于成对监督信息的半监督社区发现算法都只是将网络划分为不同的社区，而无法如监督学习中那样确定某个节点的具体的社区标签，因此我们采用在无监督聚类算法中常用的归一化互斥信息（Normalized Mutual Information (NMI)) [86] 来度量社区检测算法在检测社区结果方面的准确度。假设 φ^a 是真实的社区结构，φ^b 是某算法检测出的社区结构，则这个算法的 NMI 可以定义为：

$$\mathrm{NMI}(\varphi^a,\varphi^b)=\frac{\sum\limits_{i=1}^{k_{gt}}\sum\limits_{j=1}^{k_{detected}}N_{ij}\log\left(\frac{N\cdot N_{ij}}{N_i^a\cdot N_j^b}\right)}{\sqrt{\left(\sum\limits_{i=1}^{k_{gt}}N_i^a\log\frac{N_i^a}{N}\right)\left(\sum\limits_{j=1}^{k_{detected}}N_j^b\log\frac{N_j^b}{N}\right)}},\tag{2.15}$$

其中 N 是节点的个数，N_{ij} 是既在真实社区 i 又在被检测到的社区 j 中的节点个数，N_i^a 是在真实社区 i 中的节点个数，N_j^b 是在被检测到的社区 j 中的节点个数。归一化互斥信息是一种比单纯统计被错分的节点个数（误判率）更具有信息量的一种度量标准。这种度量标准在如 Lancichinetti-Fortunato-Radicchi 网络（2.6.2.2节）和真实网络（2.6.1节）在内的社区大小不均匀的网络结构中更能体现社区划分算法的好坏。

此外，为了在自动化社区发现算法（第五章）中度量算法在确定社区个数方法的性能，我们还设计了如下的度量标准

$$\mathrm{diff}(k_{gt}, k_{detected}) = |k_{gt} - k_{detected}| + 1, \tag{2.16}$$

其中 k_{gt} 是真实社区的个数 $k_{detected}$ 是算法检测到的社区的个数。可见，这个函数当且仅当 $k_{gt} = k_{detected}$ 时为 1，而其他情况下均大于 1。此外，改函数的值随着 k_{gt} 和 $k_{detected}$ 之间差距的增大而增大。

2.6 常用数据集

为了对不同的社区发现模型及算法进行验证，研究人员收集并生成了很多的网络数据。根据数据的来源，可以将这些网络分为真实世界网络和人工生成网络。下面我们简要介绍本文中将用到的一些真实世界网络和两类人工生成网络。我们将主要在这些网络上验证我们提出的各类算法。

2.6.1 真实网络数据集

真实网络数据集通常是以社会科学家为代表的各个领域的研究人员通过对相关领域中的复杂系统数据进行收集、分析、建模，并抽象得到的具有典型特点的的网络数据。这里用到的真实网络的相关信息如表2.1所示。其中包括线下社交网络（如海豚网络、空手道成员网络和美国高校足球大联盟网络）、在线社交网络（美国各高校学生 Facebook 关系网络）、论文引用网络（机器学习论文引用网络和 Citeseer 网站论文引用网络）、互联网络（政治博客网络）等。我们对其中的一部分做简要的介绍。

Zachary 空手道网络（Karate）[109] 是 Zachary 收集的 19 世纪 70 年代美国某高校空手道俱乐部中 34 名成员之间的朋友关系。其中的每个成员用网络中的一个节点表示，而这些成员之间的朋友关系用网络中的边表示。在这个网络中，该俱乐部的主管和校长之间在关于提高俱乐部收费等问题上发生了分歧，这就使整个俱乐部的成员分为了倾向于校长的一部分和倾向于教练的一部分。这就导致整个网络分裂为两个分别以校长和教练为核心的两个社区。通过调查成员对教练和校长的态度可以获得他们所属的真实社区。此网络是应用最为广泛的一个验证社区发现算法的真实网络。在此网络上，社区发现算法就是要根据成员之间的朋友关系，找到成员所属的社区。

Girvan 和 Newman 根据美国 2000 年秋季常规赛季 IA 联赛高校之间比赛情况收集的的美国高校足球大联盟网络（Football）[28]。其中的 115 支球队分别用 115 个节点表示，如果两个球队之间在这个赛季有过比赛，那么他们之间就存在链接。这些球队根据他们所处的州被分为了 12 个联盟。通常这些球队同联盟内部的其他球队比赛较多，和其他联盟的球队较少比赛。社区发现算法是要根据他们之间的比赛关系判定这些球队所属的联盟。

相比于线下各类社交网络需要通过各类调查与走访来获得关系数据和标签数据，在线网络、论文引用网络和互联网络的数据相对更加容易收集。Adamic 和 Glance 收集的美国政治博客网络（Polblogs）[1] 由 1,490 个节点组成，其中每个节点代表一个政治博客，这些博客分为自由派和保守派。它们之间会引用其他博客的文章链接，通常这些引用来自相同政治观点的博客，也有一些引用是为了驳斥对手而引用的。这些博客的派别是根据 2004 年美国大选期间博文的内容进行人工标注的。社区发现算法要根据这些博客间的引用情况对其所属的派别进行判断。由 Krebs 收集的美国政治书网络（Polbooks）[68] 是根据 2004 年美国大选前后在美国亚马逊网站（www.amazon.com）上政治相关的图书销售情况构建的网络。网络中包含 105 本书，如果两本书常被共同购买，他们之间就存在一条链接。根据图书的的描述及在亚马逊网站上的评论，Newman 将这些书分为了自由派、保守派和中立派三个类别。通常相同派别的书会被同时购买。社区发现算法要根据这些图书的联合购买情况判别这些图书的类别。

随着如 Facebook 和 Twitter 等在线社交网络（SNS）的盛行，研究人员可以从中获得更大规模的网络数据。这里我们采用 Traud 等人提供的美国各高校学生 Facebook 社交网络数据 [89]，如之前的具有真实标签的网络不同，这些网络并没有真实的节点所属社区的标签，而是提供了每个学生个体包括宿舍楼、主修专业、辅修专业、入学年份、性别及所在高中等信息。Traud 等人在对数据的分解中发现这些属性中根据入学年份划分的网络具有更好的社区结构。因此我们这里也遵循这一点。

值得注意的是，虽然有很多大规模具有节点所属社区标签的大规模的网络，如 [32, 104]。但是由于在本文中我们只对非重叠半监督社区发现问题进行讨论，而这些网络通常都是针对重叠社区采集的，即部分节点具有不止一个标签。因此我们并没有采用这些网络作为验证算法的数据集。

表 2.1: 本文中用到的真实网络。

数据集	节点数	链接数	社区数	描述
Karate [109]	34	78	2	Zachary 的空手道俱乐部网络
Dolphins [54]	62	159	2	海豚社交网络
Friendship6 [100]	68	220	6	美国高校朋友关系网咯
Friendship7 [100]	68	220	7	美国高校朋友关系网络
Football [28]	115	613	12	美国高校大联盟足球俱乐部网络
Polbooks [68]	105	441	3	美国政治书网络
Polblogs [1]	1,490	16,718	2	美国政治博客网络
Cora [107]	2,708	5,429	7	机器学习领域论文引用网络
Citeseer [107]	3,312	4,732	6	Citerseer 网站上论文引用网络
Word Association [64]	5,017	29,148	-	人们通常连接使用的单词网络
Baylor [89]	12,803	679,817	7	贝勒大学学生 Facebook 网络
USF [89]	13,377	321,214	7	旧金山大学学生 Facebook 网络
Syracuse[89]	13,653	543,982	7	雪城大学学生 Facebook 网络
Temple[89]	13,686	360,795	7	天普大学学生 Facebook 网络
Northeastern[89]	13,882	381,935	7	东北大学学生 Facebook 网络
UCSD[89]	14,948	443,221	7	圣迭戈加利福尼亚大学学生 Facebook 网络
UC[89]	16,808	522,147	7	加利福尼亚大学学生 Facebook 网络
UVA[89]	17,196	789,321	7	弗吉尼亚大学学生 Facebook 网络
USC[89]	17,444	801,853	7	南加州大学学生 Facebook 网络
NYU[89]	21,679	715,715	7	纽约大学学生 Facebook 网络
UCLA[89]	20,467	747,613	7	加州大学洛杉矶分校学生 Facebook 网络
Maryland[89]	20,871	744,862	7	马里兰大学学生 Facebook 网络

2.6.2 人工网络数据集

虽然真实网络是验证社区发现算法最为直接的数据，但是由于下面两个原因，真实网络通常不足以全面验证社区发现算法的优劣。一方面，虽然现实生活中的很多系统都可以通过复杂网络来进行建模，但是获取这些网络并对其中的节点进行相应的标注通常是十分困难的，尤其是对于大规模的网络。这就导致很难有大量的真实网络可以用来进行算法验证。此外，网络的很多特性都会直接影响社区发现的性能，而对于某真实网络通常其所有特性都是固定的，而无法通过使这些特性在一定范围内变动来验证这些特性对算法性能的影响。另一方面，真实网络中的社区结构通常具有具体的功能含义，且社区结构的形成受到多方面因素的综合影响，因此这样的社区结构并不一定完全符合理论上的社区的定

义。单纯使用这些真实网络进行验证无法完全体现一个算法在某些方面的特性。

为了能更加全面合理地验证社区发现算法，研究人员根据网络及社区应满足的如幂律分布和小世界等特性，构建了很多人工网络基准测试数据生成器。一方面，这些生成器都是根据各种网络拓扑性质相关的参数随机生成网络拓扑结构，这样不仅可以得到大量用于验证的网络，还可以针对不同的目的改变网络拓扑结构特性参数来验证算法的鲁棒性。另一方面，虽然通过生成器生成的网络也同时融入了随机性特点，但是通过使用不同的随机数可以生成符合同一网络属性的多个网络，对检测结果进行统计分析可以去除随机性在其中所产生的影响。这为合理比较算法奠定了基础。下面我们介绍两类被广泛采用的人工网络，即 GN 网络和 LFR 网络

2.6.2.1 GN 网络

Girvan-Newman 网络（简称 GN 网络）[28] 是由 Girvan 和 Newman 提出的一个最为基本的验证社区发现算法性能的人工网络生成器。其所生成的每个网络由 128 个节点组成，这些节点被平均的分为 4 个社区，每个社区 32 个节点。每个节点平均有 16 条边。其中的 Z_{in} 条边随机地与同一社区中的其他节点相连，Z_{out} 条边随机的与其他社区中的节点相连，即满足 $Z_{in} + Z_{out} = 16$。为每一对 Z_{in} 和 Z_{out} 组合，我们随机生成 10 个网络。可见当 Z_{out} 较小的时候，社区的结构相对清晰，而随着 Z_{out} 的增加，网络中的社区结构变得模糊，这使得社区检测的任务变得困难。

2.6.2.2 LFR 网络

由于 GN 网络只考虑了社区内边与社区间边的比例，而没有考虑网络的其他性质，因此与现实世界中的网络有着较大的差异。为此 Lancichinetti 等人提出了一个更为复杂的人工网络生成器，Lancichinetti-Fortunato-Radicchi 网络 (即 LFR 网络)[36]。与 GN 网络相比，它具有“结点度和社区规模呈幂率分布”的异构特性，而这一点与大量真实网络相似。LFR 网络生成器可以指定社区中节点的个数，节点平均度，社区大小的分布，节点度的分布，最小及最大社区大小，社区间边的比例（μ）等。其中的社区间边的比例与 GN 网络中的平均社区间边的数目 Z_{out} 有着相同的作用，即控制社区结构的清晰程度。随着 μ 的增加，网络中的社区结构变得模糊。

在本文中，我们采用和 Lancichinetti 等人的论文 [36] 中相一致的参数设定，即节点个数为 1000，节点平均度为 20，最小社区规模为 10 或 20，最大社区规模是最小社区规模的 5 倍，混合参数从 0 到 0.8，间隔 0.05，节点度的幂次分布系数为-2，社区规模的幂次分布系数为-1。

第三章　统一的半监督社区发现

3.1　引言

在过去的几十年中，虽然有大量的社区发现算法被提出，但是他们中的大多数只考虑了网络的拓扑信息，并认为社区是一组具有类似链接性质的节点的集合。当社区内部链接数量远远大于社区间链接数量时，这类方法有着良好的表现。而当节点有大量边连向其他社区的节点时，这类方法的性能通常都会有明显的下降。实际上，社区发现并不仅仅是一个忽略节点含义的图分割问题，而是一个具有语义的聚类问题。也就是说，单纯的链接信息不足以精确的确定社区结构。这主要是由于以下的两个原因造成的：一方面，网络中社区结构的复杂性。通常情况下，网络中的社区结构存在重叠性 [101]、层级性 [81, 82] 和动态性 [6] 等复杂特征，这使得很多传统方法在社区结构不清晰的情况下无法起效。如在美国政治书网络中，网络结构中只有两个紧密连接的社区结构，即自由派图书社区和保守派图书社区。但是由于有一个中立派图书社区是由这两个紧密连接的社区的重叠部分构成的，真实的社区数量却是三个。另一方面，网络中链接信息具有稀疏性和噪音性的特点。最新的关于社区可检测性的研究表明社区内与社区间的链接数的差值存在一个相变的阈值，当低于这个阈值的时候任何单纯基于拓扑结构的方法都无法正确检测社区结构。包括谱方法在内的很多社区发现方法都只能在网络链接足够稠密的情况下能够得到令人满意的结果，而随着社区间边的数目逐渐增多，他们的性能将显著下降。

在很多实际社区发现应用场景中，除了网络拓扑信息外还存在很多的先验信息。这些先验信息通常可以分为两类：独立节点的先验信息和节点之间关系的先验信息。如在一个论文引用网络中 [107]，除了论文之间的引用关系之外，我们还可以获得包括论文关键词和词频分布在内的单节点先验信息，以及那些论文出自同一个作者这样的节点之间关系的先验信息。显然这些信息的引入，将显著的提升算法的精确性和鲁棒性，尤其是在一些社区结构不清晰的真实网络中。近来，出现了一些综合利用拓扑信息和先验信息的社区发现方法。这些方法通常被统称为半监督社区发现方法。他们中的一部分是针对独立节点先验信息设计的，另一部分是针对节点之间相互关系设计的。但是通过对先验信息的

处理，无论是独立节点先验信息还是多节点之间关系的先验信息，都可以转化为成对监督信息。针对独立节点先验信息，我们可以认为具有相同先验信息的节点之间存在必连关系，而具有不同先验信息的节点之间存在成对的必不连关系。针对多节点之间关系的先验信息，我们也可以将其拆分为多组成对的先验信息进行处理。

这里，我们首先对一些被广泛使用的基于矩阵的社区发现算法（非负矩阵分析、谱聚类及其变种）给出一种全新的统一化解释。此类方法的输入是表示网络拓扑结构的邻接矩阵，并且邻接矩阵的每行都可以认为是相应节点的特征或者表示。他们首先通过优化不同的目标函数获得这些节点在隐空间中的全新表示，并根据这些全新的表达进行不同的聚类操作。

之后，针对这个统一化解释，我们提出了一个统一的半监督社区发现框架。这个统一的半监督社区发现框架不仅可以结合先验信息和拓扑结构信息，还可以平衡这两类信息来提升社区发现的性能。基于我们的统一解释，节点是根据隐空间中的全新表达进行聚类的。因此只有具有相似隐空间表达的节点才会被分到同一个社区中。如果我们有先验信息表示一对节点属于同一个社区，那么我们应该使这两个节点有着尽量相似的隐空间表达。这里，我们通过在原有待优化的目标函数中引入隐空间图正则来将先验信息与原有拓扑信息相融合。此外，提出的框架还可以根据先验信息的可信度来平衡拓扑结构信息和先验信息对最终结果的影响。最后，由于我们采用目标函数中不同的项来刻画不同信息，我们将半监督社区发现算法当做一个全新的问题，而不单单将其看做是传统社区发现算法的前处理过程。

3.2 统一化解释

正如2.2节和2.3节所述，基于非负矩阵分解的随机块模型 [72] 和基于谱聚类的最大模块性模型 [65] 是两类进行社区发现的常用方法。虽然这两类方法的工作原理不同，但是都可以从聚类的角度出发给出一个统一的解释。首先，此类社区发现算法的输入是可以用邻接矩阵及其变种（如最大模块性模型中的模块化矩阵）表示的网络的拓扑结构。其中矩阵的每一行可以看做是对应于每个节点的特征或者表示。之后，此类方法通过优化不同的目标函数来得到一个全新的节点表示，我们称之为隐空间表示。如在基于非负矩阵分解及其变种的随机块模型中，目标函数（2.4），（5.12）和（2.6）是对原有邻接矩阵进行分解，并使

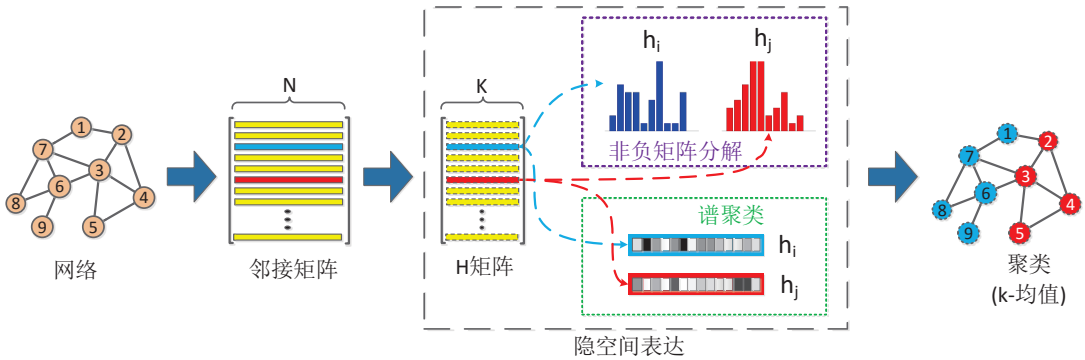


图 3.1: 对包含随机快模型及最大模块性模型在内的一些经典的社区发现算法的统一解释。

得分解之后的非负矩阵能够最大程度的重建原有邻接矩阵。而基于谱聚类的最大模块性模型是通过最大化网络的模块性函数（2.13）和（2.14），从而将社区发现问题转化为一个图切（Graph Cuts）的问题并松弛到谱聚类算法。最后，此类方法基于节点全新的隐空间表达运用不同的算法进行聚类操作，从而得到分割后的社区结构。如在非负矩阵分解中，使用的聚类算法是基于最大元素下标进行的聚类；而在谱方法中通常采用基于 k-均值的聚类方法。故此类社区发现方法可以统一的解释为“基于隐空间特征聚类算法”，如图3.1所示。除此之外，还有很多其他的社区发现的方法也可以用如上的框架进行解释，如马尔科夫聚类算法 [90] 等。

3.3　基于隐空间图正则的统一半监督框架

3.3.1　模型概述

本章，我们将基于3.2节中所述的统一的解释框架提出统一的基于隐空间图正则的半监督社区发现算法。这个半监督框架适用于所有满足统一解释框架的传统无监督社区发现算法，并可以通过结合先验信息大幅提升社区发现的性能。这个框架的主要思想如图3.2所示。由于许多社区发现算法都可以被解释为基于隐空间特征的聚类算法。容易发现属于同一社区的节点应该具有相似的隐空间表达。因此，如果我们知道节点 i 和 j 属于同一个社区，则其隐空间表达 $\mathbf{h}_i$ 和 $\mathbf{h}_j$（即 $\mathbf{H}$ 的第 i 和第 j 行）应该相似。故而，通过最小化这两个节点隐空间表达之间的差异，可以使这两个节点在聚类算法中被分到同一社区中。由于在不同

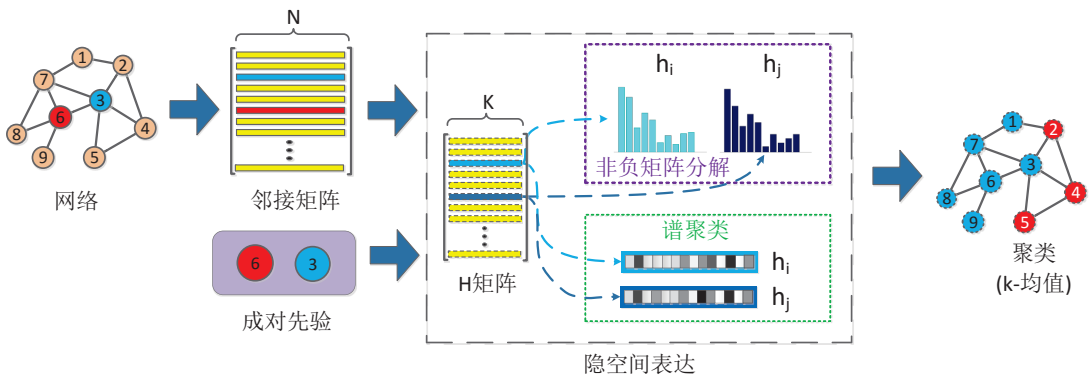


图 3.2: 能够将成对先验信息与拓扑结构信息相融合的统一半监督社区发现框架。

的算法中，隐空间的表达有着不同的含义，适合选取的度量两个向量相似程度的方法也不同。如在非负矩阵分解中，从生成模型的角度看，分解之后矩阵的每行 $\mathbf{h}_i$ 代表节点 i 属于不同的社区的概率分布，因此更加适合使用 KL 散度来度量两个分布之间的距离：

$$\mathcal{D}_{KL}(\mathbf{h}_i||\mathbf{h}_j) = \sum_{k=1}^{K}\left(h_{ik}\log\left(\frac{h_{ik}}{h_{jk}}\right) - h_{ik} + h_{jk}\right). \tag{3.1}$$

而对于谱聚类方法，由于求得的隐空间表达要进行 k-均值聚类，而常用的 k-均值聚类中常用基于欧式空间距离的度量标准，因此在度量两个向量的时候，也应该优先选择欧式距离：

$$\mathcal{D}_{LSE}(\mathbf{h}_i, \mathbf{h}_j) = ||\mathbf{h}_i - \mathbf{h}_j||_2^2 = \sum_{k=1}^{K}(h_{ik} - h_{jk})^2 \tag{3.2}$$

如果我们有先验信息表明一些节点属于同一社区，我们可以将这些信息用一系列的三元组 (i, j, o_{ij}) 进行表示。三元组 (i, j, o_{ij}) 的含义为节点 i 和 j 以可信度 o_{ij} 属于同一个社区。如果我们没有关于节点 i 和 j 的关系的先验信息，我们就令 $o_{ij} = 0$。我们可以将这一系列的三元组记为集合 $C = \{(i, j, o_{ij})\}_{i,j=1}^{N}$。

因此，我们可以将基于三元组集合 C 和欧式距离的约束公式化为：

$$\begin{aligned}\mathcal{R}_{LSE}(\{o_{ij}\}, \mathbf{H}) &= \frac{1}{2}\sum_{(i,j,o_{ij})\in C} o_{ij}\mathcal{D}_{LSE}(\mathbf{h}_i, \mathbf{h}_j) \\ &= \frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} o_{ij}||\mathbf{h}_i - \mathbf{h}_j||_2^2. \end{aligned} \tag{3.3}$$

通过定义 $\mathbf{O} = [o_{ij}] \in \mathbb{R}_+^{N\times N}$，我们可以将公式（3.3）改写为:

$$\begin{aligned}\mathcal{R}_{LSE}(\mathbf{O},\mathbf{H}) &= \sum_{i=1}^{N}\mathbf{h}_i^T\mathbf{h}_i d_{ii} - \sum_{i\neq j}\mathbf{h}_i^T\mathbf{h}_j o_{ij} \\ &= \mathrm{Tr}(\mathbf{H}^T\mathbf{D}\mathbf{H}) - \mathrm{Tr}(\mathbf{H}^T\mathbf{O}\mathbf{H}) = \mathrm{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}),\end{aligned} \tag{3.4}$$

其中 Tr(.) 代表矩阵的迹，$\mathbf{D} = [d_{ij}] \in \mathbb{R}_+^{N\times N}$ 是一个对角矩阵，其对角线上元素的值为 $\mathbf{O}$ 的对应行上所有元素值的和，即 $d_{ii} = \sum_{j=1}^{N} o_{ij}$，$\mathbf{L} = \mathbf{D} - \mathbf{O}$ 是先验信息 $\mathbf{O}$ 的图正则矩阵 [7, 12]（拉普拉斯矩阵）。类似的，基于三元组集合 C 和 KL-散度的约束可以被公式化为:

$$\begin{aligned}\mathcal{R}_{KL}(\mathbf{O},\mathbf{H}) &= \frac{1}{2}\sum_{(i,j,o_{ij})\in C} o_{ij}\Big(\mathcal{D}_{KL}(\mathbf{h}_i||\mathbf{h}_j) + \mathcal{D}_{KL}(\mathbf{h}_j||\mathbf{h}_i)\Big) \\ &= \frac{1}{2}\sum_{i,j=1}^{N}\sum_{k=1}^{K} o_{ij}\Big(h_{ik}\log(\frac{h_{ik}}{h_{jk}}) + h_{jk}\log(\frac{h_{jk}}{h_{ik}})\Big),\end{aligned} \tag{3.5}$$

其中考虑到了 KL-散度的不对称性，即 $\mathcal{D}_{KL}(\mathbf{h}_i||\mathbf{h}_j) \neq \mathcal{D}_{KL}(\mathbf{h}_j||\mathbf{h}_i)$，因此将两者进行平均。通过最小化 $\mathcal{R}_{LSE}(\mathbf{O},\mathbf{H})$ 或 $\mathcal{R}_{KL}(\mathbf{O},\mathbf{H})$，我们希望如果有先验信息指出节点 i 和 j 可能属于相同的社区（即 $o_{ij} \neq 0$），他们在隐空间中新的表达应该尽量的相似。

既然已经有了能够将成对先验信息进行刻画的图正则项 $\mathcal{R}_\beta(\mathbf{O},\mathbf{H})$（其中 $\beta \in \{LSE, KL\}$），我们就可以将其与传统的基于拓扑结构的社区发现方法的目标函数（2.4），（5.12），（2.6）和（2.14）相融合，如下公式所示:

$$\mathcal{F}_{\alpha,\beta}(\mathbf{H}|\mathbf{A},\mathbf{O}) = \mathcal{L}_\alpha(\mathbf{A},\mathbf{H}) + \lambda\mathcal{R}_\beta(\mathbf{O},\mathbf{H}), \tag{3.6}$$

其中 $\alpha \in \{LSE, KL, SYM, MOD, ADJ, LAP, NLAP\}$ 分别代表基于欧式距离非负矩阵分解、基于 KL-散度非负矩阵分解、对称非负矩阵分解、基于模块性、邻接矩阵、拉普拉斯矩阵和归一化拉普拉斯矩阵的谱聚类方法的目标函数，λ 是平衡拓扑信息和先验信息之前权重的参数。在大多数情况下 λ 的符号是正的，而对于 $\mathcal{L}_{ADJ}$ 和 $\mathcal{L}_{MOD}$，由于原始的目标函数是要最大化而图正则项是要最小化，因此 λ 取负值。为了计算的简便性，我们为公式（3.6）中的两项选择同样的距离度量标准。针对 $\alpha \in \{LSE, SYM, MOD, ADJ, LAP, NLAP\}$ 的 $\mathcal{L}_\alpha$，我们令 $\beta = LSE$；而当 $\alpha = KL$ 时，我们令 $\beta = KL$。从而，我们得到如下的一

系列目标函数:

$$\mathcal{F}_{LSE}(\mathbf{H}|\mathbf{A},\mathbf{O}) = ||\mathbf{A} - \mathbf{W}\mathbf{H}^T||_F^2 + \lambda Tr(\mathbf{H}^T\mathbf{L}\mathbf{H}), \tag{3.7}$$

$$\mathcal{F}_{SYM}(\mathbf{H}|\mathbf{A},\mathbf{O}) = ||\mathbf{A} - \mathbf{H}\mathbf{H}^T||_F^2 + \lambda Tr(\mathbf{H}^T\mathbf{L}\mathbf{H}), \tag{3.8}$$

$$\begin{aligned}\mathcal{F}_{KL}(\mathbf{H}|\mathbf{A},\mathbf{O}) &= \sum_{i=1}^{N}\sum_{j=1}^{N}\Big(a_{ij}\log(\frac{a_{ij}}{\sum_{k=1}^{K} w_{ik}h_{jk}}) - a_{ij} + \sum_{k=1}^{K} w_{ik}h_{jk}\Big) \\ &+ \frac{\lambda}{2}\sum_{i=1}^{N}\sum_{j=1}^{N}\sum_{k=1}^{K}\Big(h_{ik}\log(\frac{h_{ik}}{h_{jk}}) + h_{jk}\log(\frac{h_{jk}}{h_{ik}})\Big)o_{ij},\end{aligned} \tag{3.9}$$

$$\mathcal{F}_{MOD}(\mathbf{H}|\mathbf{A},\mathbf{O}) = -\operatorname{Tr}(\mathbf{H}^T\mathbf{B}\mathbf{H}) + \lambda\operatorname{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}), \tag{3.10}$$

$$\mathcal{F}_{LAP}(\mathbf{H}|\mathbf{A},\mathbf{O}) = \operatorname{Tr}(\mathbf{H}^T(\mathbf{D}-\mathbf{A})\mathbf{H}) + \lambda\operatorname{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}). \tag{3.11}$$

基于邻接矩阵的半监督谱聚类方法的目标函数 $\mathcal{F}_{ADJ}$ 和基于归一化拉普拉斯矩阵的目标函数 $\mathcal{F}_{NLAP}$ 与基于模块性矩阵的目标函数 $\mathcal{F}_{MOD}$ 和基于拉普拉斯矩阵的目标函数 $\mathcal{F}_{LAP}$ 除去用 $\mathbf{A}$ 和 $\mathbf{I}-\mathbf{D}^{-1}\mathbf{A}$ 替换 $\mathbf{B}$ 和 $\mathbf{D}-\mathbf{A}$ 外完全一致. 关于求解这些目标函数的细节，我们想在下一节中进行讨论。

3.3.2 优化算法

在本节，我们将具体讨论优化问题（3.7）- （3.11）的求解方法。

3.3.2.1 半监督谱方法的优化算法

根据公式（3.10）和（3.11），由于所有的四个基于谱聚类的半监督社区发现方法 $\mathcal{F}_\alpha(\mathbf{H}|\mathbf{A},\mathbf{O}), \alpha \in \{MOD, ADJ, LAP, NLAP\}$ 都可以被重写为如下的统一形式

$$\begin{aligned}\mathcal{F}_{SC}(\mathbf{H}|\mathbf{A},\mathbf{O}) &= \operatorname{Tr}(\mathbf{H}^T\mathbf{G}\mathbf{H}) + \lambda\operatorname{Tr}(\mathbf{H}^T\mathbf{L}\mathbf{H}) \\ &= \operatorname{Tr}(\mathbf{H}^T(\mathbf{G}+\lambda'\mathbf{L})\mathbf{H}),\end{aligned} \tag{3.12}$$

其中 $\lambda' = \frac{\lambda}{N}$，并且当 $\alpha = MOD, ADJ, LAP, NLAP$ 时，$\mathbf{G}$ 分别等于 $-\mathbf{B}$，$-\mathbf{A}$，$\mathbf{D}-\mathbf{A}$ 和 $\mathbf{I}-\mathbf{D}^{-1}\mathbf{A}$。这个统一的目标可以与传统的基于谱聚类的最大模块性模型一致地通过寻找相应矩阵对应于最小特征值的特征向量来进行最优化求解。

3.3.2.2　半监督非负矩阵分解的优化算法

由于和原始的非负矩阵分解中的情况一样，公式 (3.7)，(3.8) 和 (3.9) 中的目标函数 $\mathcal{F}_{LSE}$，$\mathcal{F}_{SYM}$ 和 $\mathcal{F}_{KL}$ 并不是同时对 $\mathbf{H}$ 和 $\mathbf{W}$ 是凸的，因此要找到能够获得全局最优解的优化算法比较困难。这里，我们采用论文 [38] 中提出的一种能够获得局部最优解的迭代优化算法。

我们以公式（3.7）中的 $\mathcal{F}_{LSE}$ 为例来说明如何最小化目标函数。由于矩阵迹和 Frobenius 模的如下一些性质 $\mathrm{Tr}(\mathbf{A}) = \mathrm{Tr}(\mathbf{A}^T)$，$\mathrm{Tr}(\mathbf{AB}) = \mathrm{Tr}(\mathbf{BA})$ 和 $||\mathbf{A}||_F^2 = \mathrm{Tr}(\mathbf{AA}^T)$，我们可以将 $\mathcal{F}_{LSE}$ 重写为：

$$\begin{aligned}\mathcal{F}_{LSE}(\mathbf{H}|\mathbf{A},\mathbf{O}) &= \mathrm{Tr}((\mathbf{A}-\mathbf{WH}^T)(\mathbf{A}-\mathbf{WH}^T)^T) + \lambda\,\mathrm{Tr}(\mathbf{H}^T\mathbf{LH}) \\ &= \mathrm{Tr}(\mathbf{AA}^T) + \mathrm{Tr}(\mathbf{WH}^T\mathbf{HW}^T) - 2\,\mathrm{Tr}(\mathbf{AHW}^T) \\ &\quad +\lambda\,\mathrm{Tr}(\mathbf{H}^T\mathbf{LH}). \end{aligned} \tag{3.13}$$

通过分别为不等式约束 $\mathbf{W} = [w_{ij}] \geq 0$ and $\mathbf{H} = [h_{ij}] \geq 0$ 引入拉格朗日乘子 $= [\psi_{ij}] \in \mathbb{R}^{N\times K}$ 和 $= [\phi_{ij}] \in \mathbb{R}^{N\times K}$，我们可以将其拉格朗日形式写为

$$\begin{aligned}\mathcal{L}_{LSE} &= \mathrm{Tr}(\mathbf{AA}^T) + \mathrm{Tr}(\mathbf{WH}^T\mathbf{HW}^T) - 2\,\mathrm{Tr}(\mathbf{AHW}^T) \\ &+ \lambda\,\mathrm{Tr}(\mathbf{H}^T\mathbf{LH}) + \mathrm{Tr}(\Psi\mathbf{W}^T) + \mathrm{Tr}(\Phi\mathbf{H}^T). \end{aligned} \tag{3.14}$$

为了得到能使 $\mathcal{L}_{LSE}$ 最小的 $\mathbf{W}$ 和 $\mathbf{H}$，我们迭代的固定一个变量优化另一个变量。由于 $\mathcal{L}_{LSE}$ 对 $\mathbf{W}$ 和 $\mathbf{H}$ 都是可微的，我们得到 $\mathcal{L}_{LSE}$ 关于 $\mathbf{W}$ 和 $\mathbf{H}$ 的偏导数

$$\begin{aligned}\frac{\partial\mathcal{L}_{LSE}}{\partial\mathbf{W}} &= -2\mathbf{AH} + 2\mathbf{WH}^T\mathbf{H} + \Psi, \\ \frac{\partial\mathcal{L}_{LSE}}{\partial\mathbf{H}} &= -2\mathbf{A}^T\mathbf{W} + 2\mathbf{HW}^T\mathbf{W} + 2\lambda\mathbf{LH} + \Phi. \end{aligned} \tag{3.15}$$

通过组合偏导数等于零和 Karush–Kuhn–Tucke（KKT）条件

$$\psi_{ik}w_{ik} = 0, \phi_{jk}h_{jk} = 0$$

我们得到 w_{ik} 和 h_{jk} 的解为：

$$\begin{aligned}-(\mathbf{AH})_{ik}w_{ik} + (\mathbf{WH}^T\mathbf{H})_{ik}w_{ik} = 0, \\ -(\mathbf{A}^T\mathbf{W})_{jk}h_{jk} + (\mathbf{HW}^T\mathbf{W})_{jk}h_{jk} + \lambda(\mathbf{LH})_{jk}h_{jk} = 0. \end{aligned} \tag{3.16}$$

最后，我们得到如下的迭代更新规则

$$w_{ik} \leftarrow w_{ik}\frac{(\mathbf{AH})_{ik}}{(\mathbf{WH}^T\mathbf{H})_{ik}}, h_{jk} \leftarrow h_{jk}\frac{(\mathbf{A}^T\mathbf{W}+\lambda\mathbf{OH})_{jk}}{(\mathbf{HW}^T\mathbf{W}+\lambda\mathbf{DH})_{jk}}. \tag{3.17}$$

当 $\lambda = 0$ 时，更新规则（3.17) 退化为标准非负矩阵分解的更新规则。

类似的，我们得到最小化 $\mathcal{F}_{SYM}$ 的更新规则

$$h_{ik} \leftarrow h_{ik}\frac{(\mathbf{AH}+\lambda''\mathbf{OH})_{ik}}{(\mathbf{HH}^T\mathbf{H}+\lambda'\mathbf{DH})_{ik}}, \tag{3.18}$$

其中，为了保证公式的一致性，我们令 $\lambda'' = 2\lambda$。

最后，我们给出最小化公式（3.9）中 $\mathcal{F}_{KL}$ 的更新规则

$$\begin{aligned} w_{ik} &\leftarrow w_{ik}\frac{\sum_j\left(a_{ij}h_{jk}/\sum_k w_{ik}h_{jk}\right)}{\sum_j h_{jk}}, \\ \mathbf{h}_k &\leftarrow \left(\sum_i w_{ik}\mathbf{I}+\lambda\mathbf{L}\right)^{-1}\widehat{\mathbf{h}}_k, \end{aligned} \tag{3.19}$$

其中 $\mathbf{h}_k$ 是 $\mathbf{H}$ 的第 k 列，$\mathbf{I}$ 是一个 $N \times N$ 的单位阵，并且

$$\widehat{\mathbf{h}}_k = \begin{bmatrix} h_{1k}\sum_i(a_{i1}w_{ik}/\sum_k w_{ik}h_{1k}) \\ h_{2k}\sum_i(a_{i2}w_{ik}/\sum_k w_{ik}h_{2k}) \\ \vdots \\ h_{Nk}\sum_i(a_{iN}w_{ik}/\sum_k w_{ik}h_{Nk}) \end{bmatrix}. \tag{3.20}$$

其中（3.17)，（3.18 ）和（3.19）中的更新规则可以分别得到公式（3.7）中的目标函数 $\mathcal{F}_{LSE}(\mathbf{H}|\mathbf{A},\mathbf{O})$，公式（3.8 ）中的目标函数 $\mathcal{F}_{SYM}(\mathbf{H}|\mathbf{A},\mathbf{O})$ 和公式（3.9）中目标函数 $\mathcal{F}_{KL}(\mathbf{H}|\mathbf{A},\mathbf{O})$ 的局部最优解。

3.3.3 时间复杂度分析

本节将具体分析在我们提出的框架下的各类半监督社区发现算法(3.17)，(3.18) 和（3.19）的时间复杂度.

3.3.3.1 半监督谱方法的时间复杂度

如前所述，求解半监督社区发现谱方法和求解常规无监督的谱方法的优化算法是一样的，都需要通过特征值分解。理论上，求解矩阵特征值的算法与矩阵乘

法的时间复杂度的上限是一致的，即为 $O(N^3)$，其中 N 是网络中节点的个数。实际上，Coppersmith 和 Winograd[20] 将其复杂度降低到 $O(N^{2.376})$。除此之外，由于邻接矩阵的稀疏性和对称性，我们还可以采用很多现有的软件包进行大规模特征值问题的求解，如 ARPACK（ARnoldi PACKage）[20] 等。

3.3.3.2　半监督非负矩阵分解的时间复杂度

这里我们将分别讨论基于欧氏距离和基于 KL 散度的半监督非负矩阵分解算法的时间复杂度，即公式（3.7）中的 $\mathcal{F}_{LSE}$ 和公式（3.9）中的 $\mathcal{F}_{KL}$。而对于半监督对称矩阵分解算法 $\mathcal{F}_{SYM}$，其时间复杂度与 $\mathcal{F}_{LSE}$ 一致。

对于基于欧氏距离的半监督非负矩阵分析算法，如公式（3.17）和（3.18）所示，如果考虑社区个数远远小于网络中节点个数 $K \ll N$，迭代中的每一步需要 $O(N^2K)$ 的浮点运算。因此，我们的统一半监督框架没有增加标准的基于欧式距离的非负矩阵分解的实际复杂度。当进一步考虑到邻接矩阵的稀疏性时，每步迭代的时间复杂度降到 $O((NK+M+R)K)$，其中 N，M 和 R 分别是其中节点数、链接数和成对必连先验的个数。如果我们加入的必连先验的数目 R 与网络中边的数目 M 是同一个数量级，我们提出的半监督框架和原始的稀疏非负矩阵分解有着相同的复杂度。

如公式（3.19）所示，基于 KL 散度的半监督非负矩阵分解的每步迭代中需要计算一个大小为 $N \times N$ 的矩阵的逆，而其需要 $O(N^3)$ 次浮点运算。但是在实际应用中，我们只需要求解线性方程组 $\mathbf{Ax}=\mathbf{b}$ 的解，而不需要计算矩阵 $\mathbf{A}$ 的逆。由于 $\sum_i w_{ik}\mathbf{I}+\lambda\mathbf{L}$ 是对称正定矩阵，我们采用共轭梯度下降法 [30] 来求解这个线性方程组。由于我们通常加入的先验信息的比例比较小，因此需要进行求解线性代数方程组的系数矩阵是稀疏的。假设矩阵 $\sum_i w_{ik}\mathbf{I}+\lambda\mathbf{L}$ 的每列平均有 p 个非零元素，共轭梯度法需要 q 次迭代收敛，我们需要 $O(q(p+4)N)$ 次浮点运算得到线性方程组的解。通常情况下，共轭梯度下降法收敛只需要很少几次迭代。因此在我们的实验中，我们设定最大的迭代次数为 20。由于需要求解 K 个线性方程组，每步的整体计算复杂度为：

$$O\Big(N^2K+q(p+4)NK\Big)=O\Big((N+q(p+4))NK\Big). \tag{3.21}$$

从前面的复杂度分析，我们可以发现我们提出的半监督社区发现框架并没有显著提升原有无监督问题的计算复杂度。此外基于欧氏距离的算法，即 $\mathcal{F}_{LSE}$

和 $\mathcal{F}_{SYM}$ 的时间复杂度与网络规模 N 呈近似线性关系，因此更适合于大规模网络。除此之外，由于矩阵乘法和非负矩阵分解都有相应的并行化算法被提出，我们还可以通过借助并行计算 [22] 或分布式计算 [46] 来使我们的框架适用于更大规模的网络。

3.3.4 模型选择

模型选择是社区发现的一个重要问题，它是要确定网络中社区的精确个数 K。有很多现成的模型选择的方法可供我们选择，如共识聚类 [43, 58]，特征值间隔 [67, 69]，交叉验证和贝叶斯信息准则 [31]。但是为了使我们的框架更加的统一，和 [68, 114, 115] 一样，我们采用被广泛使用的模块性 Q 函数作为确定社区个数的准则。这种策略的最大好处在于它与特定的社区发现算法是无关的。对于我们不知道社区个数的网络，我们可以选择能最大化模块性 Q 的那个社区个数 K。在下面的实验中，我们假设社区的个数都是提前已知的。

3.4 实验结果与分析

为了验证我们提出的半监督社区发现框架，我们在两个常用的人工网络和一些广泛使用的真实网络上进行了实验。并将这个框架与基于更新拓扑结构信息的转化框架进行比较。为了充分说明我们提出的框架的广泛适用性，我们将其应用于3.3节中提到的所有方法，即基于欧氏距离的非负矩阵分解 (NMF_LSE), 基于 KL 散度的非负矩阵分解 (NMF_KL), 对称非负矩阵分解 (SNMF), 基于邻接矩阵的谱聚类 (SC_ADJ), 基于标准 Laplacian 矩阵的谱聚类 (SC_LAP) 及基于归一化 Laplacian 矩阵的谱聚类 (SC_NLAP)。

所有的实验都是在一台配备 Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz. 处理器和 8G 内存的个人电脑上进行的。在一个具有 N 个节点的无向网络中，总共有 $N(N-1)/2$ 对节点关系，而在其中能够标明两个节点属于同一个社区的关系共有：

$$N_{pairs} = \sum_{k=1}^{K} N_k(N_k - 1)/2, \tag{3.22}$$

其中 K 是社区的个数，N_k 是第 k 个社区中节点的个数。本文中用的成对监督信息的比例就是基于 N_{pairs} 计算的。由于算法的性能不仅与加入的成对监督信

息的数量有关系，还与具体加入的成对监督信息有关，因此对于每个实验我们随机选取 10 组监督信息，并将这 10 组数据所得结果的平均值作为最终结果。

3.4.1　人工网络

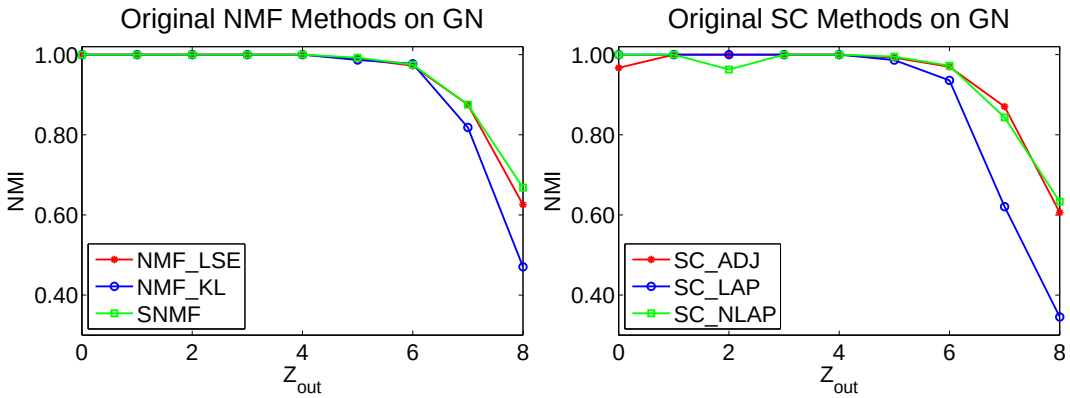


图 3.3: 在 GN 网络上，原始社区发现算法的性能随着社区间边数目 Z_{out} 的变化情况。

我们首先在 GN 人工网络上验证算法的有效性。如2.6.2.1节中所述，在 GN 网络中，随着 Z_{out} 的增加，网络的社区结构变得模糊，且检测社区变得十分有挑战性。在图 3.3中我们给出了在不同 Z_{out} 值的时候，各种非复矩阵分解和谱聚类算法的性能。容易发现所有的方法在 $Z_{out} \leq 6$ 的时候可以达到非常好的效果，但当 $Z_{out} > 6$ 时，这些方法的性能都会显著下降。且当 $Z_{out} = 8$ 时最差的性能只有 0.35，而最好的性能也不会超过 0.7。这就意味着当 $Z_{out} > 6$ 时，单纯的网络拓扑结构不足以进行精确的社区检测，从而需要先验信息的辅助。因此，在 GN 网络的相关实验中，我们主要考虑半监督社区发现算法在 Z_{out} 为 7 或者 8 的网络上的性能。

为了验证提出的框架，我们首先固定平衡参数 $\lambda = 1$，并且当我们有先验信息节点指出 i 和 j 属于同一社区时，令 $o_{ij} = 1$。我们在图3.4中给出了我们框架应用到不同方法时的性能。上面两幅图是基于非负矩阵分解方法，下面两幅图是基于谱聚类方法。左边两幅图是在 $Z_{out} = 7$ 的 GN 网络上的结果，右边两幅图是在 $Z_{out} = 8$ 的 GN 网络上的结果。。我们可以发现：虽然不同方法有着不同的增长趋势，所有方法的 NMI 都会一致地随着所使用的先验信息的数量的增加而增加，且当先验信息足够多时，所有方法的 NMI 都可以达到 1，即完全正确

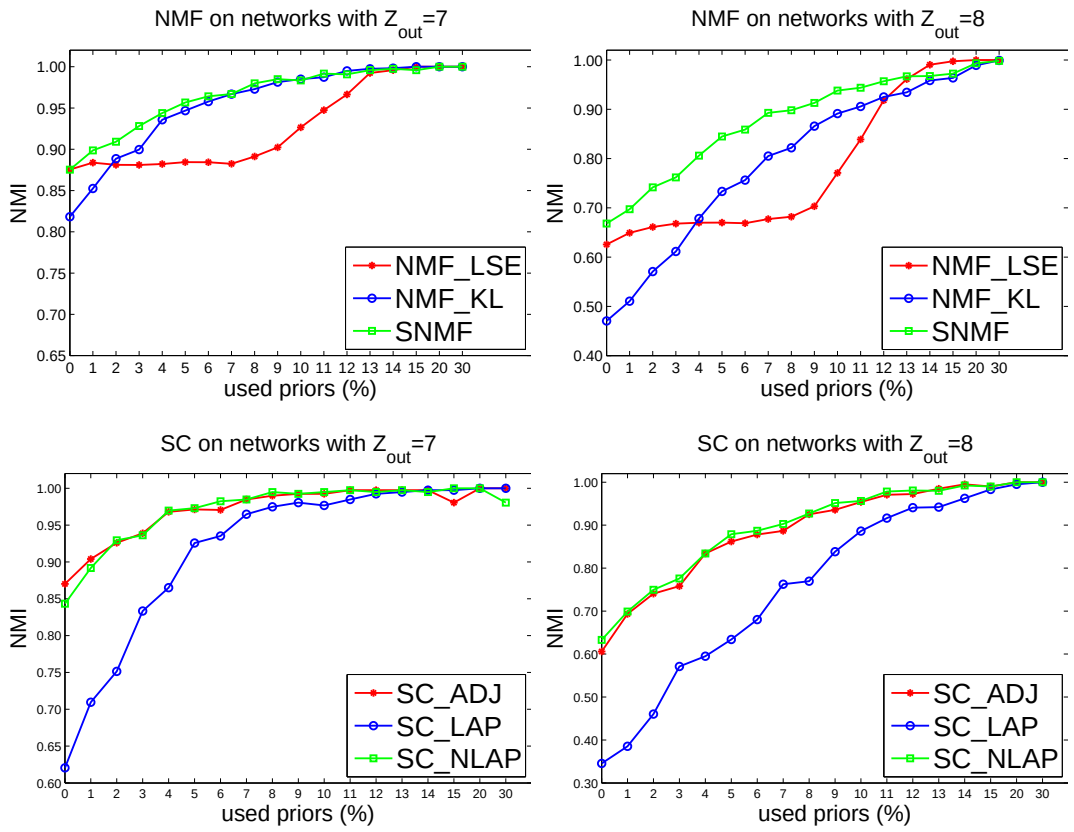


图 3.4: 在 GN 网络上，我们框架的性能随着加入先验信息的比例的变化情况。

检测社区结构。这充分验证了我们算法的有效性。此外，对比左侧和右侧的图可以发现，在加入相同比例的先验信息后，在社区结构模糊（即 $Z_{out} = 8$）的网络中，性能提升要更加显著。如当在基于 KL 散度的非负矩阵分解方法中融入 10% 成对先验信息时，在 $Z_{out} = 7$ 的网络上平均可以提升 15%，而在 $Z_{out} = 8$ 的网络上平均却可以提升近 40%。这也与我们方法的初衷相吻合，即通过嵌入成对先验信息来提升在具有不清晰社区结构的网络上社区发现的性能。

我们其次在更加复杂的 LFR 网络上验证算法的有效性。如在 GN 网络上的实验一样，我们这里固定平衡参数 $\lambda = 1$，可信度参数 $o_{ij} = 1$。我们从社区规模和网络结构清晰程度两个角度对框架的性能进行分析。

在社区结构清晰度方面，LFR 网络中的混合参数 μ 起着和 GN 网络中 Z_{out} 相同的角色，及当 μ 变大时社区结构变得模糊，社区发现算法变得困难，我们这里分别考虑 $\mu = 0.7$ 和 $\mu = 0.8$ 时的情况，相应的结果分别在图3.5和3.6中给

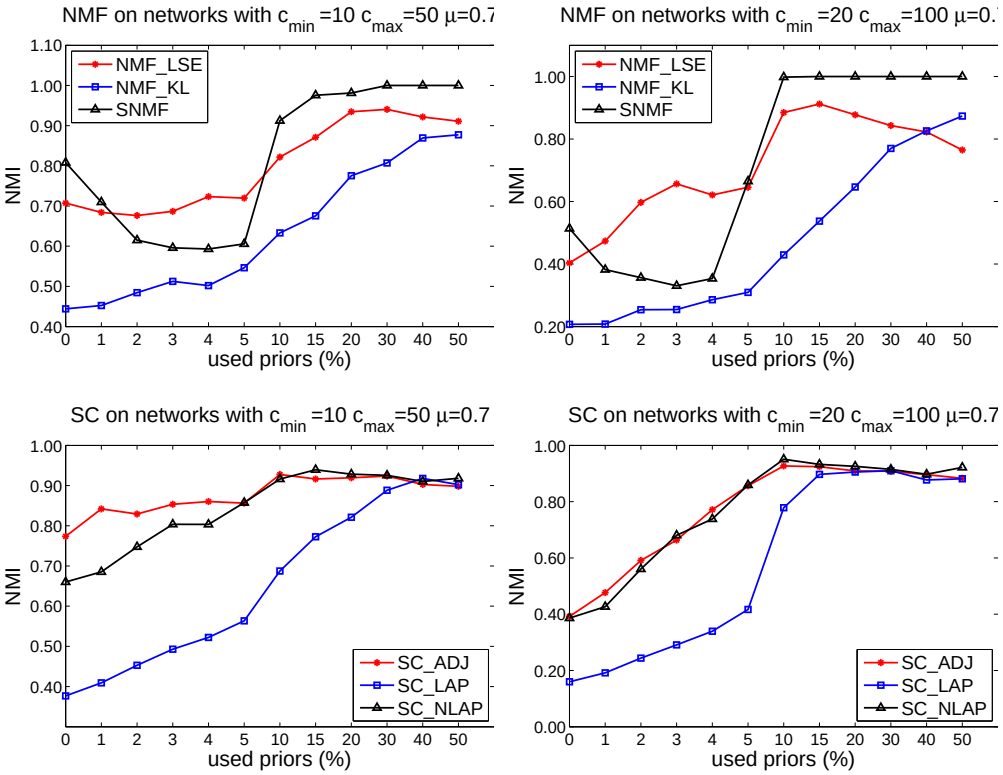


图 3.5: 在社区结构相对清晰（$\mu = 0.7$）的 LFR 网络上，我们框架的性能随着加入先验信息的比例的变化情况。

出。在社区规模上，我们分别考虑小规模社区（社区大小介于 10 至 50 之间）和大规模社区（社区大小介于 20 至 100）。第一行是基于非负矩阵分解的算法的性能，第二行是基于谱聚类算法的性能。第一列是在社区规模较小的网络（10-50）上的实验结果，第二列是在社区规模较大的网络（20-100）上的实验结果。

我们可以发现三种不同的谱聚类方法有着相似的性能曲线，即随着先验信息的增加 NMI 持续的提升。

而三类不同的非负矩阵分解算法有着各自的优缺点。虽然 NMF_KL 在不加入先验信息的时候性能最差，但是由于基于 KL 散度的图正则充分考虑到了对称性的特征，因此随着先验信息的加入，性能可以持续的提升。NMF_LSE 有着较好的初始性能，但是当加入监督信息的比例超过 30% 的时候，性能会出现明显的下降，这主要是由于在加入大量先验信息后基于欧式空间的图正则没有很好的考虑监督信息的对称性的问题，导致分解后的两个矩阵不能很好的平衡

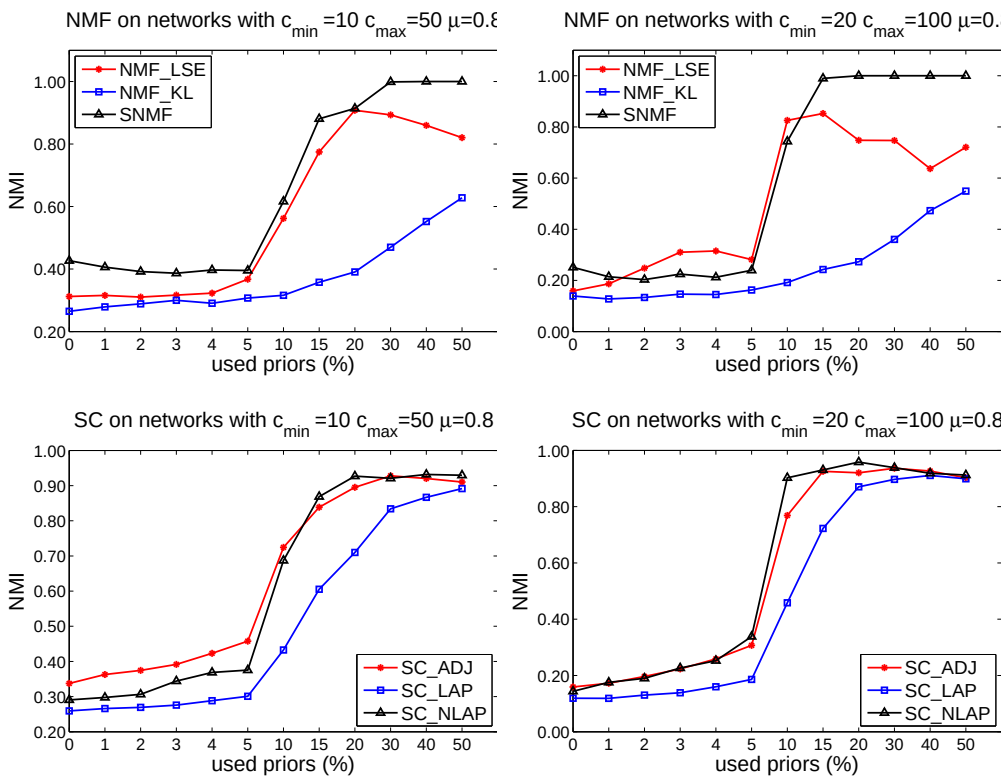


图 3.6: 在社区结构相对模糊（$\mu = 0.8$）LFR 网络上，我们框架的性能随着加入先验信息的比例的变化情况。

信息。这里表现最好的是 SNMF，它有着很高的初始性能，并可以再加入少量监督信息后（4% or 5%）NMI 达到 1。但是这种方法由于在矩阵分解的时候考虑到了对称性，而欧式空间图正则没有考虑对称性，在加入信息非常少的时候性能会出现一定程度的下降，这主要是由于破坏了对称分解的性质。此外，和在 GN 网络中类似，我们提出的框架在社区结构模糊的网络中有着更加良好的表现。针对不同社区规模，我们发现框架整体在社区规模相对较大的网络中有着更加突出的表现。即图3.5和3.6中第二列的性能提升要高于第一列的性能提升。这主要归结于当社区规模较大时，能表示两个节点属于同一社区的关系信息数量 N_{pairs} 相对较多。这样，当加入同样比例的监督信息时，如此加入的监督信息的数量更多。从实验结果我们可以发现 SNMF 方法更适用于有充足先验信息的情况，而最为基本的 NMF_LSE 更适用于先验信息有限的情况。

为了说明我们提出的框架的优越性，我们将其下的各种方法与 Zhang [114]

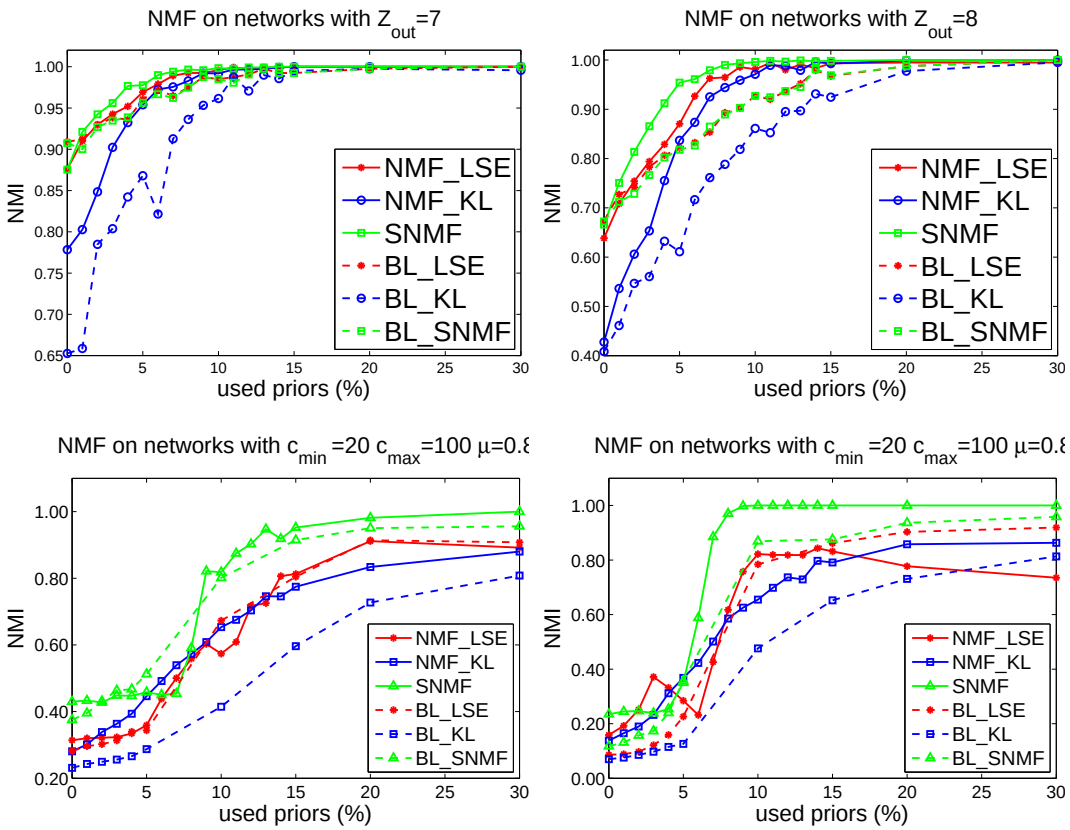


图 3.7: 与其他半监督社区发现算法的性能比较。

提出的框架下的相应方法相比较。Zhang [114] 的方法首先通过连接两个必连节点来直接改变网络的拓扑结构，然后使用传统的方法进行求解。由于这两种框架在半监督谱聚类算法上得到相同的优化问题，我们仅比较他们在各类非负矩阵分解方法上的性能。在所有的网络上，我们令平衡参数 $\lambda = 10$。图3.7分别展示了在 GN 网络（第一行）和 LFR 网络（第二行）上的结果。我们可以发现基于我们框架的方法除了 NMF_LSE 在 LFR 网路外，性能上显著超越了基于 Zhang 的框架的方法（BL_LSE, BL_KL 和 BL_SNMF）。这进一步说明了我们算法在利用监督消息上是十分高效的。

3.4.2　错误先验影响

通常情况下，先验信息被认为是来自与人工标注的正确标签。但是在实际情况下，错误的监督信息也是难免的。为了说明我们提出的框架的鲁棒性，我们研究

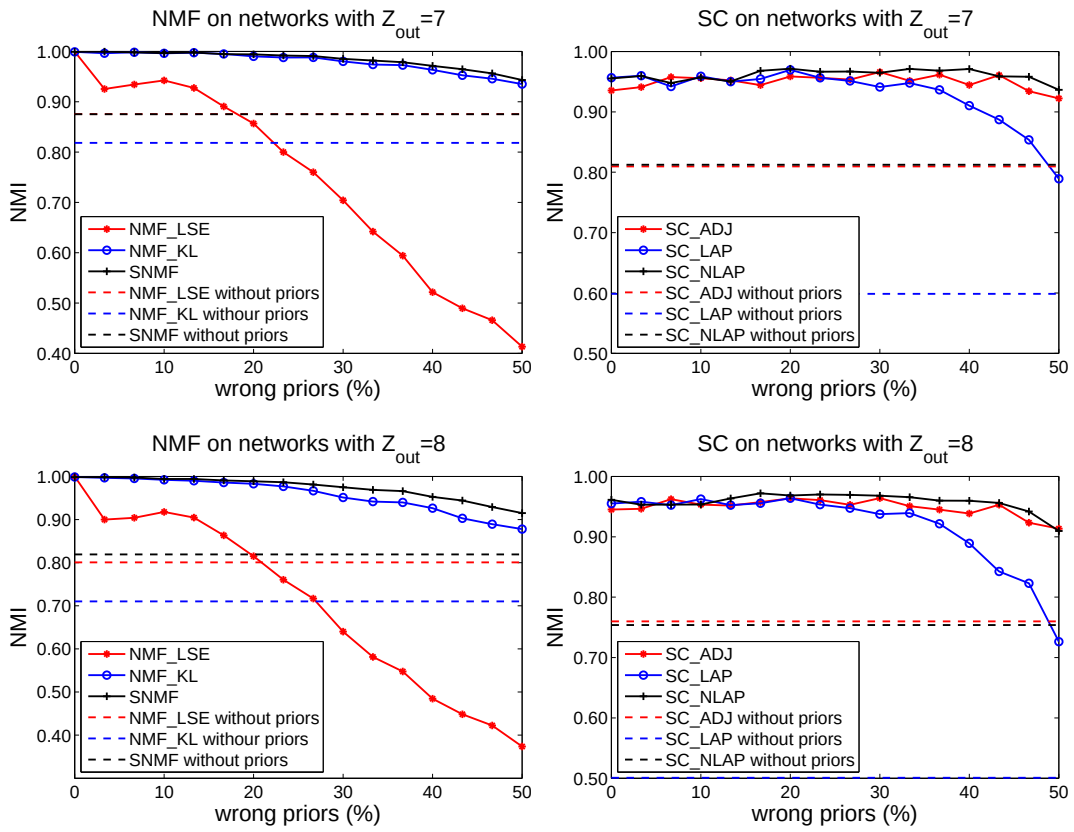


图 3.8: 错误先验信息对性能的影响。

错误先验信息对性能的影响。从图3.4中我们可以发现，大多数的方法可以在加入 30% 正确先验信息时得到令人满意的结果。因此我们在算法中加入 30% 的先验信息，但是让其中的一部分是错误的先验信息。我们令错误的先验信息的比例从 0% 到 50%。在图3.8中，我们展示了错误先验信息对性能的影响。水平虚线代表各种方法在无先验信息时的性能。图片中的其他曲线表明在加入 30% 先验信息时候各种方法的表现，其中横坐标表示这 30% 的先验信息中错误信息的比例。通过融入 50% 错误先验信息，即 15% 的正确先验信息和 15% 的错误先验信息，绝大多数的方法（NMF_KL, SNMF, SC_LAP and SC_NLAP）只会在性能上降低 10% 左右，并且结果仍然高于原始的不加入任何先验信息的情况。这充分说明了我们的方法对于噪音和错误先验的鲁棒性。其中 NMF_LSE 方法对错误先验信息相对敏感的一个可能的原因在于他无论是非负分矩阵分解的部分还是对必连的正则约束都没有考虑到网络的对称性，且只将约束施加在

分解后的一个矩阵上。

3.4.3　参数设置

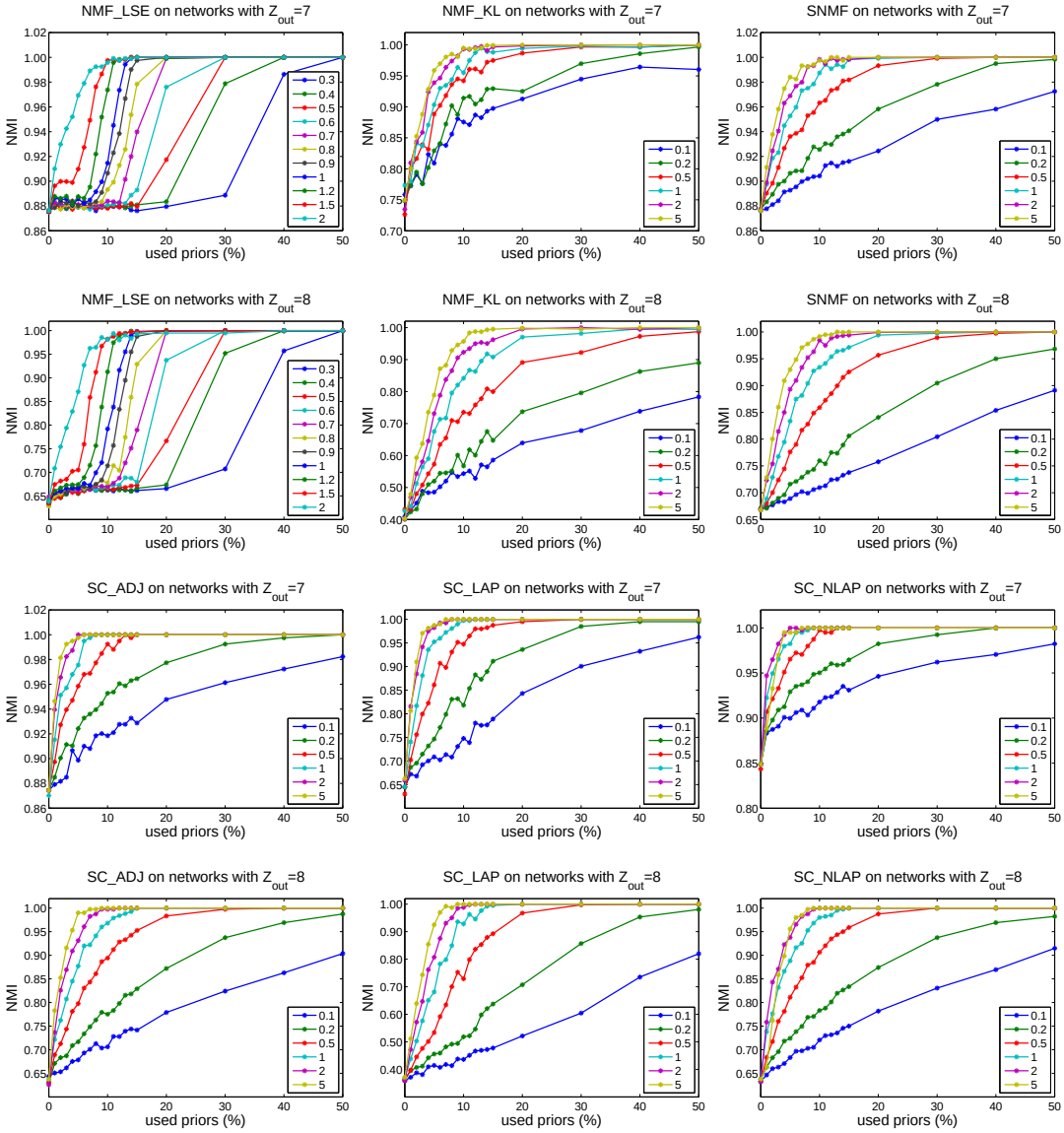


图 3.9: 在 GN 网络上，基于我们提出的框架的各类算法的性能随着平衡参数 λ 的变化情况。

为了说明平衡参数的作用以及讨论如何在实际应用中确定它，这里我们评估在公式（3.6）中，参数 λ 对性能的影响。在图3.9中，我们展示了在 GN 网络

中基于我们框架的非负矩阵分解和谱聚类算法的性能随着 λ 从 0.1 变到 10 的变化情况。其中前两行是各类基于非负矩阵分解算法的性能，后两行是各类基于谱聚类方法的性能。此外第一行第三行是在 $Z_{out} = 7$ 的 GN 网络上的性能结果，第二行第四行是在 $Z_{out} = 8$ 的 GN 网络上的性能结果。每条曲线代表框架性能在一个特定 λ 值是随加入先验信息的变化情况。为了更好的展示，我们只选取了一小部分 λ 的值。由于我们加入的先验信息是完全正确的，在加入固定比例的先验信息时，随着在框架中增加先验信息的权重（即增大 λ 值），算法的性能将持续提升，即在加入同等数量先验信息时，权重大的平衡参数将使算法性能有更大的提升。以在 $Z_{out} = 8$ 的 GN 网络上的 SNMF 为例，当加入 15% 先验信息的时候，当我们令 λ 分别为 0.1，0.2，0.5，1，2 和 5 时，算法分别可以达到 0.73，0.81，0.93，0.97，0.99 和 1。此外，从另一个角度，为了使算法的 NMI 可以达到 1，当 λ 为 5，2，1 和 0.5 时，我们需要加入的先验信息的分别是 13%，30%，40% 和 50%。

此外，我们也将分析在 LFR 网络中平衡参数 λ 对性能的影响。在图3.10我们给出了在混合参数 $\mu = 0.8$ 社区大小介于 10 和 50 之间的 LFR 网络上各种算法受平衡参数 λ 的整体影响情况。其中每条曲线代表某一具体方法的性能在一个特定 λ 值时随加入先验信息的变化情况。并且我们针对参数 λ 对 SNMF 算法的影响给出了更加详细的分析，结果如图3.11所示。其中的参数设定是将图3.10中的社区大小变为从 20 至 100，且混合参数 μ 分别为 0.7，0.75 和 0.8。我们之所以选择 SNMF 作为例子进行详细分析是因为我们在图3.10中我们发现 SNMF 的性能受参数 λ 影响最为明显。从这些结果中我们可以发现如下的一些规律：

1. 如图3.10所示，参数 λ 对绝大多数方法的影响是正面的。这也就是说随着 λ 的增大，绝大多数方法可以在使用相同数量先验信息时得到更好的性能，或者为了达到同样的性能可以使用更少的先验信息。

2. 如果网络结构不够清晰，如图3.11中最右侧的子图所代表的 $\mu = 0.8$ 的 LFR 网络，增加 λ 可以显著并一致的提升性能。这也意味着先验信息在社区结构模糊的网络上的社区发现起了更加重要的作用。因此我们可以适当的选择较大的平衡参数 λ 来凸显框架在这些网络上的优势。

3. 在具有相对清晰社区有结构的网络上，如图3.11中最左侧的图代表的 $\mu =$

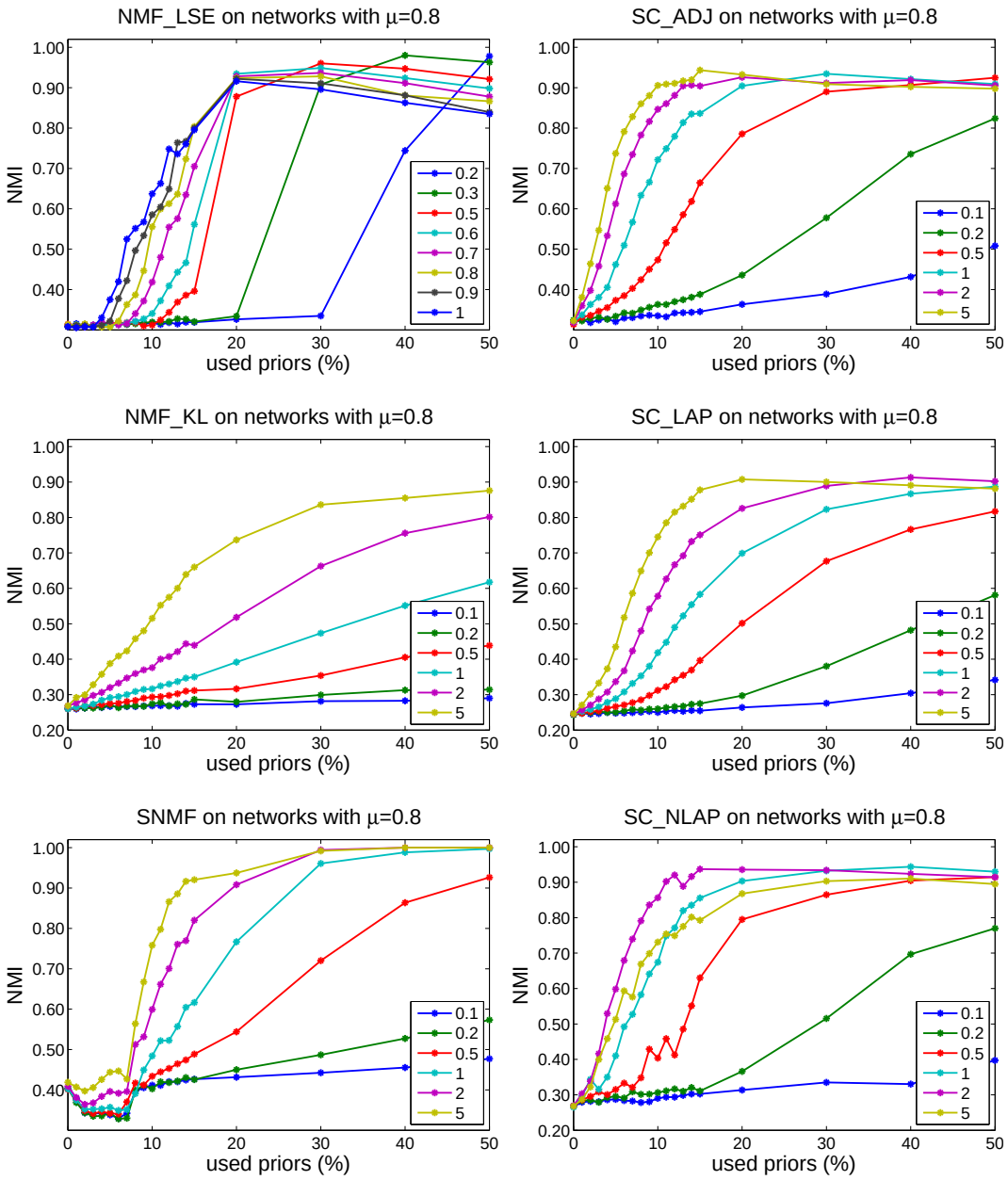


图 3.10: 在混合参数 $\mu = 0.8$ 的 LFR 网络上，基于我们提出的框架的各类算法的性能随着平衡参数 λ 的变化情况。

0.7 的 LFR 网络，SNMF 方法的性能对参数 λ 十分的敏感。但是这个敏感度将随着网络拓扑结构清晰度的降低而降低，如在图3.11中右侧的图代表的 $\mu = 0.8$ 的 LFR 网络。

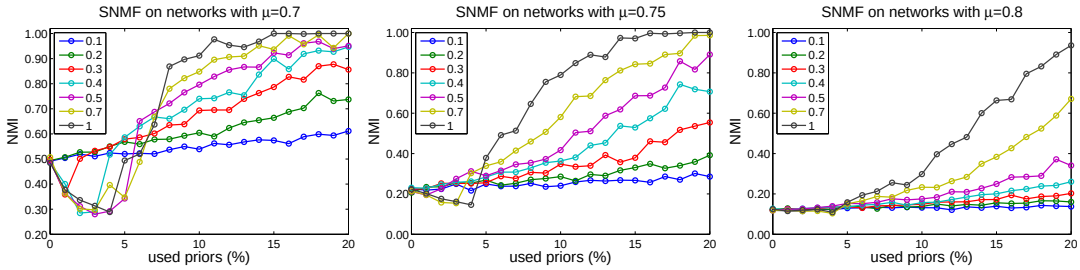


图 3.11: 在社区规模较大的 LFR 网络上，基于我们框架的 SNMF 算法的性能的变化曲线。三幅子图分别对应混合参数 $\mu = 0.7$，0.75 和 0.8 的网络。

4. 在图3.11中左侧的图中，当我们选用一个较大的 λ 但是先验信息却相对有限的时候，随着 λ 的增加性能将有明显的降低。这主要是由在基于拓扑结构的对称非负矩阵分解的的对称性被加入的非对称的先验信息所破坏导致的。因此在先验信息相对较少的情况下，我们应当选取一个相当较小的 λ 。

总之，选择参数 $\lambda = 1$，即让拓扑结构和先验信息对最终结果有着相同的贡献的做法通常可以得到令人满意的结果。如果我们已知网络的结构相对模糊，我们可以适当的增大平衡参数 λ，即让先验信息在整个框架中起更加重要的作用；反之则适当的减小平衡参数。

3.4.4 真实网络

在这一节，我们将评估我们提出的框架在表2.1列出的前 8 个被广泛应用的真实网络上的性能。为了节省空间，我们这里只分别在非负矩阵分解和谱聚类方法中各选择了两个方法（NMF_LSE, NMF_KL, SC_ADJ 和 SC_LAP）作为代表来说明我们框架在性能提升方面的效果。结果如表3.1所示。从结果中我们可以发现虽然在增长趋势上存在一定的不光滑性，但是所有方法的归一化互斥信息（NMI）都可以随着加入的先验信息的增加而增长。此外，不仅在社区结构模糊的网络中（如 Polbooks 美国政治图书网络）可以达到很好的性能提升，在传统方法已经可以得到很好结果的结构清晰的网络上（如 Zachary's 空手道俱乐部网络和美国高校足球队网络），我们提出的半监督框架依然不会有性能损失。在所有的 32 个方法-网络组合中（4 个方法，8 个网络），有 15 个可以在加入 30% 先验信息时候将社区结构完全划分正确，即 NMI 达到 1，而对于原始的不

表 3.1: 基于我们提出框架的 4 种不同方法在 8 个真实网络上的性能。

Prior	**Word**				**Dolphins**			
Percent	**LSE**	**KL**	**ADJ**	**LAP**	**LSE**	**KL**	**ADJ**	**LAP**
Baseline	0.341	0.418	0.013	0.001	0.814	0.864	0.142	0.889
2%	0.341	0.583	0.009	0.022	0.814	0.938	0.692	0.978
5%	0.352	0.740	0.009	0.023	0.814	1.000	0.918	1.000
8%	0.807	0.906	0.016	0.040	0.814	1.000	0.956	1.000
10%	0.914	0.957	0.021	0.021	0.814	1.000	0.956	0.978
15%	1.000	1.000	0.133	0.392	1.000	1.000	1.000	1.000
20%	1.000	1.000	0.572	0.730	1.000	1.000	1.000	1.000
30%	1.000	1.000	0.961	0.987	1.000	1.000	1.000	1.000
Prior	**Football**				**Karate**			
Percent	**LSE**	**KL**	**ADJ**	**LAP**	**LSE**	**KL**	**ADJ**	**LAP**
Baseline	0.927	0.897	0.919	0.922	1.000	1.000	1.000	0.836
2%	0.926	0.909	0.927	0.921	1.000	1.000	1.000	0.935
5%	0.927	0.913	0.920	0.925	1.000	1.000	1.000	0.935
8%	0.924	0.920	0.930	0.920	1.000	1.000	1.000	0.935
10%	0.925	0.911	0.923	0.924	1.000	1.000	1.000	1.000
15%	0.926	0.928	0.931	0.918	1.000	1.000	1.000	1.000
20%	0.925	0.921	0.920	0.925	1.000	1.000	1.000	1.000
30%	0.929	0.943	0.943	0.942	1.000	1.000	1.000	1.000
Prior	**Polblogs**				**Polbooks**			
Percent	**LSE**	**KL**	**ADJ**	**LAP**	**LSE**	**KL**	**ADJ**	**LAP**
Baseline	0.527	0.527	0.014	0.001	0.540	0.562	0.505	0.574
2%	1.000	0.965	0.310	0.966	0.532	0.614	0.388	0.616
5%	1.000	0.990	0.611	0.986	0.578	0.664	0.570	0.660
8%	1.000	0.995	0.974	0.993	0.676	0.763	0.626	0.710
10%	1.000	1.000	0.987	1.000	0.698	0.759	0.660	0.722
15%	1.000	1.000	1.000	1.000	0.787	0.774	0.717	0.769
20%	1.000	1.000	1.000	1.000	1.000	0.866	0.810	0.843
30%	1.000	1.000	1.000	1.000	1.000	0.946	0.925	0.937
Prior	**Friendship6**				**Friendship7**			
Percent	**LSE**	**KL**	**ADJ**	**LAP**	**LSE**	**KL**	**ADJ**	**LAP**
Baseline	0.798	0.874	0.786	0.892	0.836	0.899	0.802	0.851
2%	0.798	0.909	0.861	0.894	0.843	0.908	0.825	0.874
5%	0.798	0.936	0.907	0.930	0.843	0.912	0.857	0.885
8%	0.801	0.909	0.887	0.950	0.833	0.921	0.898	0.934
10%	0.803	0.952	0.917	0.940	0.843	0.951	0.888	0.925
15%	0.808	0.937	0.927	0.960	0.846	0.930	0.913	0.935
20%	0.824	0.963	0.929	0.962	0.846	0.977	0.920	0.954
30%	0.889	0.967	0.951	0.965	0.884	0.985	0.943	0.973

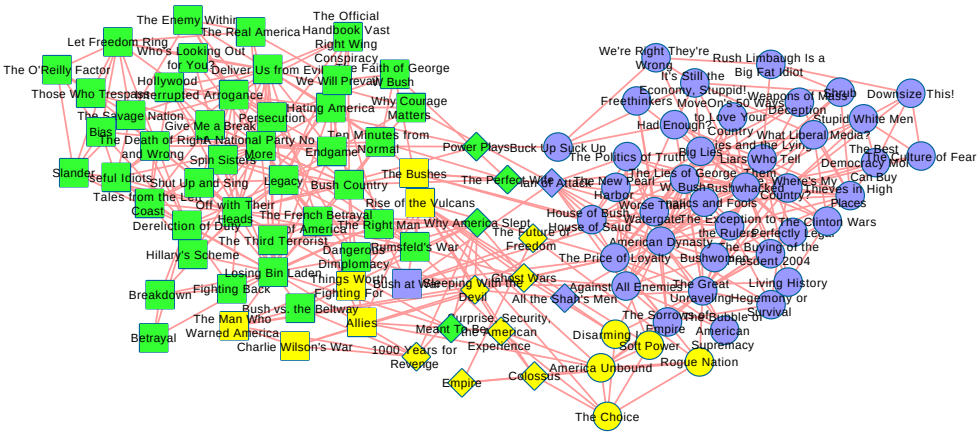
融入先验信息的无监督社区发现场景，他们中只有 3 个能做到这一点。这再一次说明了我们提出的框架的有效性。

3.4.5 用例研究

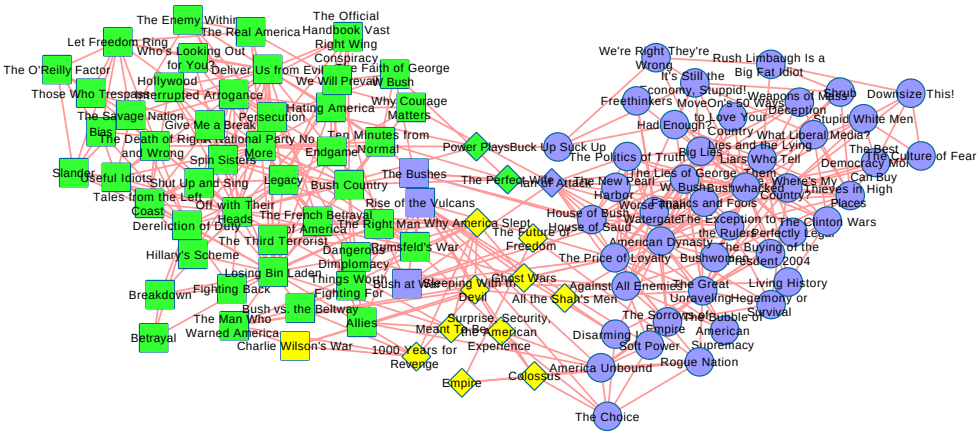
在图 3.12中，我们给出了一个针对美国政治书网络的实例研究。其中每个节点代表在亚马逊网站上销售的一本有关美国政治的书籍，节点之间的链接表示相应的两本书经常被同一个顾客同时购买。根据政治观点，这些图书被分为三类，即 “保守派”, “自由派” 和 “中立派” 。在图 3.12中我们用形状来表示真实的图书的类别，用颜色来表示来自我们框架的算法在不同监督信息的情况下得到的结果。我们这里使用错分节点的个数来从直观上衡量社区划分的结果的好坏。基于被同时购买的图书通常具有相似的政治观点这一假设，传统社区发现算法旨在将政治书网络仅仅根据共同购买的情况（即网络的拓扑结构）对图书的类别进行划分。但是网络的拓扑结构并不是十分的清晰，且单纯基于网络拓扑结构的方法有如下的几个缺点。

1. 根据节点之间的链接信息，我们只可以发现两个紧密链接的社区（即保守派图书社区和自由派图书社区），而实际图书社区的分类确是三个。这主要是因为，第三个社区也就是 “中立派” 图书通常是与 “保守派” 或 “自由派” 同时购买，而这些 “中立派” 图书并不是经常被单独购买。由于 “中立派” 图书社区并不具备清晰的社区结构，即这些节点之间并不是紧密连接的，因此只根据拓扑结构信息不足以清晰的划分出这个社区结构，从说就需要加入先验信息来辅助社区检测。

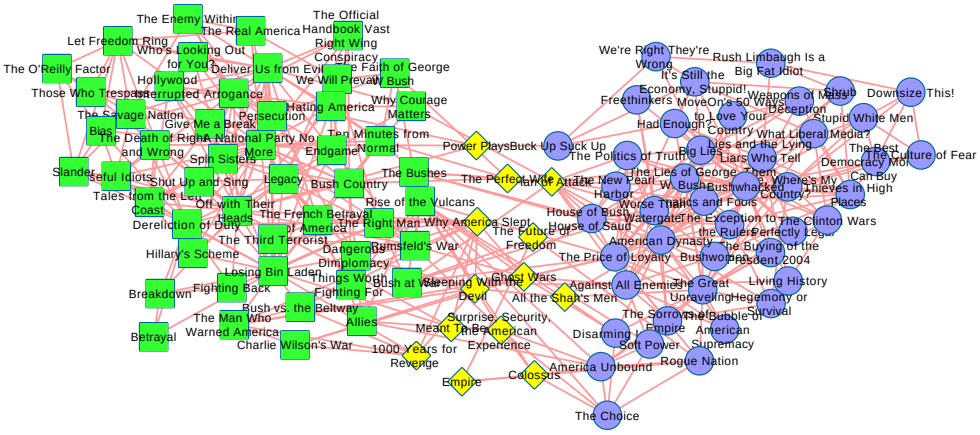
2. 从图3.12(a)中不含任何先验信息的 NMF_LSE 方法的结果来看，我们发现一些节点仅仅根据拓扑结构无法被正确划分。例如图书 *Power Plays* (节点 47) 与三个保守派的图书 *Arrogance* (节点 48), *A National Party No More* (节点 9) 和 *Off with Their Heads* (节点 13) 经常被同时购买；图书 *Buck Up Suck Up* (节点 103) 通常被和三本自由派图书 *We're Right They're Wrong* (节点 96), *Had Enough?* (节点 94) 和 *It's Still the Economy, Stupid!* (节点 95) 经常被同时购买；并且他们两本书也同行被一起购买。但是根据真实的结果 *Buck Up Suck Up* (节点 103) 是一本 “自由派” 的图书而 *Power Plays* (节点 47) 却是一本 “中立派” 的图书。而不含先验信息的 NMF_LSE



(a) 原始方法 (不融入监督信息的 NMF_LSE)，错分率: 18/105.



(b) 融入 10% 先验信息的 NMF_LSE，错分率: 7/105.



(c) 融入 20% 先验信息的 NMF_LSE，错分率: 0/105.

图 3.12: 一个关于美国政治书的网络的实例研究。

却将 *Power Plays* 错误的判定是一本“保守派”的图书。这个现象也说明了在结构模糊的网络中，单纯依靠拓扑结构信息不足以正确判别节点所属社区。

3. 对于原始的单纯基于拓扑结构的方法很难完全正确地判定社区的边界。例如在图 3.12(a) 中左下方的节点完全被原始的方法错误分类的。我们希望可以通过融入先验信息来更加清晰准确的确定社区的边界。

在图3.12中，我们给出基于我们框架的 NMF_LSE 方法在分别融入 0%, 10% 和 20% 先验信息时对美国政治书网络的社区检测结果。我们可以发现，随着加入先验信息的增多，被错误划分的节点的个数显著降低（从没有监督信息时的 18 个节点被分错，变为 10% 监督信息时 7 个节点被分错，再到 20% 监督信息时没有节点被分错）。此外，当我们加入 10% 先验信息时，如图3.12(b)所示，“自由派”图书社区和“中立派”图书社区之间的边界变得清晰了。进一步，当加入的先验信息增加到 20% 时，如图3.12(c)所示，“保守派”图书社区和“中立派”图书社区之间的边界也变得清晰了，且“保守派”图书社区可以被完全正确的检测出来。总之，随着加入的监督信息的增多，前面提到的三个问题都可以被逐渐的解决。通过实例研究，我们可以发现我们提出的半监督社区发现框架可以很好的解决单纯依赖拓扑结构的社区发现算法在结构不清晰的网络上进行社区发现所存在的大部分问题，并能显著的提升社区发现的准确度。

3.5 小结

在本章，我们首先对一组现有的社区发现算法给出了一种统一的解释，即基于隐空间节点表达的聚类算法。之后基于隐空间相似性，我们提出了一种统一的半监督社区发现框架。这种算法可以通过隐空间的图正则将成对先验信息与网络拓扑结构信息相融合，从而提升传统单纯基于网络拓扑结构的社区发现算法的性能。提出的半监督社区发现框架可以应用于绝大多数符合我们之前统一解释的基于矩阵的社区发现算法，如非负矩阵分解算法、谱聚类算法及他们的变种算法。与之前通过直接修改邻接矩阵将先验信息转化为拓扑结构信息的半监督社区发现算法不同，我们将半监督社区发现算法公式化一个整体的基于隐空间图正则的优化问题并可以权衡来自拓扑结构信息和先验信息对最终性能的贡献。在人工网络和真实网络上的大量实验说明了我们提出的框架在融入先验信息上的鲁棒性和高效性。

第四章　主动半监督社区发现

4.1　引言

如上一章所述，近年来研究人员提出了一些半监督社区发现的算法。通过结合监督信息或者是背景知识信息，这些方法可以显著的提升传统的单纯基于拓扑结构的社区发现算法的性能。通常来说，其中的监督信息可以被分为单节点的监督信息和节点间关系的监督信息（包括两节点之间的成对监督信息和多节点的高阶监督信息）。通过我们上一章的工作，我们可以将所有的监督信息都转化为基于链接的成对监督信息。成对形式的监督信息不仅具有普适性而且在人工标注上也更加方便。例如在一个社会化网络中，通过用户的个人信息，通常很难直接判断某个用户所所属的社区（兴趣组或者社区关系群）。但是如果我们发现两个人的个人信息比较相似，那么通常可以判定他们所属的社区是相同的，因此人工标注起来更加容易，成本也更加的低廉。通常来说，成对的监督信息通常可分为两类，即必连关系和必不连关系。有些算法只是结合其中的一类，如 [123] 只考虑如何融合必连成对关系的监督信息，而 [84] 更多的考虑的是如何融入必不连关系的监督信息。有些算法可以同时结合这两类成对监督信息，但是根据 [114, 115] 的实验表明，通常情况下必连成对监督信息比必不连成对监督信息在性能提升上有着更好的表现。

虽然成对监督信息融合策略已经成为半监督社区发现的主流研究方向且在性能的提升上有了长足的进步。但这些方法都预先假设成对监督信息是事先已知的，而忽略了哪些监督信息对性能的提升更为重要。这些方法在实验部分通常是随机选择一定比例的成对监督信息进行融入。但实际上，传统的无监督社区发现算法已经可以仅根据网络拓扑结构就将一些节点进行正确的分类，因此再增加与这些节点相关的监督信息就是无用的，且对性能的提升也没有过多的帮助。由于之前打绝大多数半监督社区发现算法被动地加入了过多对性能没有提升的监督信息，因此就需要大量的监督信息来提升最终的性能。

在机器学习领域，主动学习是这样的一种策略，它根据现有标注数据主动的选择一些靠近分类面的未标注的样本供人工标注，并将获得的标注样本加入到训练过程中，提升性能。由于其选择的样本靠近分类面，且是不确定性最高

的，因此对最终分类面的位置有着较大的影响，从而实现只依靠少量标注的监督信息就可以实现性能的大幅提升。

受到主动学习的启发，为了减少对监督信息的需求量，我们在本章提出了一种主动的链接选择框架。这个框架可以主动的选择一些链接供人工进行标注。但是与机器学习中的主动学习常常被应用于监督学习中不同，我们的社区发现本身是一个无监督学习的过程，因此并不存在最初的分类面。通过对现有社区发现算法的研究，我们发现在传统无监督社区发现算法得到的社区结构中，社区边界上的点是最容易被分错的节点。而且要确定一个社区，最主要的就是要确定其明确的边界。因此，我们认为处于社区边界上的节点是最不确定的节点，同时也是对社区发现算法最终性能影响最大的节点。

因此我们的主要想法是通过传统的无监督社区发现方法（以非负矩阵分解为例）先找出网络中处于社区边界上的节点，并通过人工标注将这些节点与其所属的社区核心节点相连接（连接策略），同时通过分析获得的监督信息将最有可能是社区之间的链接断开（断连策略）。通过以上的两步操作我们可以比添加随机选择的成对监督信息更加高效地使邻接矩阵的块结构变得清晰。我们迭代的运行传统的社区发现算法、连接策略、断连策略并修改网络拓扑结构，通过选择最不确定、最有信息量并对性能影响最大的链接进行标注，从而实现只需要少量监督信息就可以大幅提升性能的愿望。进一步，由于非负矩阵分解算法是初值依赖的并且通常需要尝试不同的初始值，并且在我们的主动链接选择过程中需要迭代的运行多次不同的非负矩阵分解，整个过程需要较长的时间。为了解决这个问题，我们提出了一种加速的策略。由于连续两次迭代过程中的邻接矩阵变化不大，因此我们将上一步非负矩阵分解的结果作为下一步的初始值。我们将本章的主要贡献总结为，为有效并高效的半监督社区发现设计了一种链接选择策略。

4.2 主动边选择策略

4.2.1 模型概述

图4.1中给出了主动链接选择框架的流程图。在每一步中，我们首先基于当前的网络拓扑结构运行一次非负矩阵分解。基于本次非负矩阵分解的结果，我们之后应用连接策略来主动的选择最不确定的链接供人工标注。值得注意的是，这并不是一个被动的接受标记数据的过程，而是一个交互的过程：首先根据算法

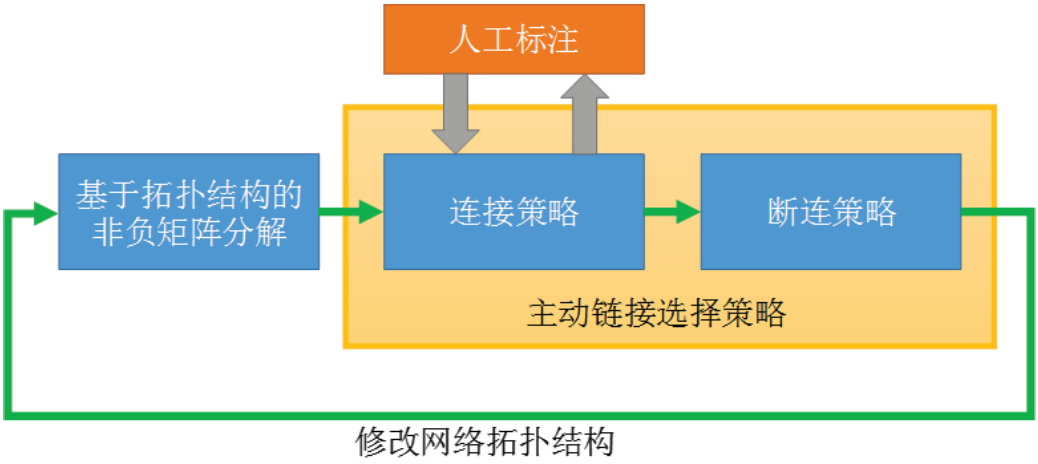


图 4.1: 主动边选择框架的流程图。

选择链接，将待标记的链接交给人工处理，最后将人工标注的结果融合到现有的网络拓扑结构中。之后为了能进一步充分挖掘获得的标注的数据中所蕴含的信息，我们使用断连策略再断开可能影响最终性能的社区间的链接。值得注意的是，在这一步我们不再需要任何人为的参与。最后我们基于前两步的结果更新网络的拓扑结构，并将其作为下一轮迭代中非负矩阵分解的输入。这个过程迭代的进行，并当连续五次迭代都没有社区结构变化时停止。

4.2.2　选择策略

根据2.2节中非负矩阵分解（2.6）的生成模型解释，我们可以知道分解后的矩阵 $\mathbf{H}$ 中的每个元素代表了节点隶属于某个社区的概率，而 $\mathbf{H}$ 的每一行 $\mathbf{h}_i$ 代表了第 i 个节点属于每个社区的概率分布。在通常情况下，我们会将节点 i 划分到其所对应的概率最高的那个社区中。我们这里就根据这个概率分布 $\mathbf{h}_i$ 来确定哪些节点是属于社区的核心，哪些节点是属于社区的边界处。如图4.2所示，如果节点 i 的隶属度分布集中在某一个社区上，那说明这个这个节点属于这个社区的置信度非常高，而属于其他社区的可能性非常小，我们认为这些节点是处于社区核心的，例如图中的节点 8。反之，如果某个节点的最大值并不是很突出，也就是说属多个社区的概率都差不多，那么虽然可以根据最大的值判定它所属的社区，但是置信度相对较低，我们认为这些节点是处于社区边界处的，例如

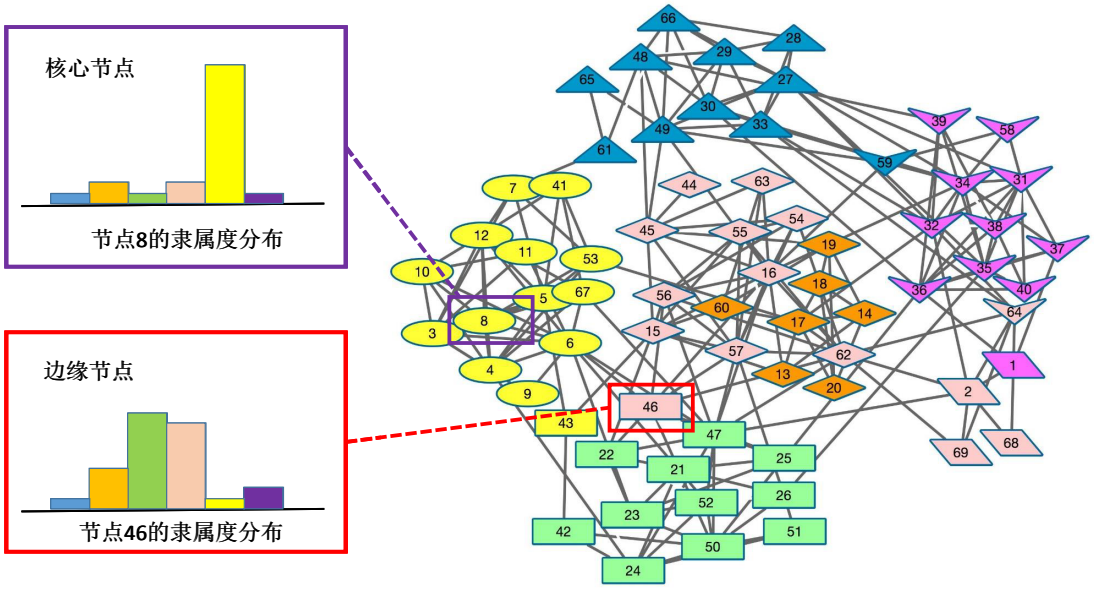


图 4.2: 基于节点隶属度分布的熵确定节点在社区中的角色。其中熵较大的认为是出于社区边界，熵较小的认为是出于社区核心。

图中的节点 46。信息熵是用来衡量一个事件不确定性高低的常用指标，当对于所有可能性的概率都相等时熵的值最大，也就代表不确定性最大；当只有某个情况发生的概率为 1 而其他情况概率均为 0 的时候熵的值最小，也就代表不确定性最小。我们在这里采用信息熵来度量节点隶属分布的不可信度

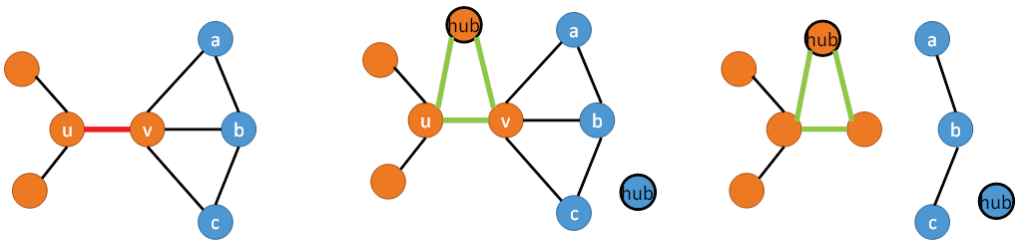
$$\mathcal{S}(\mathbf{h}_i) = -\sum_{k=1}^{K} P(\mathbf{h}_i = k)\log(P(\mathbf{h}_i = k)) = -\sum_{k=1}^{K} h_{ik}\log(h_{ik}). \tag{4.1}$$

当得到单个节点的熵后，一条边的熵就可以被定义为其所连接的两个节点的熵的平均值。为了方便后面的叙述，我们给出如下的两个形式化的定义。

定义 4.1 (社区 ε-核心节点)**.** 社区的 ε-核心节点是社区中具有最小信息熵的 ε 个节点的集合。

定义 4.2 (社区 ε-边界节点)**.** 社区的 ε-边界节点是社区中具有最大信息熵的 ε 个节点的集合。

根据社区结构的定义，社区是网络中的一组点的集合，集合中的点之间有着紧密的连接，而结合之间的点连接较少，因此当按照真实社区结构对节点进



(a) 网络拓扑结构　(b) 连接策略　(c) 断连策略

图 4.3: 主动边选择策略的可视化解释。(a) 网络拓扑结构及被选择的链接。(b) 连接策略。(c) 断连策略。

行排序后，网络的邻接矩阵就有如图4.5(a) 所示的块结构，其中蓝色表示其行列对应的节点之间无边相连，红色表示行列对应的节点之间存在边相连。当网络的块结构明显时，社区发现方法通常都能得到很好的结果；而当块结构不清晰时，很多方法的性能就会出现明显下降。而导致网络中块结构不清晰的主要原因有两个：一方面社区之间的连接过于紧密，另一方面社区内部的节点的连接不够紧密。

社区之间连接过于紧密的问题，也就是社区之间有过多互连的边的问题，通常体现在网络邻接矩阵中在块结构之外的地方存在过多的非零元素，如图4.5(a) 中，在块结构之外也有很多值为 1 的元素。而由于网络边界处的节点与其他的社区有较强的联系，这些社区间链接的两个端点都处于社区边界的可能性较大。而且这些节点是不确定性最高的节点，也是最容易被分错的节点。因此我们应该优先选择处在不同社区的边界且他们之间存在的连接的两个节点。将这些边提取出来供人工标注，且人工只需要确定这两个点是否属于同一个社区，即这条已经存在的边是否应该被断开。

而社区内部节点间连接不够紧密的问题，主要是由节点与社区内的其他节点连接过少而与其他社区的节点连接过多造成的。如图 4.3(a) 所示，橙色和蓝色节点分别属于两个社区，节点 u 和 v 属于橙色的社区，但是节点 v 却只连接了橙色社区中的一个节点，而连接了蓝色社区中的三个节点，这就使得单纯基于拓扑结构的算法通常把节点 u 分到橙色社区，而将节点 v 分到蓝色社区。在这种情况下，即使我们选出边 $< u, v >$ 为社区之间的边供人工标注，人工标注的结果确定节点 u 和 v 属于同一社区，且他们之间的边应该存在，我们也无

法正确的将节点 v 分到橙色社区中。为了解决这个棘手的问题，在运行完每一步非负矩阵分解后，我们选择每个社区的 1-核心节点，并由人工标注确定每个选出的处在边界上的节点与哪个社区的核心节点属于同一个社区，并增加一条相应的必连的边。与前面一样，人工只需要判定两个节点是否属于同一个社区，而不用确定具体所属的社区。如图4.3(b) 所示，通过连接边界节点与其对应的社区核心节点，节点 v 将被显著的拉向橙色社区。虽然我们可以通过增加权重来提升新边的影响，但是出于方便及易用性起见，我们令所有边的权重为 1。我们将前面的连接策略总结为：

定义 4.3 (连接策略（策略 1）)**.** 基于当前非负矩阵分解的结果，我们为每个社区选择其 1-核心节点（熵最小的节点），并选择 n_{select} 条社区间的链接，这些链接的两个节点都是处于社区边界处节点并且属于不同的社区。之后，通过人工标注确定是否选出的链接的两个节点属于同一个社区，如果是我们保留这条链接，如果不是我们断开这条链接。我们还通过人工标注将每一个边界处的节点与其所属社区的核心节点相连。

进一步，既然我们是根据非负矩阵分解的结果判定链接的两个端点是否属于同一个社区，而不是真实结果，从而这个确定的结论有可能是错误的。因此我们可以将连接策略（策略 1）放松到忽略所选连接两个端点是否属于不同的社区。我们将这个策略称为松弛连接策略（松弛策略 1）。

为了进一步挖掘人工标注的边界节点和核心节点之间的关系信息，我们还将断开和边界节点相连的但是不属于经过人工标注相连的核心节点所在社区（基于非负矩阵分解得到的社区结果）的节点之间的连接。直觉上的想法是，除去被人工标注必连的核心节点外，边界节点还有可能和其他社区中的节点相连，因此我们有很大的理由将这些连接断开。如图4.3(c) 所示，由于节点 a，b 和 c 和节点 v 连接到的橙色核心节点（橙色 hub）不属于同一个社区，因此我们断开 v 和节点 a，b，c 之间的连接。值得注意的是，虽然我们可以通过人工标注得到精确的关于是否 v 和节点 a，b，c 属于同一个社区的结果，但是这会需要大量的的人工参与。既然我们提出的框架的目标是减少对监督信息的需求量，即降低人工参与的程度，因此我们仅仅使用非负矩阵分解的结果来判定是否属于同一个社区。虽然这可能会引入一些错误的断连操作，但是实验结果表明，这将大幅度提升社区发现的性能。我们将断连策略总结如下：

定义　4.4 (断连策略（策略 2）). 基于当前非负矩阵分解的结果，我们断连选定的边界节点和不属于被人工标注必连核心节点所在社区的节点之间的边。

通过使用连接策略，我们可以根据非负矩阵分解的结果交互的选择最不确定的连接供人工标注。此外，通过采用断连策略，我们可以充分利用已添加的边的信息。这两个策略可以帮助我们增强邻接矩阵的块结构，即一方面通过断开社区间的链接使社区间的链接变得稀疏；另一方面通过将社区内最不确定的节点与社区核心节点相连使社区内的连接变得紧密。

4.2.3　加速策略

由于非负矩阵分解本身不是凸的，因此通过梯度下降法求解得到的结果是依赖于初始值的。因此要得到令人满意的结果，通常我们需要用不同的初始值重复迭代过程（通常 20 个随机初始值）并选择能够最小化重建误差的结果。这将严重影响非负矩阵分解算法的效率。除此之外，由于我们的算法需要迭代的更新网络的邻接矩阵并对其进行非负矩阵分解，因此这个问题在我们的框架下就变得更为严重。通过实验我们发现，虽然每次迭代过程会修改邻接矩阵，但是由于我们每次只修改非常少的边（$2n_{select}$ 条边），相邻两次的邻接矩阵的差异不是很大，这使得他们的分解结果应该差异也不是很大。因此，我们用前一步非负矩阵分解的结果作为下一次非负矩阵分解的初始值，而无需再尝试其他的初始值。这将使得我们的算法更加的高效。一方面，由于采用相似矩阵分解结果作为初始值，非负矩阵分解的初始化离最终结果很近，这缩短了更新迭代的次数，使得算法只需非常少的几步就可以收敛。另一方面，我们不需要重复尝试不同的初始值。因此我们的算法所需要的时间不足原来的二十分之一，这对于大规模的网络尤为重要。

4.2.4　时间复杂度分析

通过考虑邻接矩阵的稀疏性,每一次非负矩阵分解迭代的时间复杂度是 $O((NK+M)K)$。连接策略需要 $O(N+M)$ 浮点运算来计算节点和边的信息熵，$O(TM)$ 浮点运算来选择 T 个最不确定的链接，$O(N)$ 浮点运算来为每个社区选择一个核心节点。因此连接策略的时间复杂度是 $O(N+TM)$。断连策略需要 $O(Td)$ 浮点运算来断开最有可能的社区间连接，其中 d 是平均节点的度。总之，在每步迭代中，算法需要 $O((NK+M)K)$ 次浮点运算来求解非负矩阵分解和

$O(N+TM+Td)$ 次浮点运算来更新网络拓扑结构。因此我们整个框架的计算复杂度是近线性与网络规模的（M 和 N）。

除此之外，我们框架的三个组成部分，即非负矩阵分解、连接策略和断连策略，都非常容易被并行化或者分布式计算。具体来说，一方面，非常多的非负矩阵分解并行算法已经被开发；另一方面，连接策略中计算节点和边的熵的运算也可以被十分容易的并行化。且断连策略中所需的计算量非常少，并且被可以很快的在单机上完成。

总之，基于以上的讨论，我们提出的半监督社区发现框架在计算复杂性上具有应用到大规模真实网络上的可能。

4.3 实验结果与分析

4.3.1 参数设定

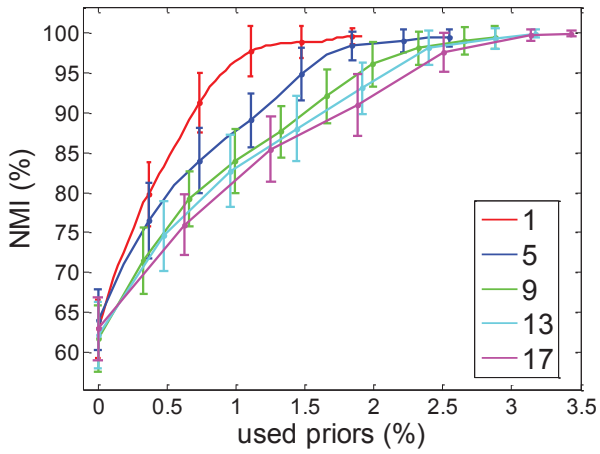


图 4.4: 在 $Z_{out}=8$ 的 GN 网络上参数 n_{select} 对性能提升的影响。其中每条曲线代表选取某一 n_{select} 时性能随着先验信息增加的增长速度。

在我们提出的主动链接选择框架中，有两个参数需要确定。一个是在连接策略中的每次选择的链接数 n_{select}，第二个是整个过程需要迭代的步数 n_{iter}。要研究选择的链接数 n_{select} 对性能提升的影响，我们在 GN 网络上开展性能增长与参数设置之间关系相关的实验，并令 n_{select} 从 1 变到 20。结果如图4.4所示，其中每条曲线代表选取某一 n_{select} 时性能随着先验信息增加的增长速度。可见，随着 n_{select} 的增大，即每次迭代选择更多的边供人工标注，虽然相同比例的先

验信息所需要的迭代次数降低了，但性能增长的速度也有些许的放缓。因此，如果仅从性能提升的角度出发，我们应该优先选择较小的 n_{select}。但是在每次迭代中都要运行非负矩阵分解，而当网络规模较大时，这个算法将耗费较多的时间。因此，出于权衡性能增长速度和算法运行时间的考虑，我们在后面的实验中，令 n_{select} 为节点个数的十分之一，即 $n_{select} = \lceil 0.1N \rceil$。而对于总体的迭代次数 n_{iter}，我们通常在有五步迭代无法改变节点的社区隶属度时停止整个迭代过程。

4.3.2　人工网络

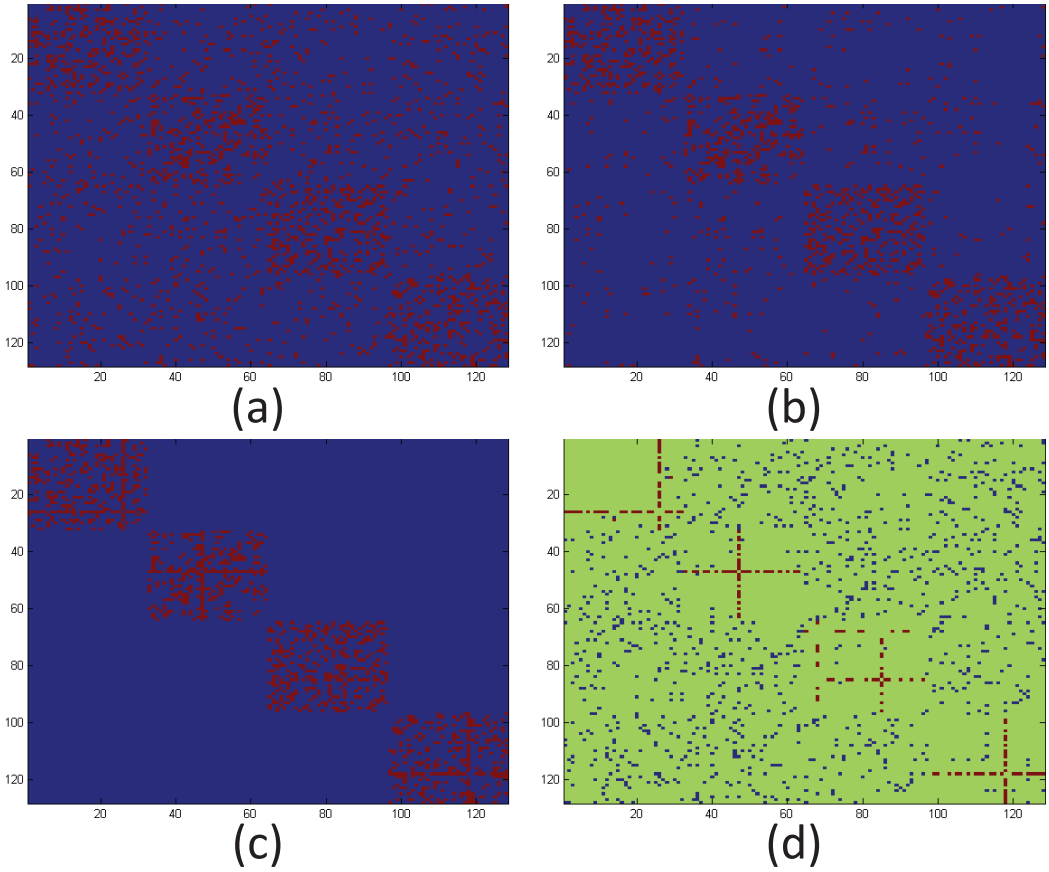


图 4.5: 在 $Z_{out} = 8$ 的 GN 网络上，邻接矩阵的变化情况。

与前一章类似，我们首先在 GN 和 LFR 这两个人工网络上验证我们提出的主动链接选择框架的高效性。图4.5给出了一个在 GN 网络上的示例，来说明

我们的框架是如何锐化网络邻接矩阵的块结构的。如图4.5(a) 所示，在 $Z_{out} = 8$ 的 GN 网络上，邻接矩阵的块结构并不是十分的清晰。这是因为有很多的非零元素遍布在邻接矩阵块对角线之外的部分，这也就意味着存在很多的社区间的链接。图4.5(b) 和 (c) 展示了迭代 20 步和 50 布后网络的邻接矩阵，图4.5(d) 表示 50 步迭代后网络拓扑结构和最初拓扑结构之间的差异。可见在迭代过程中，通过将一些不确定的节点和他所属社区的核心节点相连接（连接策略）并断链大量与非其所在社区核心节点相连节点之间的边（断连策略），块结构逐渐变得清晰，且社区检测任务逐渐变得容易。

人工网络上的定量结果如图4.6所示。其中第一行是在 GN 网络上的结果，和添加随机选择的成对标签的基准方法相比，我们提出的框架下所有四种组合策略都在使用相同数量监督信息的情况下大幅提升了社区发现的性能。例如在 $Z_{out} = 8$ 的 GN 网络上，在添加 1% 随机选择链接的监督信息后，基准方法的性能只能从 62% 提升到 68%。而在我们的框架下，通过添加基于连接策略选择出来的 1% 的链接的监督信息后，性能可以提升到 85%。在此之后再应用我们的断连策略，可以进一步将性能提升至 95%（断连策略并不需要任何额外的人工干预）。可见我们提出的主动链接选择策略可以在使用同等数量监督信息的情况下使性能的增长量比基准方法提升将近 5 倍。

为了进一步说明算法的高效性，我们在更加复杂的 LFR 网络上开展实验，我们这里主要设置混合参数 $\mu = 0.8$，及两种规模的社区（小规模的社区中节点个数在 10-50，大规模社区中节点个数在 20-100），如图4.6第二行所示。一方面我们可以发现框架对性能的提升不受社区规模的影响。另一方面从结果中可以发现，我们提出的框架在结构相对复杂的大规模网络（LFR）上的效果提升比在结构相对简单的小规模网络（GN）上的提升更为显著。例如在社区大小在 20-100 的网络中，在随机加入 0.4% 监督信息的情况下，基准方法只能将性能从 22% 提升到 23%。而通过添加根据我们框架连接策略选择出的链接的监督信息后，性能可以提升至 42%，而当再次使用断连策略后，性能可以进一步提升至 85%。上面的两组实验充分说明了我们提出的主动链接选择框架在人工网络上的有效性和高效性。

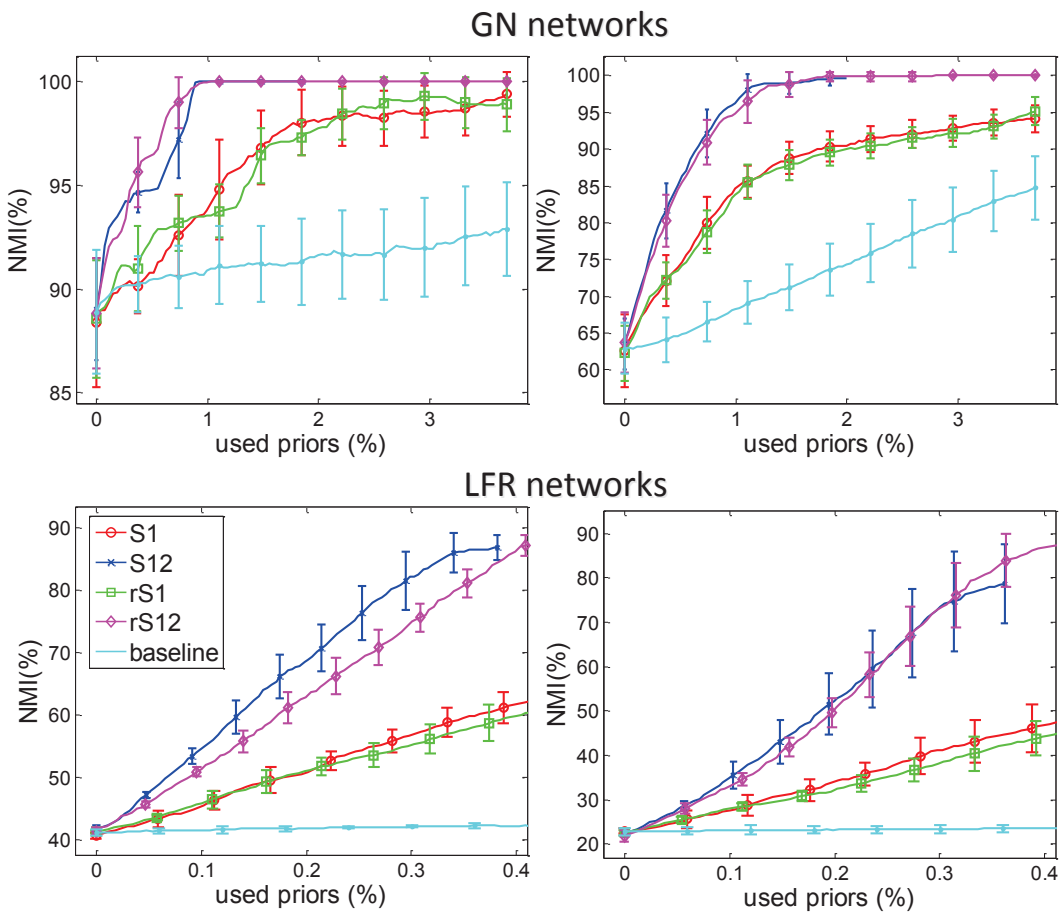


图 4.6: 在人工网络上主动链接选择框架的性能。

4.3.3　真实网络量化比较

在本节，我们将在 18 个广泛使用的真实网络上验证我们提出的主动链接选择策略。其中值得注意的是美国高校网络-6 和美国高校网络-7 具有相同的拓扑结构和不同的社区划分。美国高校网络-6 将学生根据他们的年级划分为六个不同的社区；而美国高校网络-7，将 9 年级学生社区再次划分成了两个子社区，即白人学生社区和黑人学生社区。

首先，我们选择 6 个经典的网络（海豚网络，美国政治书网络，美国政治博客网络，美国高校足球俱乐部网络，美国高校网络-6 和美国高校网络-7）进行细致的比较。这些网络的规模都小于 1,500，因此我们用不同的初始值和随机数重复每个实验 10 次。图4.7中，我们给出了在这六个真实网络上基准方法和

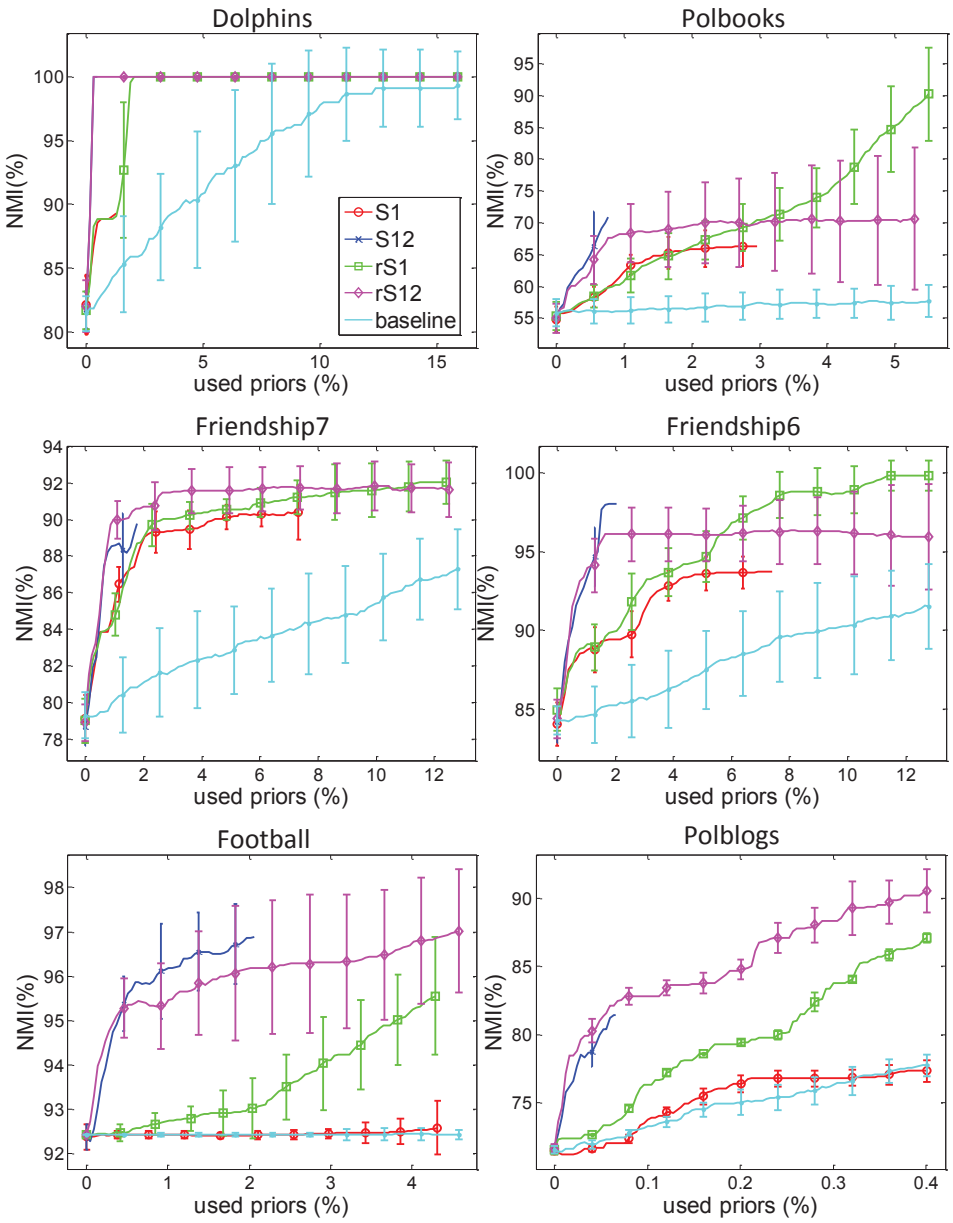


图 4.7: 在 6 个真实网络上主动链接选择框架的性能。

我们框架下的四种策略组合的性能。每条曲线代表不同方法随着加入成对监督信息比例增加的变化情况。容易发现，我们框架下的四种策略组合的性能增长速度都比基准方法要快。进一步，组合了断连策略和连接策略（松弛连接策略）的 S12（rS12），比单纯使用连接策略（松弛连接策略）的 S1（rS1）增长的速度

要更快。以海豚网络为例，基于最初拓扑结构的非负矩阵分解算法的性能可以达到 82%。为了使性能达到 100%，即完全正确划分，随机半监督算法（基准算法）需要超过 15% 的链接监督信息，而我们提出的 rS12 和 rS1 仅仅 0.5% 和 2.5% 的链接监督信息。以美国高校网络为例，如果令真实的社区个数为 6，添加 5% 的监督信息后，基准方法仅可以达到 80%，而我们提出的 S1 和 S12 分别可以达到 86% 和 90%。如果我们令社区的个数为 7，为了达到 92% 的的性能，基准方法需要添加 13% 的监督信息，而我们提出的 rS1 和 rS12 分别仅需要不足 3% 和 1% 的监督信息。

虽然组合连接策略和断连策略这两种策略可以在加入少量监督信息的时就有非常快的增长速度，但是在中小型网络上当加入的监督信息达到一定量之后，他们仅会保持一个稳定的状态，而不再增长。这里的主要原因可能是，由于我们使用的上一次非负矩阵分解的结果来进行断连策略判断的，因此虽然它可以断开一些社区间的链接，但同时也有可能断开社区内部的链接。虽然我们可以通过使用人工标注的链接信息来取代从非负矩阵分解中得到的结果来作为断链操作的依据，但是这将大幅增加对监督信息的需求。这一点完全违背了我们要减少对监督信息需求的初衷。此外，虽然标准的连接策略更加直观一些，但是松弛的连接策略通常能得到类似或者更好的性能。这主要是因为在连接策略中判断两个节点是否属于同一个社区是由之前的非负矩阵分解的结果推断的，而不是真实的标签。因此这样的推断存在偏差，从而使连接策略和松弛的连接策略从本质上是一致的。

为了进一步验证我们的框架，我们在 12 个具有较大规模的真实社交网络上将其与基准方法相比较。这些网络来自美国 12 所高校在校学生的 Facebook 社交网络，他们的规模介于 12,000 到 21,000 之间。在这些网络中，朋友关系用无向边表示，此外还提供了每个学生的宿舍楼、主修专业、辅修专业、入学年份、性别及毕业的高中这六种信息。与 Traud 等人指出的相一致，我们这里使用学生的入学年份作为社区真实结构的划分依据。为了简单起见，我们只将在我们框架下同时采用松弛连接策略和断连策略的 rS12 方法与基准方法进行比较。结果如图4.8所示。从结果中容易发现，在大部分网络上（除去 NYU 网络），原始基于拓扑结构信息的非负矩阵分解算法的性能只能达到不到 40% 的精度（基于 NMI），且通过添加随机选择成对监督信息的基准方法对性能的提升是十分有限的。但通过采用我们提出的框架，性能却可以显著提升。以 Northeastern 网络为例，传统基于非负矩阵分解的方法的性能是 38.2%。通过添加 0.03%,

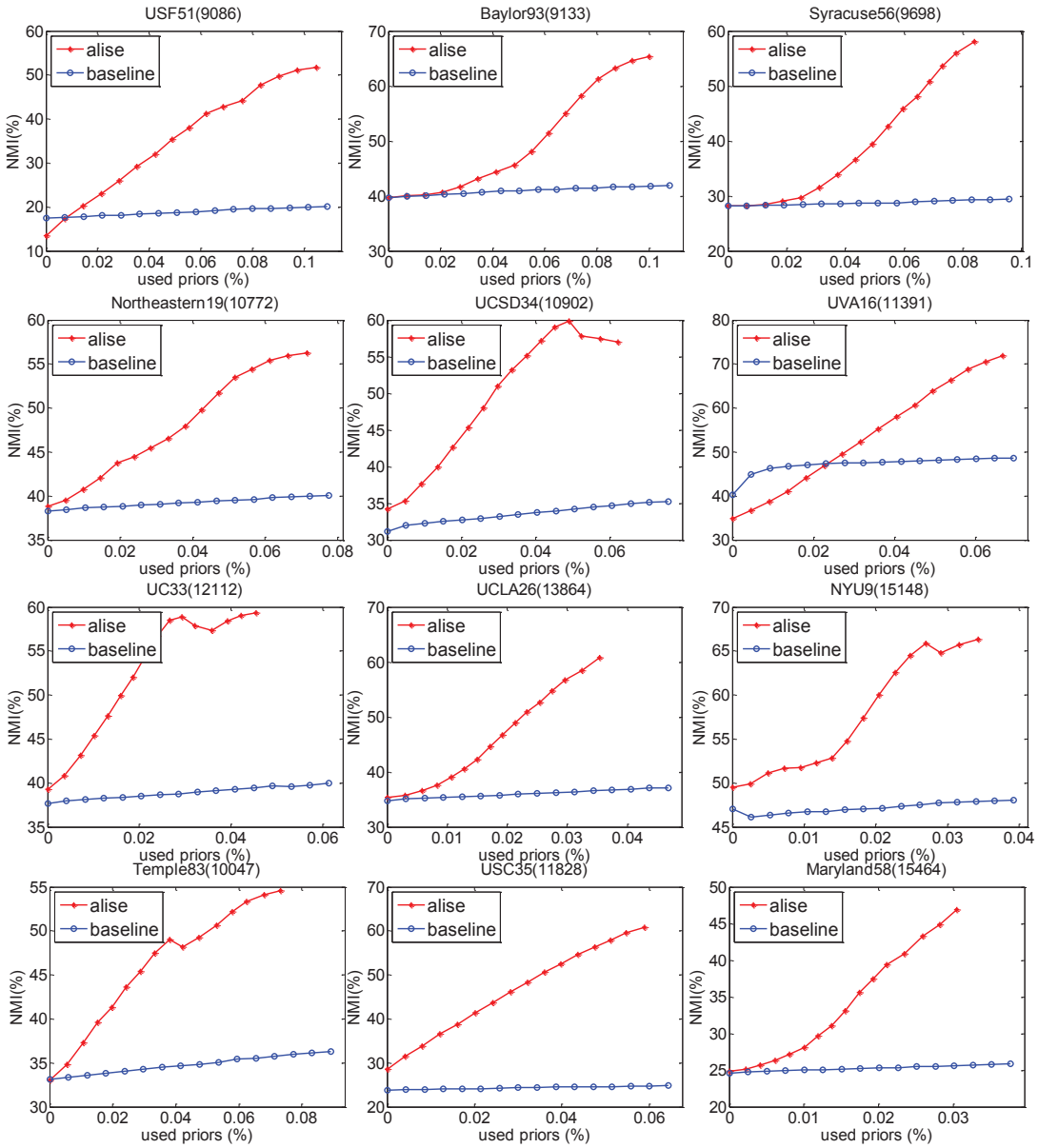


图 4.8: 在 12 个美国大学 Facebook 网络上主动链接选择框架的性能。

0.04%, 0.05% 0.06% 和 0.07% 的成对监督信息，基准算法的的性能只能分别达到 39.00%, 39.27%, 39.48%, 39.75% 和 39.99%，而我们的主动链接选择框架却可以分别达到 45.77% 48.70%, 52.82%, 55.09% 和 56.19%。我们框架的性能提升大约是基准方法的 9 倍。

总之，从量化比较，我们发现我们提出的主动链接选择策略和现在最为先

进的随机半监督社区发现方法相比，要达到同样的性能所需的监督信息量要少很多。也就是说，我们提出的框架在监督信息的利用上更加的高效。

4.3.4　真实网络用例研究

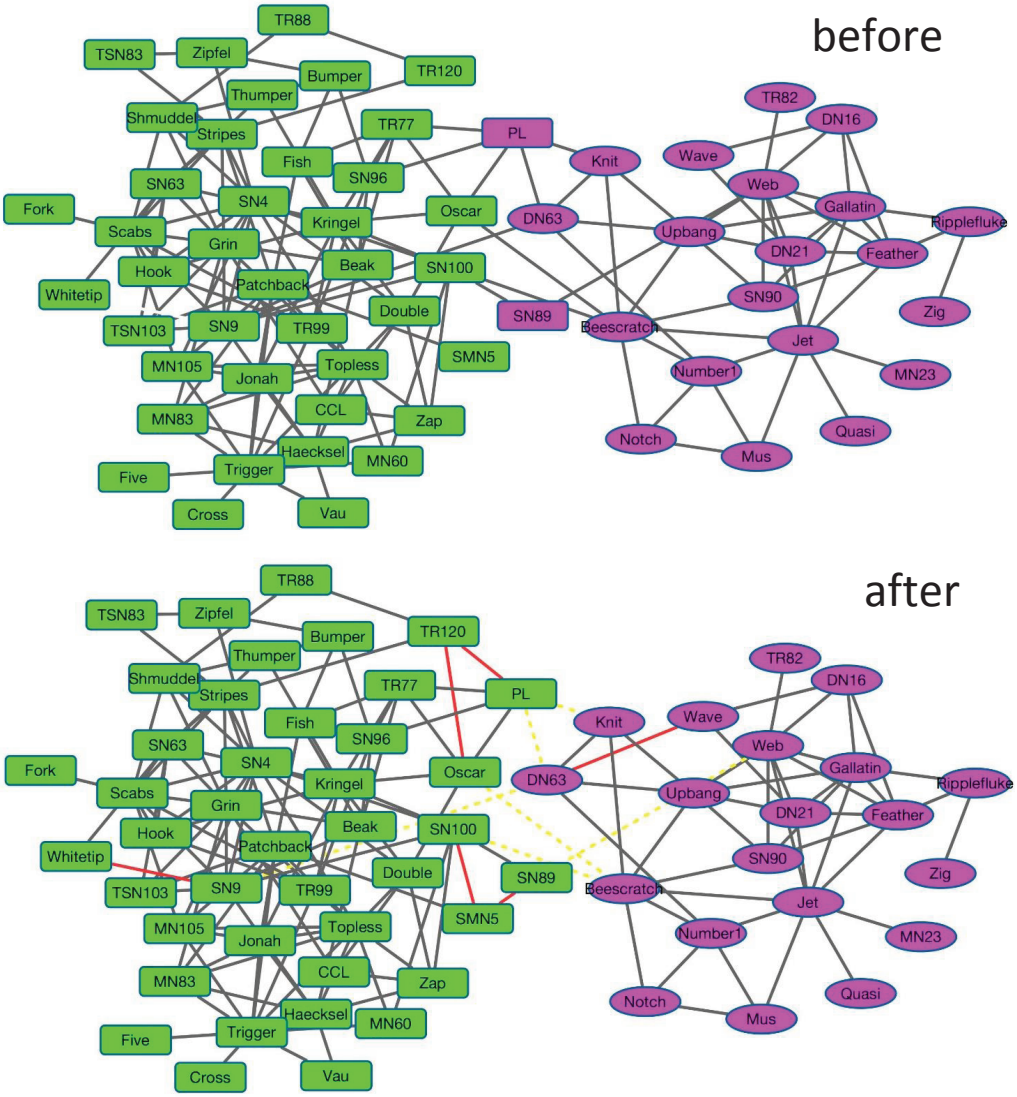


图 4.9: 在使用我们的主动链接选择策略前后的海豚网络的网络拓扑结构和检测到的社区结构。

在本节，我们选取 4 个有代表性的真实网络进行实例研究，而其中尤其以海豚社交网络做最为细致的讨论。如图4.9所示，在不使用我们提出的主动链接

选择框架时，单纯基于网络拓扑结构的非负矩阵分解算法可以正确的划分大部分的海豚，且基于 NMI 的性能可以达到大约 81%。但它却无法正确获得 FL 和 SN89 两只海豚所属的社区，即把这两只雌性海豚误判为雄性海豚。

我们首先基于原始网络拓扑结构进行一次非负矩阵分解。根据主动链接选择框架，我们根据非负矩阵分解的结果确定每个节点的信息熵，并据此为每个社区选择一个核心节点。TR120 被选作雌性海豚的核心节点，Wave 被选作雄性海豚的核心节点。我们根据非负矩阵分解的结果进行第一轮的主动链接选择。根据连接策略，我们选择海豚 PL 和 Oscar 之间的链接来进行人工标注（因为这条边的信息熵最大，也就是说明最不确定）。由于根据人工标注这两个节点属于同一个社区，我保留这条链接，并根据人工标注将这两个节点与其对应的社区的核心节点（TR120）相连接。根据断连策略，由于根据非负矩阵分解的结果 Knit, DN63 和 Beescratch 不属于 TR120 所在的社区，我们断开链接 <PL, Knit>, <PL, DN63> 和 <Oscar, Beescratch>。这样我们就得到了新的网络拓扑结构，我们基于此进行第二次的非负矩阵分解。从这次非负矩阵分解的结果我们可以而发现节点 PL 已经被正确的分到雌性海豚一类，并且算法基于 NMI 的性能达到了 89%，但是节点 SN89 依然没有被正确分类。

之后我们基于第二次非负矩阵分解的结果进行第二轮的主动链接选择。这里值得注意的是，我们并不确定哪些节点是否被正确划分。和第一轮类似，我们选择海豚 SN100 和 SN89 之间的链接来供人工标注。根据标注的结果，我们依然保留这条边，并将它们连接到新的社区核心 SMN5，并断链 <SN100, Beescratch> 和 <SN89, Web>。基于第二次修订后的网络，我们运行第三次非负矩阵分解。通过结果我们发现所有的海豚都已经被正确划分了，且算法的归一化互斥信息达到了 100%。此时，我们并不知道是否所有节点都被正确分类了。如果我们继续进行这个过程，链接 <SN9, DN63> 将被选取，一些边被加入另一些边被移除。但是之后的过程不会影响最终的社区检测的结果。这个迭代过程再继续 5 轮，直到没有节点隶属度变化为止。

最终，使用我们主动链接选择策略后的网络拓扑结构和社区检测结果如图4.9第二行所示。在其中，红色实线代表加入的链接而黄色虚线代表移除的链接。从最终的结果中我们发现，主动链接选择框架将海豚网络划分为两个不连通的子图，这使得社区发现变得极为容易。从这个可视化的过程我们可以发现，我们提出的策略不仅可以选择最为不确定的且对检测性能影响最大的节点

供人工标注，还可以通过融入少量的人工标注结果高效并正确地划分网络节点。

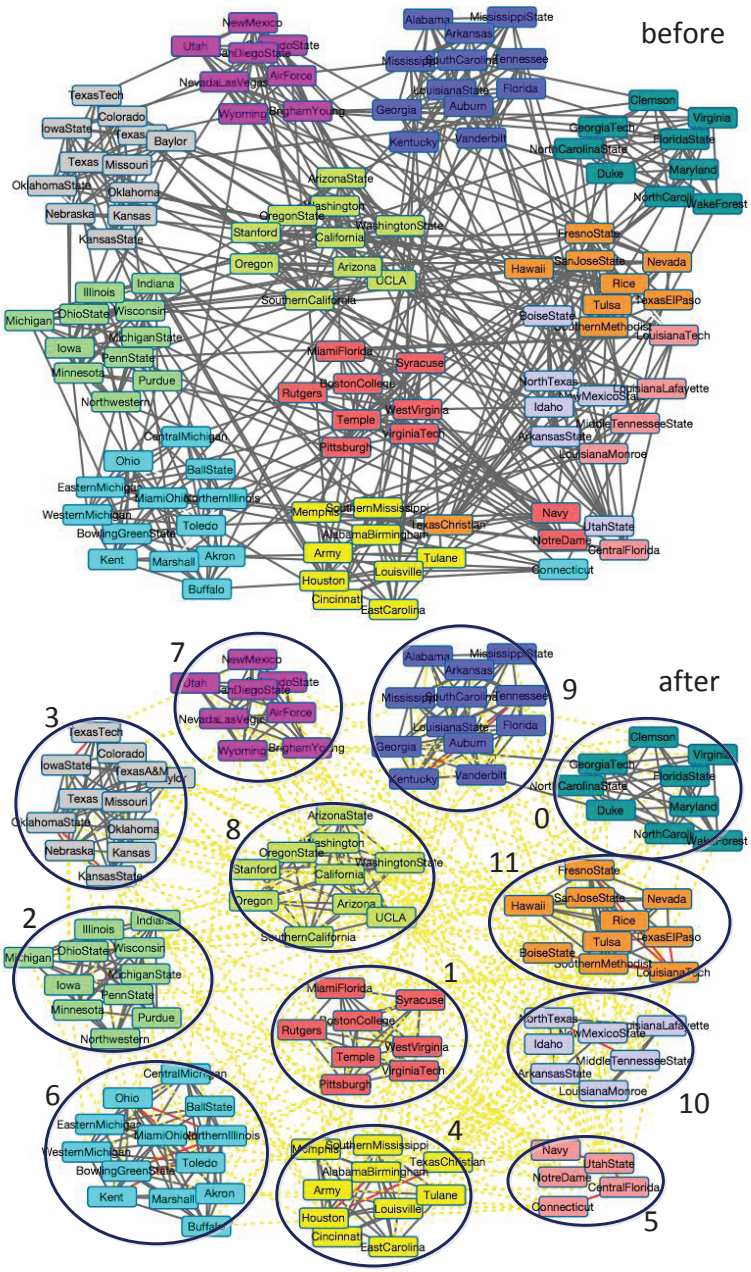


图 4.10: 在使用我们的主动链接选择策略前后的美国高校足球俱乐部网络的拓扑结构和检测到的社区结构。

如图4.10中第一行所示，在美国高校足球俱乐部网络中，当设定社区个数

为 12 时，原始的基于网络拓扑结构的非负矩阵分解算法只能正确划分 8 个社区。在联盟 4（USA 联盟），只有一个节点（TexasChristian）由于过多的与联盟 11（Western Athletic 联盟）中的队伍进行比赛（与 11 各队伍中的 8 个比赛过），而被算法错误划分。但是由于下面两个原因，联盟 5 (Independents 联盟) 和联盟 10 (Sun Belt 联盟) 中的节点大部分都被错误分类。一方面，有些队伍过多的和其他联盟中的球队进行比赛，如联盟 5 中的 Navy 和 NotreDame。从社区发现的角度解释，这些节点有着过多的社区间链接而社区内部的链接相对较少。另一方面，在有些联盟中存在一些队伍，虽然他们之间比赛频繁，但是却与本联盟的其他球队比赛较少。从社区发现的角度分析，这意味着社区内部的链接过少。也就是说一个大的社区将被划分为一些小的社区。例如联盟 10 就被分为两个小的子社区。

图4.10中第二行显示迭代的进行主动链接选择 50 次之后的网络拓扑结构和被检测出的社区结构。通过人工标注 150 条链接，及之后的连接策略和断链操作，大多数的社区间的边被去除了，且一些重要的社区内部的边（链接边界节点和核心节点的链接）被添加了。以 BoiseState 球队为例，由于与他比赛的球队都在其他的联盟中，十分困难将其进行正确的分类。通过将其与联盟 11 的核心 LouisianaTech 球队链接并断链他的其他所有的链接，BoiseState 队可以正确的被我们的算法分到联盟 11 中。传统的通过添加随机选择链接监督信息的半监督社区发现算法，需要添加 1,199 条具有监督信息链接来正确分类所有节点。而我们提出的基于主动链接选择的框架只需要添加不到 150 条边就可以达到同样的性能。这再一次说明了我们的主动链接选择策略在使用链接监督信息上更加的高效。

根据学生的年级，美国高校学生网络可以被分为 6 个社区。但是有两个因素导致这个网络上社区发检测无法十分精确。一方面 12 年级仅有四名学生 (1, 2, 68 和 69) ，并且他们之间的链接并不是很紧密。另一方面在 9 年级学生中，有两个相对独立的子社区，即黑人学生社区和白人学生社区。如图4.11第一行所示，当设定社区个数为 6 时，单纯基于拓扑结构的非负矩阵分解算法既无法正确划分 9 年级及 12 年级的学生（将 9 年纪学生分为两个子图，且其中的一个子图与 12 年纪合并为一个社区）又无法正确划分处于社区边缘的学生（错误划分了学生 43, 46, 59 和 64）。

如图4.11第二行所示，通过采用我们提出的主动链接选择框架，只需要人

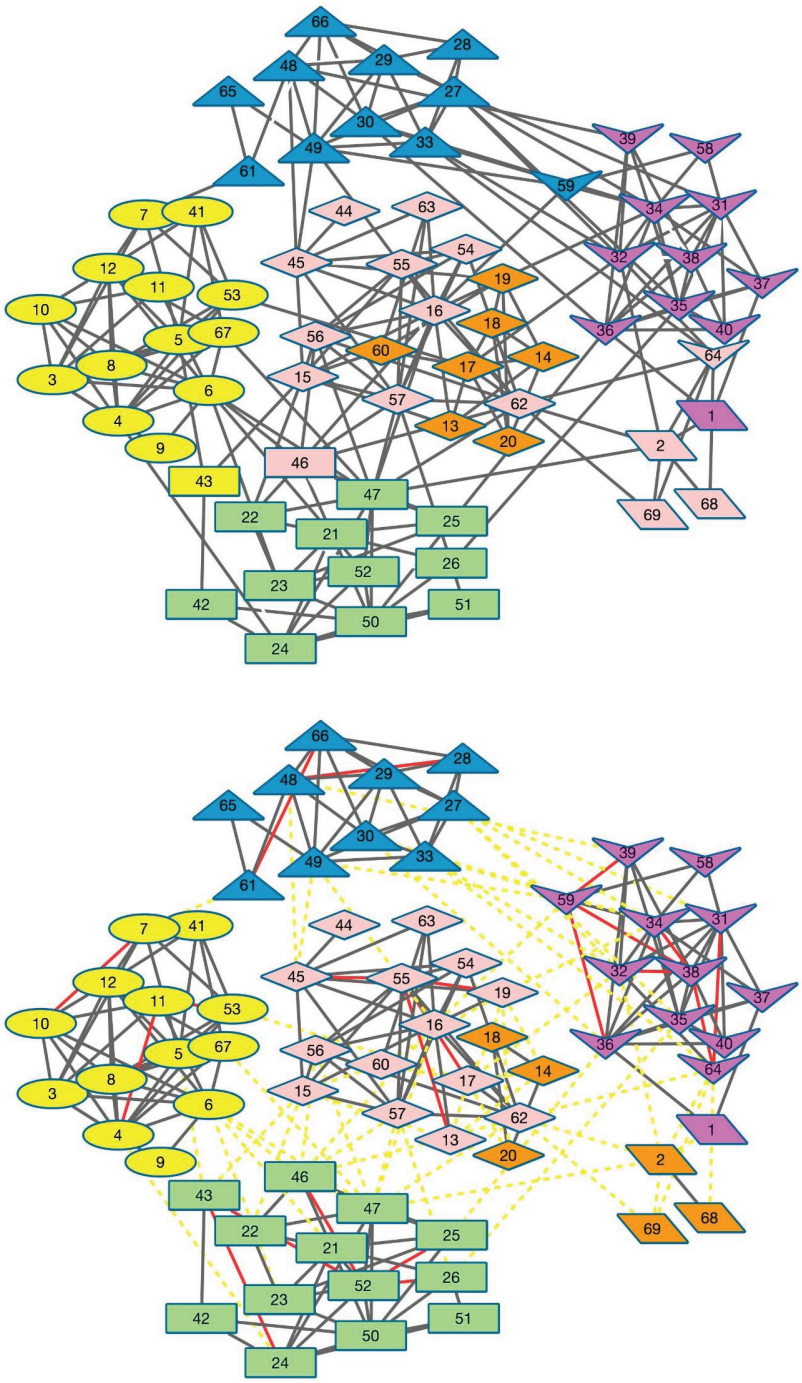


图 4.11: 在使用我们的主动链接选择策略前后的美国高校网络的网络拓扑结构和检测到的社区结构。

工标注 42 条链接，前面遇到的这两个问题可以被显著的改善。首先，所有处于社区边缘的学生都被正确的分类了。例如，由于处于 8 年级的学生 43 和 46 与其他年级的学生有着较多的联系，用传统的基于拓扑结构的方法他们就被误分到了其他社区。通过我们的算法，将他们与 8 年级社区的核心节点 52 号学生相连，基于新的拓扑结构他们就可以被正确分类了。其次，通过断链 9 年级和 12 年级两个社区之间的一些链接，我们基本上可以正确划分这两个年级了（如学生 14，18 和 20 都被正确分类了）。但依然有一些 9 年级的学生被错误分类，这主要原因是在这个年级中并不存在明显的核心节点。这就导致我们无法增加生社区内部的链接。如4.7节中图4.6所示，当我们令这个网络的社区个数为 7 的时候，我们仅仅使用连接策略就可以将网络中所有节点的社区都划分正确。

如图4.12所示，在美国政治书网络中，我们的算法依然有着非常优越的性能。这个网络的最大特点是，他仅有两个连接紧密的社区结构，即保守派和自由派图书。而真实世界中，却将其划分为还包括中立派图书在内的三个社区。而这个中立派图书社区是前面两个社区的重叠部分。可以发现原始的基于网络拓扑结构的非负矩阵分解算法在划分这个重叠社区及处理社区边界的问题上无法得到令人满意的结果。而通过将边界处的节点与社区核心节点相连接（红色实现）并将社区间的链接断开（黄色虚线），主动链接选择策略可以非常好的解决这两个问题。

上面的 4 个例子展示了主动链接选择策略的工作原理并充分说明了它的高效性。这个高效性主要源自高效的链接选择策略和高效的成对监督信息的利用策略。首先，连接策略可以选择对性能影响最显著的关键性链接供人工标注。大部分被选出的链接都处于社区的边界处。这也说明了为什么我们提出的框架能够很好的解决之前方法在边界处划分错误的问题。其次，断连策略可以进一步充分利用获得的成对监督信息来将社区之间的链接断开。虽然这个过程会有一些错误的断链，但是绝大多数的断链错误都是正确的并对最终的性能起到了促进的作用。从上面四个例子的图示我们也可以看出我们删除的边（黄色虚线）几乎都对传统基于拓扑结构的社区检测性能起负面作用的边。

4.4 小结

在这篇文章中，我们为高效的半监督社区发现提出了一个主动的链接选择框架。这个框架由连接策略和断连策略两部分组成。通过使用连接策略，我们根据非

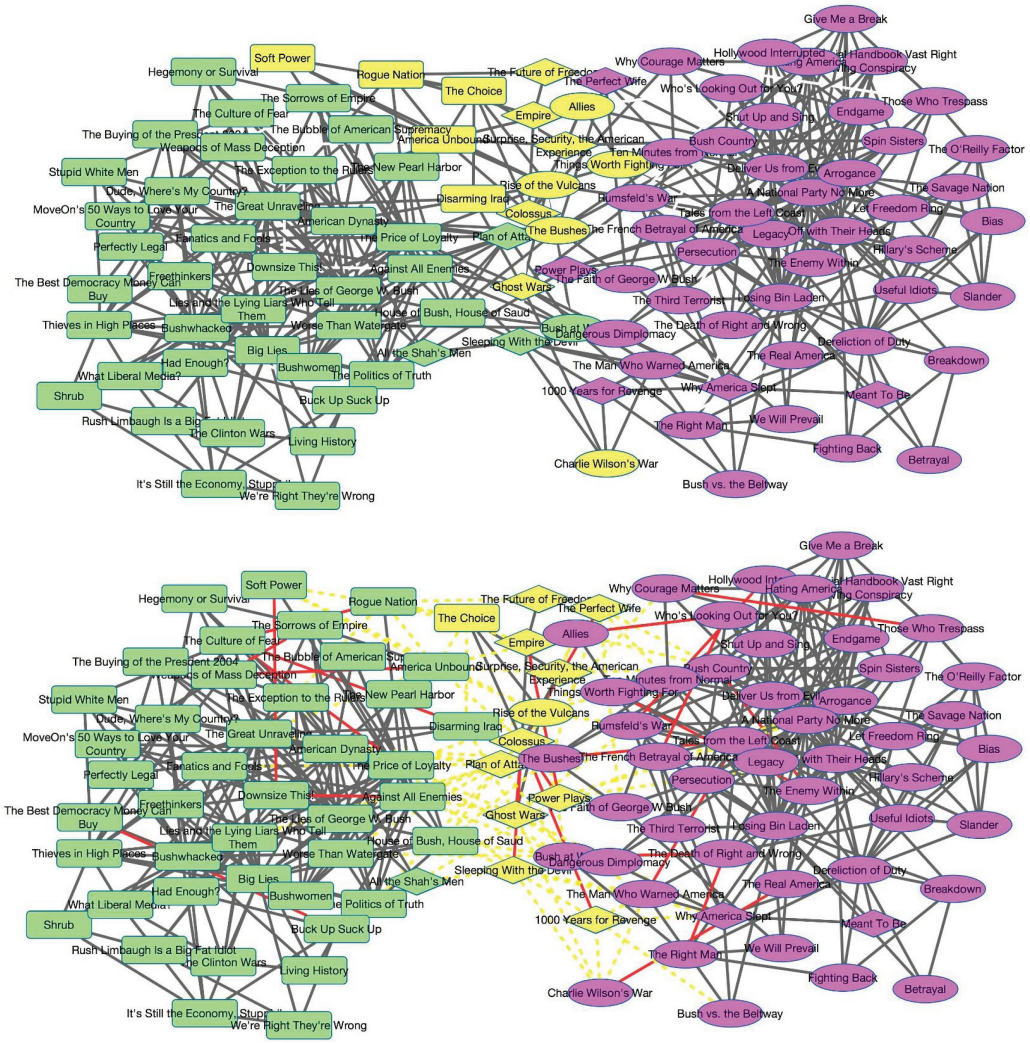


图 4.12: 在使用我们的主动链接选择策略前后的美国政治书网络的网络拓扑结构和检测到的社区结构。

负矩阵分解的结果交互的选择最不确定的边供人工标注，并将对应的节点与社区核心节点相连。为了进一步挖掘加入的成对监督信息，我们设计的断连策略再断开可能是社区之间的链接。这两个策略联合的高效锐化网络邻接矩阵的块结构。之后基于标注的链接，我们更新网络拓扑结构并运行下一轮的非负矩阵分解。和传统的通过加入随机选择的监督信息的半监督社区发现方法相比，我们提出的框架主动的选择最不确定的链接供标注。这显著的减少了对监督信息的需求并明显的提升了半监督社区发现的效率。此外，我们为了加速框架迭代

过程中的非负矩阵分解，我们提出了一种可以加速十倍以上的加速策略。在两类人工网络和一些广泛使用的真实网络上的实验验证了我们算法的高效性。

第五章　自动化社区发现

5.1　引言

如前所述，非负矩阵分解方法经常被用于作为半监督社区发现及其变种的核心组成部分，如我们在第三章和第四章中提出的基于隐空间图正则的半监督统一社区发现框架和基于主动链接选择的高效社区发现算法。但是由于用于社区发现的非负矩阵分解需要提前将待检测的社区个数作为分解后的矩阵的列数，因此社区个数的确定将很大程度上影响基于拓扑结构的社区检测算法和基于先验信息的半监督社区发现算法的性能及可解释性。但是在现实生活中，社区个数在检测社区结构之前通常是很难确定的，甚至是无法确定的。这就导致虽然非负矩阵分解在假设社区个数已知的情况下有着出色的表现，但是在实际应用中却显得不够实用。这也导致传统的基于非负矩阵分解的社区发现算法都不能完全自动的完成，即给定拓扑结构直接获得网络中的社区个数及对应的社区结构。

非负矩阵分解作为一种旨在将数据分解为嵌入在其中的隐含的成分和模式的方法已经被广泛的应用于很多领域，如图像处理 [37, 117]、音频处理 [9, 85] 和文本挖掘 [102, 123] 中。本质上，非负矩阵分解是用一个已知秩（即成分或者社区的个数）的低维近似来逼近原始的数据。这种想法之后被扩展到在未知秩的情况下，直接得到其内部的低维表达。鲁棒的主成分分析（Robust PCA）[13] 就是其中最为知名的一种方法。但是可惜的是，由于在数据的结构上加入了限制，即要求数据中的噪音要是稀疏的且清晰数据的秩要充分的低，而这些要求在现实中通常很难满足，这就导致鲁棒主成分分析通常获得的表达并不是低秩的，从而影响了其自动得到成分个数的准确性。

我们提出了一种自动加权低秩约束的非负矩阵分解算法来进行自动的社区发现。一种直观的想法是首先初始化社区个数远远大于真实社区个数，并在社区有检测过程中合并部分社区。主要的想法是首先将分解后的矩阵列数初始化为远大于社区个数，并通过加权的低秩约束将其中的一些列压缩零。具体的说，我们首先证明通常低秩正则项，即核模，等价于同时地同等程度的将分解后的矩阵的每列压缩到零。因此，为了使一部分列被压缩到零，我们就需要加入一个自适应加权的低秩约束。这就等价于在不同的列加入不同的压缩权重。如果

权重较大，相应的列更容易被压缩到零，反之亦然。那么剩下的问题就是如何确定这些权重。这里我们并不是通过人工的方式设定这些参数，而是在这些参数的分布上加入了两个约束。1）不能每个权重都很小，否则加权的低秩约束就起不到应有的作用。2）只能保留一部分的权重很大，因为我们只希望将部分列压缩到零。最后的优化函数是组合了非负矩阵分解的重构误差、加权低秩约束和在权重分布上的约束。通过最小化目标函数，我们可以使分解后的矩阵具有列稀疏的特性。最终非零列的个数即为社区的个数，且我们可以根据非零列将节点分到不同的社区中。我们的算法有两大优势。1）我们可以在不增加非负矩阵分解算法复杂度的同时确定社区个数。2）系统中只有一个容易调节的参数，且这个参数取 1 或 2 时便可以得到令人满意的结果。

5.1.1 相关工作

虽然很多的工作都指出确定数据内在的秩是非常重要的一个问题，但是由于其困难性，现有的解决此问题的方法却非常少。这些方法可以被大致的分为两类。第一类方法不是直接确定数据的秩，而是通过评估每一个候选的秩的可能性，如通过马尔科夫链蒙特卡洛采样 [15] 或使用可逆跳步马尔科夫链蒙特卡洛方法 [118]。这类采样方法对计算量相对较大，因此很难用到大规模问题中。第二类方法通过使用各类贝叶斯方法来获得矩阵的秩。一些方法假设分解后的矩阵中的元素符合指数先验 [60, 61]，其他一些方法则假设他们符合伽马先验 [9]。如贝叶斯非负矩阵分解方法（BNMF）[87] 假设重构误差，分解后的矩阵元素及参数分别服从泊松分布，半正态分布和高斯分布。这类基于贝叶斯的统计分析的方法主要存在两个比较大的缺陷。一方面他们会引入过多的参数，这些参数的调节对最终的结果有很大的影响。虽然通常都会给出一些调参的建议，但是由于参数之间存在耦合性，通常很难直接被应用到实际中。另一方面，贝叶斯方法普遍缺乏对问题内在原理的可解释性，这为最后的参数调节和结果分析带来了进一步的困难。

5.2 低秩矩阵近似

由于绝大多数的数据都不是纯净的，而是被各种噪声所侵蚀的，如何去除这些噪声并揭示数据的真实结构在很多领域都是非常重要的问题，如信号处理、计算机视觉和模式识别。低秩近似是一类被广泛采用的获取给定数据的内在结构

方法。这些方法都是基于如下的假设：数据的内在结构处于一个低维的子空间中，而实际观测到的数据是被噪音所侵蚀的高维数据。低秩近似的方法可以被分为已知秩的近似和未知秩的近似两类。

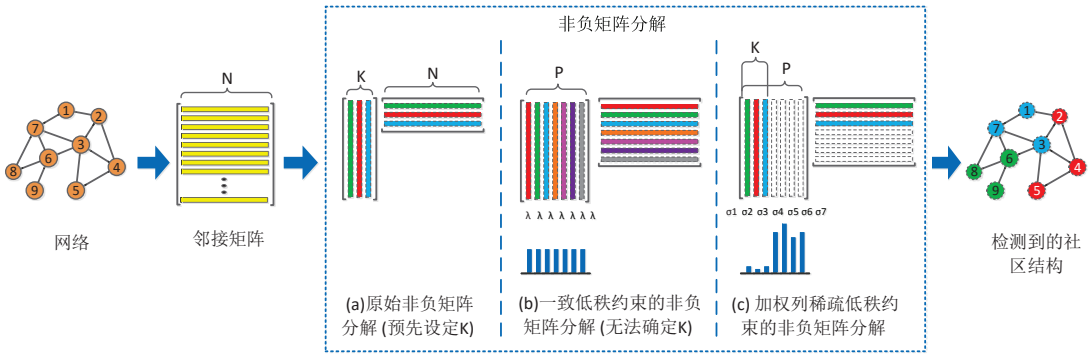


图 5.1: 三种非负矩阵分解方法的比较。

5.2.1　已知秩的低秩近似

给定一个具有 N 个数据（节点）的数据集（网络），每个数据由一个 F 维的特征表示。数据集可以用一个 $\mathbf{X} \in \mathbb{R}^{F\times N}$ 的矩阵表示，其中每列代表一个数据，每行代表一维特征。我们可以通过最小化

$$\underset{\mathbf{U},\mathbf{V}}{\operatorname{argmin}}\, \mathrm{Dis}(\mathbf{X}, \mathbf{U}\mathbf{V}^T), \tag{5.1}$$

来将 $\mathbf{X}$ 分解为两个低维矩阵 $\mathbf{U} \in \mathbb{R}^{F\times K}$ 和 $\mathbf{V} \in \mathbb{R}^{N\times K}$ 的乘积从而近似原始的 $\mathbf{X}$。其中 $K \ll \min(F, N)$ 是已知的隐空间的维数或者潜在数据的秩。$\mathrm{Dis}(\mathbf{X}, \mathbf{U}\mathbf{V}^T)$ 代表原始数据 $\mathbf{X}$ 和重构数据 $\mathbf{U}\mathbf{V}^T$ 之间基于某些特定距离度量的误差。常用的距离有 KL 散度、ℓ_1 模距离和 Frobenius 模（ℓ_2 模）距离等。非负矩阵分解是一个著名的已知秩的低秩矩阵近似的方法，其要求分解后的矩阵 $\mathbf{U}$ 和 $\mathbf{V}$ 中的所有元素是非负的。除此之外，$\mathbf{U}$ 和 $\mathbf{V}$ 在很多不同的应用中有着不同的含义。如在人脸聚类算法中 [37]，$\mathbf{U}$ 的列被因为是图像的一组基（隐空间字典元素），$\mathbf{V}$ 的每一列被称为原始元素基于 $\mathbf{U}$ 中所蕴含的基的新的表示。相比于原始的 F 维特征，这个新的 K 维表达对于聚类问题来说更加的高效和鲁棒。此类方法的最大问题在于隐空间的维数 K 必须提前设定。而在现实应用中，这是非常困难的。如图5.1(a) 所示。

5.2.2 未知秩的低秩近似

当数据内在的秩未知时，我们可以通过同时对数据内在的秩和数据的重构误差进行建模，并通过平衡两者的之间的关系来近似原始的数据 $\mathbf{X}$，一个通用的范式如下所示：

$$\underset{\mathbf{A}}{\operatorname{argmin}} \operatorname{Dis}(\mathbf{X}, \mathbf{A}) + \lambda \operatorname{rank}(\mathbf{A}),$$

其中 $\operatorname{rank}(\mathbf{A})$ 代表矩阵 $\mathbf{A}$ 的秩，λ 是平衡两项之间权重的参数。通常由于求秩的函数是非凸的，我们可以通过优化其凸包来得到其低秩近似：

$$\underset{\mathbf{A}}{\operatorname{argmin}} \operatorname{Dis}(\mathbf{X}, \mathbf{A}) + \lambda ||\mathbf{A}||_*, \tag{5.2}$$

其中 $||\mathbf{A}||_*$ 是矩阵 $\mathbf{A}$ 的核模 [57]，即 $\mathbf{A}$ 的所有奇异值的和。最为知名的未知秩的低秩矩阵近似是鲁棒的主成分分析 (Robust PCA，RPCA [13])。这种方法已经被广泛的应用到包括背景抽取 [120, 121]、纹理修复 [45, 113] 和子空间分割 [50, 51] 等计算机视觉问题中。RPCA[13] 使用 ℓ_0 模来度量原始数据 $\mathbf{X}$ 和低秩近似矩阵 $\mathbf{A}$ 之间的差异，即 $\operatorname{Dis}(\mathbf{X}, \mathbf{A}) = ||\mathbf{X} - \mathbf{A}||_0$，其中 $||\mathbf{X}||_0$ 是 $\mathbf{X}$ 的非零元素的个数。与用核模来代替秩函数相同，由于 $||\mathbf{X}||_0$ 不是凸的，因此常用其紧致的凸包函数 $||\mathbf{X}||_1$ 来进行代替。从而得到在一定范围内（相对于 $\mathbf{A}$ 的秩误差矩阵 $\mathbf{X} - \mathbf{A}$ 充分稀疏）等价的两个优化问题：

$$\underset{\mathbf{A}}{\operatorname{argmin}} ||\mathbf{X} - \mathbf{A}||_0 + \lambda \operatorname{rank}(\mathbf{A}),$$
$$\underset{\mathbf{A}}{\operatorname{argmin}} ||\mathbf{X} - \mathbf{A}||_1 + \lambda ||\mathbf{A}||_*.$$

这类方法虽然能够在未知真实秩的情况下获得低秩矩阵，但是他们也有其固有的缺陷。首先就是这类方法一般对数据的限制比较严格，即噪音要相对稀疏且真实数据的秩要充分的低。在真实应用中，由于在不满足前面约束的前提下最小化秩函数和其凸包核模函数并不是等价的，因此通常无法得到低秩的矩阵 $\mathbf{A}$，从而也很难得到其真实的秩。其次，即使能获得除去噪音后的真实数据，此类方法也不能如非负矩阵分解那样直接得到隐空间中的字典和基于这个字典的全新表达。而这个全新的隐空间表达对包括社区发现在内的很多聚类问题非常重要。最后，虽然已经有工作指出在很宽松的条件下参数 λ 可以从理论上直接获得，但在真实应用中这个参数依然需要进行细致地人为调节来获得令人满意的结果。

5.3　自动加权的低秩矩阵近似

本章我们将要提出的自动加权的低秩约束非负矩阵分解（NMF-AWL）是在传统的非负矩阵分解的基础上增加了一个自动加权的低秩正则项。我们首先提出一个最为直接的确定的矩阵秩的方法，即在非负矩阵分解中加入一个无权重的低秩约束项。之后讨论这种直接的方法无法起效的原因并基于此提出我们的加权的列稀疏低秩约束非负矩阵分解。最后我们给出一个策略来自动确定每一列的权重系数。

5.3.1　等权低秩约束

一种最为直接的结合上一节中提到的两类低秩近似算法优点的方式是将未知秩的低秩近似的优化问题中的待求低秩矩阵分解为两个矩阵乘积的形式，如通过优化：

$$\underset{\mathbf{U},\mathbf{V}}{\operatorname{argmin}} \operatorname{Dis}(\mathbf{X}, \mathbf{U}\mathbf{V}^T) + \lambda ||\mathbf{U}\mathbf{V}^T||_*, \tag{5.3}$$

将公式（5.2）中的 $\mathbf{A}$ 分解为 $\mathbf{U} \in \mathbb{R}^{F\times P}$ 和 $\mathbf{V} \in \mathbb{R}^{N\times P}$。其中 $P \gg K$ 和 $P \leq \min(F, N)$ 是预先定义的分解后的矩阵 $\mathbf{U}$ 的最大列数。需要注意的是，这个预先设定列数通常可以远远大于最终的社区个数（最终的非零列数）。公式（5.3）也可以被认为是将未知秩的低秩近似中的秩函数（或松弛后的核模函数）直接加到已知低秩近似方法的优化函数中，即在公式（5.1）中加入正则项 $\lambda||\mathbf{U}\mathbf{V}^T||_*$。通过这样做，我们希望得到的矩阵 $\mathbf{U}$ 的秩可以像在 RPCA 中一样可以自动地被确定。

5.3.2　加权低秩约束

上一小节中讨论的公式（5.3）相当于先对数据矩阵运行未知秩的低秩矩阵近似，然后对得到的低秩矩阵再进行分解操作。但是如5.2.2节中所述，通常未知秩的低秩矩阵近似很有可能得到的矩阵不是低秩的，这就导致，分解后的矩阵 $\mathbf{U}$ 也不是低秩的。更进一步，即使分解后的矩阵 $\mathbf{U}$ 是低秩的，可能依然非常困难确定哪些列应该被挑选出来作为数据的新的表达（如在社区发现中的隶属度关系）。因此，我们希望能够使到的矩阵具有列稀疏的特点，从而有助于选择非零列作为最终的隶属度矩阵。这里，我们首先解释为什么原始的低秩正则项无法

得到列稀疏的矩阵。我们可以将公式（5.3）重写为：

$$\underset{\mathbf{U},\mathbf{V}}{\operatorname{argmin}} \operatorname{Dis}(\mathbf{X}, \mathbf{U}\mathbf{V}^T) + ||(\mathbf{U}(\sqrt{\lambda}\mathbf{I}))(\mathbf{V}(\sqrt{\lambda}\mathbf{I})^T||_*, \tag{5.4}$$

其中 $\mathbf{I}$ 是一个 P 维的单位阵。由于 $\mathbf{U} = [\mathbf{u_1}, \mathbf{u_2}, ..., \mathbf{u_P}]$，其中 $\mathbf{u_i}$ 是 $\mathbf{U}$ 的第 i 列，因而 $\mathbf{U}(\sqrt{\lambda}\mathbf{I}) = [\sqrt{\lambda}\mathbf{u_1}, \sqrt{\lambda}\mathbf{u_2}, ..., \sqrt{\lambda}\mathbf{u_p}]$。这就意味着在公式 (5.3) 中加入的低秩约束一致地使 $\mathbf{U}$ 的每一列被压缩到零，我们可以通过下面的定理来正式的证明这一假设。

引理 1. *对于任意的矩阵 $\mathbf{A} \in \mathbb{R}^{F\times N}$ 有如下的等式*

$$||\mathbf{A}||_* = \min_{\mathbf{U},\mathbf{V},\mathbf{A}=\mathbf{U}\mathbf{V}^T} \frac{1}{2}(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2).$$

且当 $\operatorname{rank}(\mathbf{A}) = K \leq \min(F, N)$ 时，最小值对应的分解是 $\mathbf{A} = \mathbf{U}\mathbf{V}^T$ 其中 $\mathbf{U} \in \mathbb{R}^{F\times K}$ 且 $\mathbf{V} \in \mathbb{R}^{N\times K}$.

基于引理1，我们有如下的结果

定理 2. *如果 $\mathbf{A}$ 是优化问题（5.2）的解，$(\mathbf{U}, \mathbf{V})$ 是如下最小化问题*

$$\underset{\mathbf{U},\mathbf{V}}{\operatorname{argmin}} \operatorname{Dis}(\mathbf{X}, \mathbf{U}\mathbf{V}^T) + \frac{\lambda}{2}(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2),$$

的解，则 $\mathbf{A} = \mathbf{U}\mathbf{V}^T$。

证明. 根据引理1，我们有

$$\begin{aligned}
&\underset{\mathbf{A}}{\operatorname{argmin}} \operatorname{Dis}(\mathbf{X}, \mathbf{A}) + \lambda||\mathbf{A}||_* \\
\Leftrightarrow &\underset{\mathbf{A}}{\operatorname{argmin}} \left(\operatorname{Dis}(\mathbf{X}, \mathbf{A}) + \min_{\substack{\mathbf{U},\mathbf{V} \\ \mathbf{A}=\mathbf{U}\mathbf{V}^T}} \frac{\lambda}{2}(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2) \right) \\
\Leftrightarrow &\underset{\mathbf{A}}{\operatorname{argmin}} \min_{\substack{\mathbf{U},\mathbf{V} \\ \mathbf{A}=\mathbf{U}\mathbf{V}^T}} \left(\operatorname{Dis}(\mathbf{X}, \mathbf{U}\mathbf{V}^T) + \frac{\lambda}{2}(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2) \right) \\
\Leftrightarrow &\underset{\mathbf{U},\mathbf{V}}{\operatorname{argmin}} \left(\operatorname{Dis}(\mathbf{X}, \mathbf{U}\mathbf{V}^T) + \frac{\lambda}{2}(||\mathbf{U}||_F^2 + ||\mathbf{V}||_F^2) \right).
\end{aligned}$$

因此我们有 $\mathbf{A} = \mathbf{U}\mathbf{V}^T$。 □

因此, 最小化 $\lambda||\mathbf{U}\mathbf{V}^T||_*$ 等价于最小化 $\frac{\lambda}{2}(||\mathbf{U}||_F^2)+||\mathbf{V}||_F^2)$, 并且

$$\lambda||\mathbf{U}||_F^2=||\mathbf{U}(\sqrt{\lambda}\mathbf{I})||_F^2=\lambda\sum_{t=1}^{P}||\mathbf{u_t}||_2^2=\sum_{t=1}^{P}||\sqrt{\lambda}\mathbf{u_t}||_2^2.$$

这样的结果是符合我们前面的直观的假设的，即引入一个低秩约束等价于一致地使矩阵 $\mathbf{U}$ 和 $\mathbf{V}$ 的每列坍缩到零。因此，直接优化公式（5.3）中的低秩约束的非负矩阵分解无法得到一个列稀疏的矩阵 $\mathbf{U}$ 。如图5.1(b) 所示。

为了得到列稀疏的矩阵 $\mathbf{U}$，我们的主要想法是引入一个可以有选择性地压缩 $\mathbf{U}$ 的一部分列到零的低秩正则项。为此，我们为不同的列使用不同的权重，即使用如下的正则项来代替 $\lambda||\mathbf{U}||_F^2$

$$||\mathbf{U}\Sigma^{\frac{1}{2}}||_F^2=\sum_{t=1}^{P}\sigma_t||\mathbf{u_t}||_2^2=\sum_{t=1}^{P}||\sqrt{\sigma_t}\mathbf{u_t}||_2^2, \tag{5.5}$$

其中 $\Sigma=\mathrm{diag}([\sigma_1,\sigma_2,...,\sigma_P])$，是一个 P 维的对角矩阵，它的第 i 个对角线元素是 σ_i，且非对角线元素全为 0。这个对角矩阵用于指定不同列的压缩程度。如果 $\sigma_i>\sigma_j$，当最小化公式（5.5）中的问题时，第 i 列将比第 j 列更强的被压缩。如图5.1(c) 所示。因此，公式（5.4）中的优化问题可以被推广到如下的加权低秩约束非负矩阵分解问题

$$\begin{aligned}&\underset{\mathbf{U},\mathbf{V}}{\mathrm{argmin}}\,\mathrm{Dis}(\mathbf{X},\mathbf{U}\mathbf{V}^T)+||(\mathbf{U}\Sigma^{\frac{1}{2}})(\mathbf{V}\Sigma^{\frac{1}{2}})^T||_*\\=\;&\underset{\mathbf{U},\mathbf{V}}{\mathrm{argmin}}\,\mathrm{Dis}(\mathbf{X},\mathbf{U}\mathbf{V}^T)+||\mathbf{U}\Sigma\mathbf{V}^T||_*.\end{aligned} \tag{5.6}$$

剩下的问题就是如何确定 Σ 中的权重。

5.3.3 自适应权重设定

由于 Σ 矩阵中有 P 个参数需要确定，因此很难通过人工来完成，故需要设计一种可以自动从数据中学习这 P 个参数的策略。我们首先做出如下的两个假设。一方面，不能每个 σ_i 都很小，否则正则项就无法起到应有的作用，整个优化问题就退化到原始的已知秩的低秩矩阵分解问题。另一方面，为了能够使一部分列被更强的压缩到零，一些 σ_i 应该比另外的一些更大，也就是说不能所有的 σ_i 都很大，否则所有的列都将被压缩到零。

为了使 Σ 满足这些条件，我们这里引入一个惩罚函数

$$F(\Sigma) = G(\Sigma) + H(\Sigma), \tag{5.7}$$

其中 $G(\Sigma)$ 用于防止所有的 σ_i 都太小，$H(\Sigma)$ 用于防止所有的 σ_i 都太大。

为了符合第一条准则，$G(\cdot)$ 应该是一个单调非递增函数。为了简单起见，我们应该选择一个可微的函数。进一步，为了使求解更加方便，我们假设我们能单独的惩罚每个 σ_i 并且当 $\sigma_i = 1$ 的时候惩罚为 0。两个满足这些要求的较为简单的函数是：

$$\begin{aligned} G_1(\Sigma) &= -\beta \log_b(|\Sigma|) = \beta \sum_{t=1}^{P} -\log_b(\sigma_t) = \beta \sum_{t=1}^{P} g_1(\sigma_t), \\ G_2(\Sigma) &= \beta \sum_{t=1}^{P} \left(\frac{1}{\sigma_t} - 1 \right) = \beta \sum_{t=1}^{P} g_2(\sigma_t), \end{aligned} \tag{5.8}$$

其中 β 是一个用来确定 $G(\cdot)$ 的强度的正常数，$|\Sigma|$ 代表 Σ 的行列式。图5.2展示了不同 b 值对应的函数 $g_1(\sigma)$ 形状。由于 Σ 是一个对角矩阵，其行列式的值为 $|\Sigma| = \prod_{t=1}^{P} \sigma_t$。其中，我们固定参数 2β 为分解后的矩阵 $\mathbf{U}$ 和 $\mathbf{V}$ 的总列数，即 $2\beta = F + N$。这样做的目的将在5.4.2节中进行讨论。在后面的讨论中，我们将 $g_1(\cdot)$ 和 $g_2(\cdot)$ 统称为 $g(\cdot)$，并将验证在实际应用中自然对数（$b = e$）通常能得到令人满意的结果。

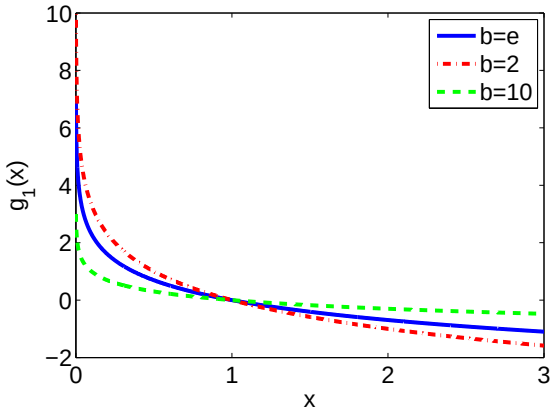


图 5.2: 损失函数 $g_1(x)$ 在不同底数 b 的时候曲线。当 b 较大时，曲线快速下降到 0 并且有一个较平的尾部，反之当 b 较小时，起初下降的较慢且尾部较陡峭。

另一方面，为了符合第二条准则，$H(\cdot)$ 应该是一个单调非递减函数。为了简单起见，我们用 Σ 的迹作为 $H(\cdot)$，即

$$H(\Sigma) = \alpha \operatorname{Tr}(\Sigma) = \alpha \sum_{t=1}^{P} \sigma_t = \alpha \sum_{t=1}^{P} h(\sigma_t), \tag{5.9}$$

其中 $\operatorname{Tr}(\cdot)$ 代表矩阵的迹，α 是一个确定 $H(\cdot)$ 的贡献程度的正常数。由于 $H(\cdot)$ 的作用相对 $G(\cdot)$ 要小很多，我们应该选择一个较小的 α。除非特别说明，在后面的讨论和实验中我们设 $\alpha = 1$。α 的角色也将在5.4.2节中进行详细的讨论。

综合前面关于的讨论，我们有：

$$\begin{aligned} F(\Sigma) &= \beta \sum_{t=1}^{P} g(\sigma_t) + \alpha \sum_{t=1}^{P} h(\sigma_t) \\ &= \sum_{t=1}^{P} \big(\beta g(\sigma_t) + \alpha h(\sigma_t)\big) = \sum_{t=1}^{P} f(\sigma_t) \end{aligned} \tag{5.10}$$

其中，$f(\sigma)$ 是为了使 σ 满足前面准则的惩罚函数。我们通常设置 $\alpha = \beta = 1$，并且为了使 $f(\sigma)$ 在 $\sigma = 1$ 时达到其最小值，我们令 $b = e$，即为自然对数。图5.3展示了不同 b 值对应的函数 $f(\sigma)$ 形状。

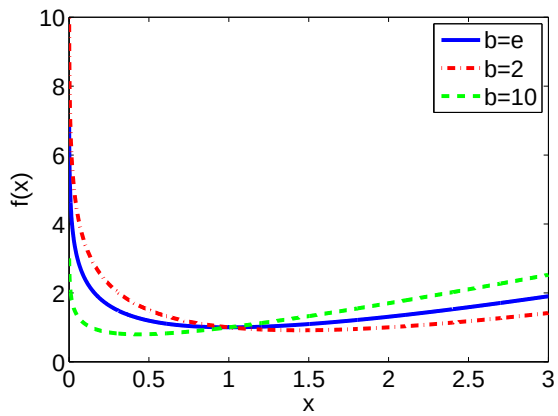


图 5.3: 损失函数 $f(x)$ 在不同底数 b 时的曲线。较大的 b 使 $f(x)$ 在较小的 x 处达到最小值，反之则在较大的时候达到最小值。因此为了使 $f(x)$ 在 $x = 1$ 时达到最小值，我们令 $b = e$。

5.3.4 目标函数与优化

在上一节中我们分别给出了加权的低秩约束非负矩阵分解的目标函数和求解权重的惩罚函数。在这一节，我们基于这两个模型，推导自动加权低秩约束非负矩阵分解的最终目标函数及相应的优化算法。为了能使分解后的矩阵是列稀疏的，加权低秩约束非负矩阵分解的目标函数和其中的权重惩罚函数应该能够相互融合，我们令最终的目标函数为两个目标函数的和。因此，整体的目标函数有两部分组成，即公式（5.6 ）中的加权组稀疏低秩约束非负矩阵分解和公式（5.10）中的权重惩罚函数。在这里我们使用 KL 散度作为原始数据和重构数据之间的距离度量标准，并为了结果的可解释性约束分解后的矩阵中的元素为非负。最终的目标函数为：

$$\underset{\mathbf{U}>\mathbf{0},\mathbf{V}>\mathbf{0},\ >\mathbf{0}}{\operatorname{argmin}} \mathcal{L}_{KL}(\mathbf{X}||\mathbf{U}\mathbf{V}^T) + ||\mathbf{U}\ \mathbf{V}^T||_* + f(\Sigma). \tag{5.11}$$

这里我们使用

$$\mathcal{L}_{KL}(\mathbf{X}||\mathbf{U}\mathbf{V}^T) = \sum_{i,j}\left(x_{ij}\log\left(\frac{x_{ij}}{y_{ij}}\right) - x_{ij} + y_{ij}\right), \tag{5.12}$$

其中 $\mathbf{X} = [x_{ij}]$，$\mathbf{Y} = [y_{ij}] = \mathbf{U}\mathbf{V}^T$。根据定理1和公式（5.12），（5.8）和（5.9），整体目标函数（5.11）可以被写为：

$$\begin{aligned}
\mathcal{L}(\mathbf{U},\mathbf{V},\Sigma) &= \mathcal{L}_{KL}(\mathbf{X}||\mathbf{U}\mathbf{V}^T) + \alpha\sum_{t=1}^{p}\sigma_t + \beta\sum_{t=1}^{p}g(\sigma_t) + \frac{1}{2}\sum_{t=1}^{P}\sigma_t(||\mathbf{u_t}||_2^2 + ||\mathbf{v_t}||_2^2) \\
&= \sum_{i,j}\left(x_{ij}\log\left(\frac{x_{ij}}{y_{ij}}\right) - x_{ij} + y_{ij}\right) \\
&+ \sum_{k=1}^{P}\left(\sum_i u_{ik}^2 + \sum_j v_{jk}^2 + \beta\right)\sigma_k + \alpha g(\sigma_k)
\end{aligned} \tag{5.13}$$

和基于 KL 散度的非负矩阵分解一样，由于整体目标函数 $\mathcal{L}(\mathbf{U},\mathbf{V},\Sigma)$ 并不是同时对 $\mathbf{U}$，$\mathbf{V}$ 和 Σ 是凸的，因此非常困难找到整体的最优解。我们设计了一种类似于非负矩阵分解中的交替乘法更新规则2.8的迭代优化算法。此类算法可以快速的达到局部最优解。整体的过程由两步组成。第一步我们计算目标函数

$\mathcal{L}(\mathbf{U},\mathbf{V},\Sigma)$ 相对于 $\mathbf{U}$，$\mathbf{V}$ 和 Σ 的梯度。第步，我们通过将他们现在的值与梯度的正负部分比例相乘来更新 $\mathbf{U}$ 和 $\mathbf{V}$。$\mathbf{U}$ 和 $\mathbf{V}$ 的更新规则如下：

$$u_{it} \leftarrow u_{it}\frac{\sum_j(x_{ij}v_{jt}/\sum_t u_{it}v_{jt})}{\sigma_t u_{it}+\sum_j v_{jt}}, \tag{5.14}$$

$$v_{jt} \leftarrow v_{jt}\frac{\sum_i(x_{ij}u_{it}/\sum_t u_{it}v_{jt})}{\sigma_t v_{jt}+\sum_i u_{it}}. \tag{5.15}$$

我们通过令 $\mathcal{L}(\mathbf{U},\mathbf{V},\Sigma)$ 对 Σ 的梯度为 0 来求出在给定 $\mathbf{U}$ 和 $\mathbf{V}$ 的情况下 Σ 的解析解。使用 $G_1(\cdot)$ 或 $G_2(\cdot)$ 的 Σ 更新规则如下

$$\sigma_t \leftarrow \frac{\beta}{\left(\frac{1}{2}(||\mathbf{u_t}||_2^2+||\mathbf{v_t}||_2^2)+\alpha\right)\ln b}, \tag{5.16}$$

$$\sigma_t \leftarrow \sqrt{\frac{\beta}{(||\mathbf{u_t}||_2^2+||\mathbf{v_t}||_2^2)/2+\alpha}},$$

其中 b 是对数的底，α 和 β 是需要设定的两个参数，我们将在下一节进行讨论。

5.3.5　参数设定与含义

基于我们后面5.4.2节的实验，$b=e$ 时，即取自然对数时通常能得到令人满意的结果，因此我们将在本节固定 $b=e$ 的情况下讨论参数 α 和 β 的作用。当 $b=e$ 且 $\alpha=0$ 时，公式（5.16）可以被重写为

$$\sigma_t \leftarrow \frac{2\beta}{||\mathbf{u_t}||_2^2+||\mathbf{v_t}||_2^2}. \tag{5.17}$$

这意味着如果 $\mathbf{U}$ 和 $\mathbf{V}$ 的第 k 列的 ℓ_2 模很小，相应的权重 σ_t 将会很大，从而这两列将在下一次迭代中被更加严厉的压缩到 0。如果我们令 $\beta=(F+N)/2$，σ_t 则是 $\mathbf{u_t}$ 和 $\mathbf{v_t}$ 中所有 $F+N$ 个元素平方平均数的倒数，即

$$\sigma_t \leftarrow \frac{1}{\frac{1}{F+N}(\sum_{i=1}^m u_{it}^2+\sum_{j=1}^n v_{jt}^2)}.$$

这表明 σ_t 是否大于 1 是由 $\mathbf{u_t}$ 和 $\mathbf{v_t}$ 中元素的平方平均数是否小于 1 决定的。

其次，当 $\alpha=0$ 时，如果第 t 列被压缩到 0，σ_t 将趋近于无穷。为了避免这种情况的发生，我们令 α 是一个比 β 更小的数。这与我们前面在5.3.3节中讨论设定 Σ 时所说的由于第一个准则（使一些列被压缩到 0）的重要性大于第二个准则（并不是所有列都被压缩到 0）相一致。因此第一项的系数 β 要大于第

二项的系数 α。如我们的实验结果中所示，α 可以被简单的设置为 1 来得到令人满意的结果。从公式（5.16）可见, 小的 α 使得大部分 σ_t 更大，所以更多的列将被压缩到 0，这将导致较小的社区个数。因此，如果我们希望将网络划分为较小的社区，我们应该适当的增大 α。我们将在5.4.4节进一步验证这一点。

最后，根据公式（5.14）和（5.15），我们可以发现传统基于 KL 散度的非负矩阵分解和我们提出的自动加权低秩约束的非负矩阵分解在更新规则上唯一的不同在于分母上多了一项 $\sigma_t u_{it}$ 或 $\sigma_t v_{jt}$。由于 σ_t 是 $\mathbf{u_t}$ 和 $\mathbf{v_t}$ 元素平方平均数的倒数，如果 u_{it} 或 v_{jt} 大于这个平均数，$\sigma_t u_{it}$ 或 $\sigma_t v_{jt}$ 将大于 1，且公式（5.14）或（5.15）使 u_{it} 和 v_{jt} 在下一轮迭代中更小，即将其压缩到 0。

5.3.6 自动社区发现

一个具有 N 个节点的无向无权图 $G = (V, E)$ 可以用一个对阵非负二值矩阵 $\mathbf{X} = [x_{ij}] \in \mathbb{R}_+^{N\times N}$ 来表示。假设 $\mathbf{X}$ 的真实的秩是 K，且远远小于 N，即 $K \ll N$，我们可以设置初始的分解后矩阵的列数为 $P = N/2 \gg K$，且 α 等于 1。通常情况下，如果我们可以估计 K 的上界，我们可以将 P 设为这个上界。在之后的5.3.7节和5.4.2节中我们会看到，适当的设定 P 不仅会使计算量降低还不会对最终社区发现的结果有太大的影响。我们之后可以通过最小化 $\mathcal{L}(\mathbf{U}, \mathbf{V}, \Sigma)$ 将矩阵 $\mathbf{X}$ 分解为一个成分（社区隶属度矩阵）$\mathbf{U}$。最终的社区个数就是分解后 $\mathbf{U}$ 的非零列数，即 K。通过移除 $\mathbf{U}$ 中为 0 的列，最终的社区隶属度矩阵为 $\mathbf{U}^* = [u_{it}^*] \in \mathbb{R}^{N\times K}$。与常规的用于社区发现的非负矩阵分解方法一致，我们可以将节点划分到具有最高隶属度概率的那个社区中。算法1中给出了具体的算法。

5.3.7 时间复杂度分析

为了方便起见，我们只分析求解公式（5.14）和（5.16）的时间复杂度。我们将公式5.14重写为如下的形式：

$$u_{it} \leftarrow u_{it} \frac{\sum_j \left(\frac{x_{ij}}{\sum_t u_{it} v_{jt}} v_{jt}\right)}{\sigma_t u_{it} + \sum_j v_{jt}}. \tag{5.18}$$

由于矩阵 $\mathbf{U}$ 的大小为 NP，u_{it} 和分式之间的乘法要重复 np 次。分式的分母由两部分组成。第一部分，$\sigma_t u_{it}$ 将对与每一个 i 和 k 分别计算，因此需要 NP 浮点乘法运算。第二项 $\sum_j v_{jt}$ 将对每个 t 分别计算，因此需要 NP 浮点加法运算。

算法 1: 用于社区发现的自动加权低秩约束的非负矩阵分解算法 (NMF-AWL)

Input: 邻接矩阵 $\mathbf{X} \in \mathbb{R}^{N\times N}$, 初始分解后矩阵列数 $P = N/2$，$\beta = N$，通常情况下 $\alpha = 1$，分层社区检测时令 $\alpha \in \{1, 2, 3\}$

Output: 社区个数 K 和节点的隶属关系

初始化: $\Sigma = \mathbf{I}$, 随机矩阵 $\mathbf{U}, \mathbf{V} \in \mathbb{R}^{N\times P}$

for $t = 1 : n_{iter}$ **do**

　通过公式（5.14）更新 $\mathbf{U}$;

　通过公式（5.15）更新 $\mathbf{V}$;

　通过公式（5.16）更新 Σ;

end

社区个数 K 是 $\mathbf{U}$ 中 ℓ_2 模大于 10^{-200} 的列数。

通过去除 $\mathbf{U}$ 中模为 0 的列，我们得到最终的隶属度矩阵

$\mathbf{U}^* = [u^*_{ij}] \in \mathbb{R}^{N\times K}$.

如果 u^*_{ik} 是 $\mathbf{U}^*$ 的第 i 行中最大的元素，我们将第 i 个节点划分到第 t 个社区。

分子由一个对 N 个元素的求和组成，且要重复 NP 次，因此需要 N^2P 浮点加法运算。类似的，括号中的乘法也需要 N^2P 次浮点乘法运算。因为 $\frac{x_{ij}}{\sum_t u_{it}v_{jt}}$ 的计算与 k 无关，它只需要 N^2 次浮点除法运算。求和运算 $\sum_t u_{it}v_{jt}$ 要分别对于每个 i 和 j 进行计算，导致 N^2P 次浮点加法。整体上，公式（5.14）需要 $2N^P + NP$ 次浮点加法，$2N^P + NP$ 次浮点乘法和 $2N^2 + NP$ 次浮点除法，因此整体的实际复杂度是 $O(N^2P)$。综上所述，根据（5.14），（5.15）和（5.16），自动加权低秩约束的非负矩阵分解没有额外增加传统基于 KL 散度的非负矩阵分解的时间复杂度。

由于邻接矩阵 $\mathbf{X}$ 的稀疏性，N^2P 和 N^2 的操作都可以降为 MP 次和 M 次，其中 M 是网络中的边数，即 $\mathbf{X}$ 中非零元素的个数。例如，由于只有 M 个非零的 x_{ij}，我们不需要计算所有的 $\sum_t u_{it}v_{jt}$，而只需要计算针对 $x_{ij} \neq 0$ 的 i 和 j 进行计算。因此其时间复杂度降为 MP。故公式（5.14）的时间复杂度为 $O((M + N)P)$。

虽然我们迭代的更新公式(5.16)中的 σ_t，他的整体时间复杂度只有 $O(NP)$。因此在考虑稀疏性的前提下，自动加权低秩约束的非负矩阵分解的实际复杂度

为 $O((M+N)P)$。如果我们可以适当的选择 $P=O(K)$，即 P 和 K 是同阶的，自动加权低秩约束的非负矩阵分解算法对 M 和 N 是线性的。

进一步，从5.4.3节的验证性实验可以发现在迭代过程中，大多数的列都被研所到 0。因此，通过考虑 $\mathbf{U}$ 和 $\mathbf{V}$ 的列稀疏性，在初始迭代每次迭代的时间复杂度是 $O((M+N)P)$，而很快的就降为 $O((M+N)K)$。这个时间复杂度与初始的列的数目 P 无关。因此，我们算法的时间复杂度相对于网络规模是线性的，从而在大规模网络上是高效的。

值得注意的是，由于我们的算法和传统的基于 KL 散度的非负矩阵分解算法唯一的差别就是在 $\mathbf{U}$ 和 $\mathbf{V}$ 更新规则的分母上的 $\sigma_t u_{it}$ 或 $\sigma_t v_{jt}$。因此为非负矩阵分解设计的并行算法及分布式算法可以被轻易的移植到自动加权低秩约束的非负矩阵分解算法上，这使得我们的算法适用于更大规模的网络。

5.4 实验结果与分析

为了评价 NMF-AWL，我们在两类人工网络和表2.1中列出的一些真实网络上开展实验。我们将 NMF-AWL 与现有的四个最先进的自动化社区检测方法相比较。这些方法包括 louvain 算法 [10], 谱聚类算法 (SP) [68], 外部优化算法 (EO) [24] 和贝叶斯非负矩阵分解算法 (BNMF) [72]。我们采用第2.5节中所介绍的两个衡量标准来度量算法的性能。这两种标准分别是用来度量自动社区检测算法在确定社区个数和社区结构上的准确程度。

所有的实验都是在一台配置 Intel(R) Core(TM) i7-2600 CPU @ 3.40GHz. 处理器及 8G 内存的个人电脑上完成的。由于算法只收敛到局部最优解，对于每个任务我们用不同的随机初始值重复算法，并选择对应于最小化目标函数的结果。本章中用到的其他算法都是从作者本人网站上获取到的，并采用相应的默认设置。

5.4.1 人工网络

与前面两章一样，我们还是首先在 GN 网络和 LFR 网络上验证我们的算法。

如前所述，在 GN 网络中随着社区间边数 Z_{out} 的增多，社区结构变得模糊，检测社区结构变得困难。在这里我们对于所有不同 Z_{out} 的 GN 网络，统一的令参数 $\alpha=2$（$\alpha=1$ 时也可以得到类似的结果）。实验结果如图 5.4所示。图中的青色、品红、绿色、蓝色和红色分别代表外部优化算法、谱聚类算法、louvain

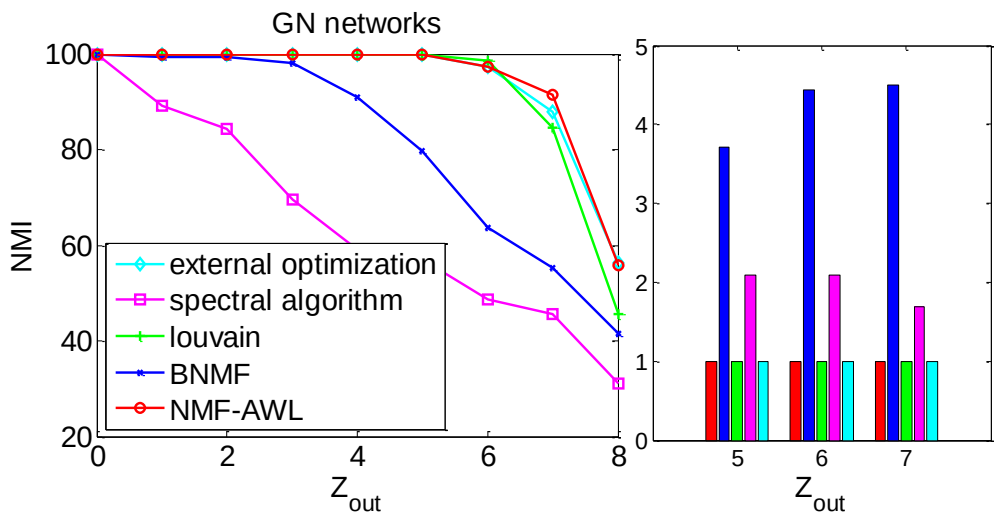


图 5.4: NMF-AWL 和 4 个现有的自动社区发现算法在 GN 网咯上的性能对比。

算法和贝叶斯非负矩阵分解算法以及我们提出的 NMF-AWL 算法的性能。左图显示不同算法在不同社区间边数 Z_{out} 的网络下基于 NMI 性能，其中值越大说明对社区的检测越准确。右图是根据公式2.16得到的检测到的社区个数和真实社区个数的差别，这个值越小说明算法在确定社区个数上越准确。从其中的右图可见当社区间边的数量 Z_{out} 为 6，7，8 时只有外部优化算法、louvain 算法和我们提出的 NMF-AWL 算法可以完全正确的确定社区个数。此外，从左图可以发现相比外部优化算法和 louvain 算法，NMF-AWL 算法在检测社区结构方面更具备优势，尤其是在具有较大 Z_{out} 值的 GN 网络上。

在 LFR 网络上，我们令混合参数 μ 从 0.1 变化到 0.6，来表示具有不同清晰程度的社区结构。对于所有的 LFR 网络，我们统一的令参数 $\alpha = 1$，并使邻接矩阵的对角线元素为对应节点的度。结果如图5.5所示。从右图中我们可以发现 NMF-AWL 比包括贝叶斯非负矩阵分解、外部优化算法及谱聚类在内的绝大多数算法有着更好的检测社区个数的能力。我们的算法只在 $\mu = 0.5$ 的网络上比 louvain 算法稍差。从左图中我们可以发现 NMF-AWL 在检测社区结构的准确度（基于 NMI）上超过了所有的方法，尤其是在具有较大混合参数 μ 的网络。

在 GN 和 LFR 网络上的实验结果充分说明了 NMF-AWL 方法在人工网络上无论是确定社区个数还是检测社区结构都超过了现有的方法。

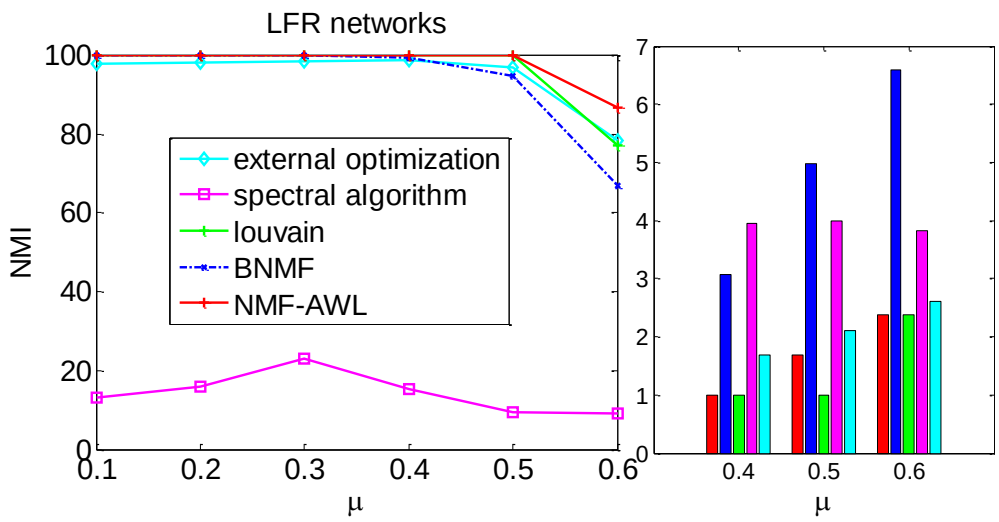


图 5.5: NMF-AWL 和 4 个现有的自动社区发现算法在 LFR 网络上的性能对比。衡量标准及符号标记与图5.4中一致。

5.4.2 参数设定

本节，我们将讨论模型中两个重要参数，即公式（5.8）中对数函数的底 b 和 α，对社区检测结果质量的影响。由于相比于 GN 网络，LFR 网络更贴近于真实网络的性质；而相对于真实网络，LFR 网络可以通过调节不同的参数（这里我们主要考虑混合参数 μ）来观察在不同结构的网络中参数设置影响对算法性能程度。此外，我们还将验证在邻接矩阵中对角线元素的设定方法对最终性能的影响。为此我们用三维柱状图来显式不同的 α 和 b 组合对最终性能（NMI）的影响。其中水平的二维坐标一个表示 b，另一个表示 α 与是否设定邻接矩阵对角线元素的组合。如图5.6中所示，在混合参数较小的 LFR 网络上（$\mu < 0.5$）大多数 α 和 b 的组合都可以得到令人满意的结果；且通常 b 在自然对数 e 的附近时可以获得最好的结果。为了更加清晰的观察在确定 $b = e$ 后，如何选择 α 及是否设定对角线元素，我们将图5.6中第二列的图在 $b = e$ 处进行二维切片，结果如图5.7所示。通过进一步对图5.7进行观察，我们可以发现通常最好的性能是在令邻接矩阵对角线元素为相应节点的度，且 $\alpha = 1$ 时达到。因此在下面的真实网络的一般实验中，我们领 $\alpha = 1$。此外，我们也将进一步讨论在将网络划分为多层社区结构时，α 在其中所起到的作用。

在算法1中，我们初始化分解后矩阵的列数为节点数的一半，即 $P = N/2$。

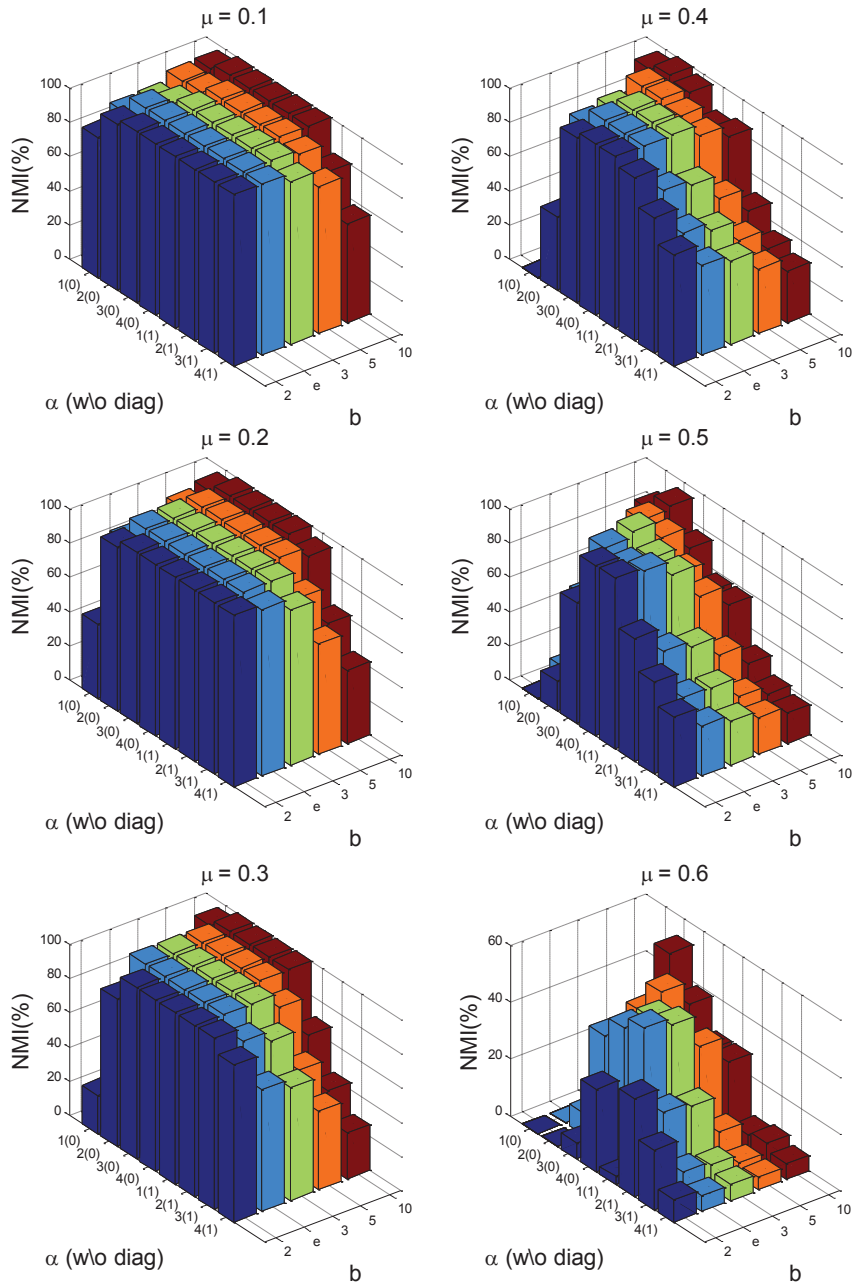


图 5.6: 在不同混合参数 μ 的 LFR 网络上，参数 α 和 b 的作用。我们给出不同的 α 和 b 的组合对基于 NMI 的社区检测结果的影响。

本节我们将说明，这个初始值对最终确定的社区个数及最终检测到的社区结构影响非常小，而且几乎是无关的。为了验证这个无关性，我们在 6 个真实网络

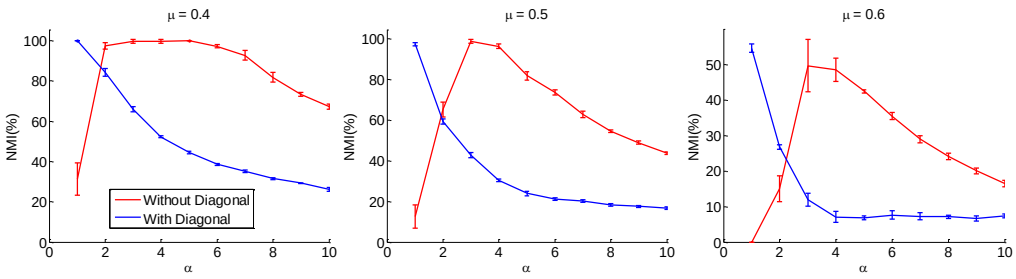


图 5.7: 在混合参数较大的 LFR 网络上（μ 从 0.4 到 0.6），对图5.6中的 3 维柱状图在 $b = e$ 时的二位横断面。

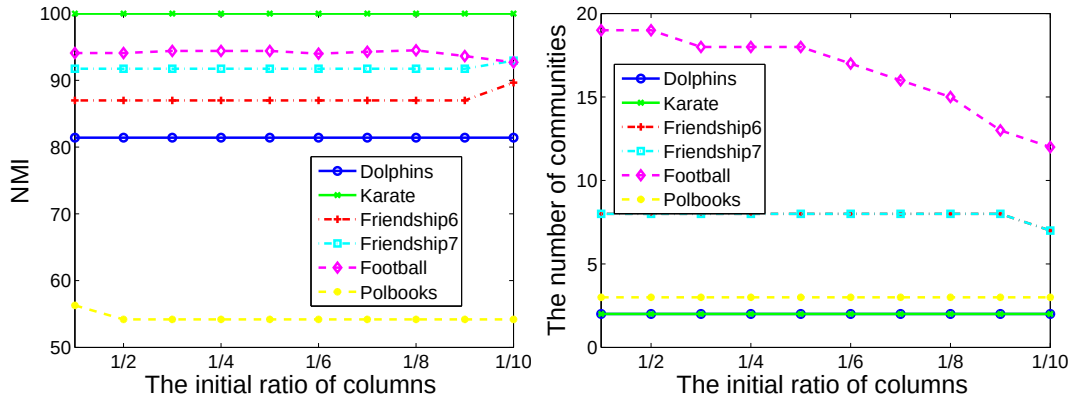


图 5.8: 初始矩阵列数 p 占比对最终结果的影响。(a) 对最终社区检测精度的影响，(b) 对最终检测到的社区个数的影响。横坐标代表初始列数的比例。

上设定初始矩阵的列数为不同的值（即从 N, $N/2$ 到 $N/10$）。结果如图 5.8所示。不同的初始列数对最终社区发现的性能的影响是非常微小的，且不同列数对除了美国高校足球俱乐部网络之外的其他网络在确定的社区个数上也几乎是没有影响的。我们下面将分析初始列数之所以对美国高校足球俱乐部网络在确定社区个数上有较大影响的原因。足球俱乐部网络本身只有 115 个节点，这些节点被分为 12 个社区。而随着初始列数比例的降低，初始矩阵的列数可能会小于实际社区的个数，即小于要被检测的社区的个数，从而对个数的影响较为剧烈。从以上的分析可以看出我们的算法对于初始的列数 P 的选择是十分鲁棒的。但是值得注意的是，如在第5.3.7节分析的，算法的实际复杂度也会受到初始矩阵列数 P 的影响。因此如果要使算法有较快的运行速度，我们应该尽量选择一个和真实社区个数处于同一数量级的初始列数。

5.4.3　算法验证

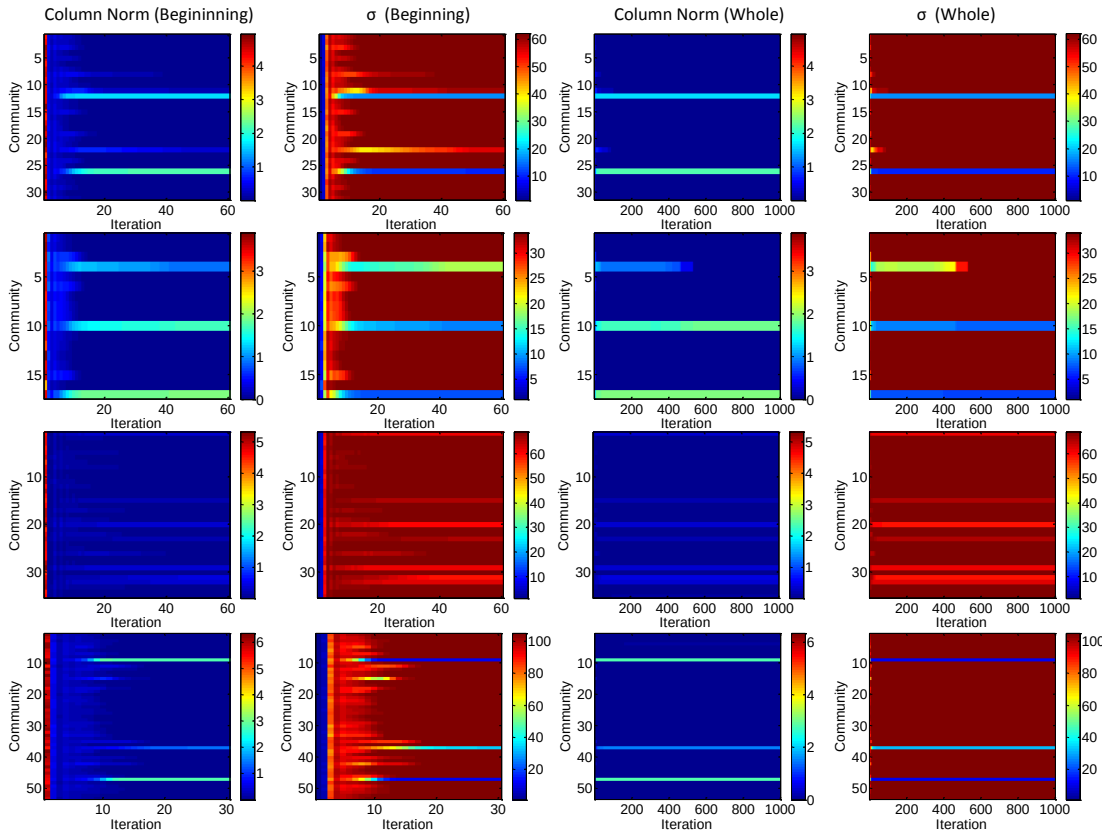


图 5.9: 随着迭代过程，分解后矩阵的列的模及相应的 σ 的变化过程。

为了得到待检测网络的社区个数，我们对分解后矩阵的不同列赋予不同的惩罚参数 σ_t，并通过迭代更新惩罚参数和相应的分解后矩阵的列，使在迭代过程中一些列坍缩到 0，而剩下的非零列数就是最终确定的社区个数，并用这些非零列来确定节点所属社区。为了验证这个过程，我们在真实网络上开展实验，观察分解后矩阵中的每列的模及相应的惩罚参数 σ_t 的变化情况。我们选择了其中的四个网络作为代表，如图5.9所示。图中每行代表一个网络，四行分别是在海豚网络、空手道网络、美国高校网络和美国政治书网络上的结果。第一、三列代表分解过程中矩阵列的模随迭代的变化情况，第二、四列是分配到每列上的惩罚参数 σ_t 随迭代的变化情况。为了能够更加直观的观察到最初的几步迭代过程中列的模和惩罚参数的变化情况，除了第三、四列给出了整体迭代全过程的变化情况外，第一、二列还给出了初始几步迭代过程中的变化情况。每个子

图中，每一行代表矩阵的一列的模或与其对应的惩罚参数在迭代过程中的变化，每一列代表某一迭代步骤中所有列或者所有惩罚参数的数值分布情况。子图中每个元素的颜色代表了他们的值的大小（第一、三列代表矩阵列的范数，第二、四列代表惩罚参数的大小），从大到小依此为红色，黄色，绿色和蓝色。

由此可以看出实验结果与我们模型的设计初衷是一致的。在迭代过程中，绝大多数的 σ 变得很大（第二、四列中的深红色），且最后只有一小部分的 σ 变得相对较小（第二、四列中的亮红色或者蓝色）。与此同时具有较大 σ 值的列的范数都被压缩到 0（第一、三列中深蓝色），这也就意味着这些列中所有元素的值都变为 0。而与具有相对较小 σ 值相对应的列的模没有坍缩到 0。这些非零列的数量就是我们检测到的社区的个数，并用他们来将节点划分到不同的社区中。以美国政治书网络为例（第三行），网络中总共有 105 个节点，我们将初始列的数目设为 53（$\lceil 105/2 \rceil$）。在第一步迭代中，我们令所有的 $\sigma = 1$，并随机初始化分解后的矩阵，并将其中每列的模归一化为 1。如图5.9中第四行所示，在三步迭代后，大多数的 σ 开始变大，相应的列开始坍缩到 0。大约 20 次迭代后，只有三个 σ 的值小于 50，而其他的都超过了 100。与此同时，与较大 σ 值对应的列中的元素都变为 0。因此我们得到最终的社区个数为 3，且非零列被用来将网络划分为三个独立的社区。在空手道网络、海豚网络和美国高校网络上我们有着类似的结果。这些结果充分验证了我们模型的正确性。

5.4.4 真实网络

这里我们使用 9 个常用的真实世界网络来验证我们的自动化社区发现算法 NMF-AWL。由于美国高校足球网络被分为相对规模较小的社区结构（平均每个社区不足 10 个节点），因此我们在这个网络上令 $\alpha = 2$，而在除此之外的所有其他网络上我们均统一的使用 $\alpha = 1$。在真实网络上，算法检测到的社区个数和社区结构的归一化互斥信息如表5.1所示。

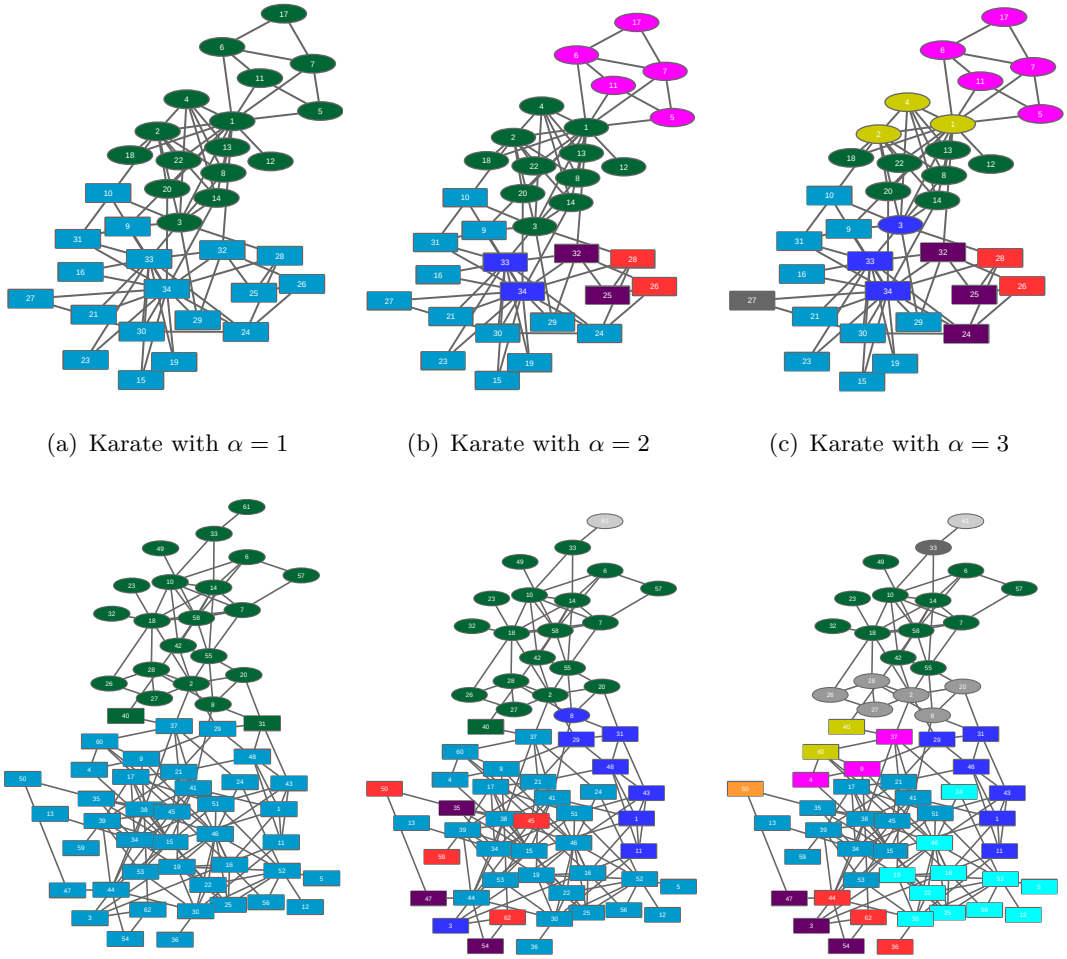
在我们验证的 9 个网路中，我们提出的 NMF-AWL 在其中的 6 个达到了最好（用黑体表示）的性能，它们分别是海豚网络、空手道网络、美国高校细分网络、美国高校足球俱乐部网络、机器学习论文引用网络和来自 Citeseer 网站的论文引用网络；在其中的两个达到次好（用下划线标注）的性能，他们分别是美国高校网络和美国政治相关博客链接网络。相应的，另一个基于非负矩阵分解的算法贝叶斯非负矩阵分解算法除了在足球网络外的性能都非常差。虽

表 5.1: 在真实网络上各种方法基于归一化互斥信息（NMI）的性能及检测到的社区个数。

Datasets	Bayesian NMF	Louvain	SP	EO	NMF-AWL
Dolphins	37.41 (16)	51.62 (5)	75.32 (2)	57.92 (4)	**81.41 (2)**
Karate	47.48 (9)	58.66 (4)	**100.00 (2)**	58.66 (4)	**100.00 (2)**
Friendship6	79.16 (12)	85.18 (7)	41.76 (2)	**95.21 (6)**	86.99 (8)
Friendship7	83.80 (12)	87.84 (7)	47.73 (2)	90.99 (6)	**91.79 (8)**
Polbooks	39.84 (15)	57.44 (3)	**59.79 (2)**	55.60 (5)	54.20 (3)
Football	93.63 (13)	89.03 (10)	33.35 (2)	88.48 (10)	**93.83 (14)**
Polblogs	23.30 (383)	**37.52 (276)**	18.76 (13)	19.00 (14)	36.29 (298)
Cora	42.01 (227)	25.96 (104)	29.49 (141)	44.07 (134)	**46.72 (163)**
Citeseer	33.19 (538)	24.38 (468)	27.70 (518)	32.75 (486)	**34.24 (654)**

然外部优化算法在五个网络中也获得了令人满意的结果，但是他们的性能都略逊于我们提出的 NMF-AWL，且在海豚网络、空手道网络和美国政治博客引用网络上的性能无法令人满意。其他的两个对比算法只能在一部分网络上得到差强人意的结果。Louvain 只能在美国高校网络、美国高校足球俱乐部网络、政治书网络和政治博客网络上得到很好的结果，而谱聚类算法只能在海豚网络、空手道网络和政治书网络上获得合理的结果。NMF-AWL 算法检测到的社区结构的可视化结果如图 5.10(a), 5.10(d) 和 5.11(a)所示。

如在5.4.2节中指出的，小的 α 会使 σ 稍大，从而分解后矩阵中有更多的的列被压缩到 0，导致检测到社区的个数相对较少；反之，较大的 α 会使检测到社区的个数相对偏多。为了从实验角度验证参数 α 在将网络划分为不同尺度社区上的性能，我们在在空手道网络、海豚网络和美国政治书网络上分别令 $\alpha = 1, 2$ 和 3。结果如图5.10和5.11中所示。容易发现，随着 α 的增大，很多大的社区被进一步划分为较小的子社区，从而实现分层社区的检测。例如在空手道网络中，当将 α 从 1 变为 2 时，图5.10(a)中的绿色社区被分为图5.10(b)中的两个子社区，即绿色的子社区和紫色的子社区。当将 α 增加到 3 时，这个全新的绿色子社区又被进一步划分为图5.10(c)中两个更小的的黄色子社区和绿色子社区。其他的两个网络上也有着类似的结果。因此这就从实验角度验证了 α 可以被作为



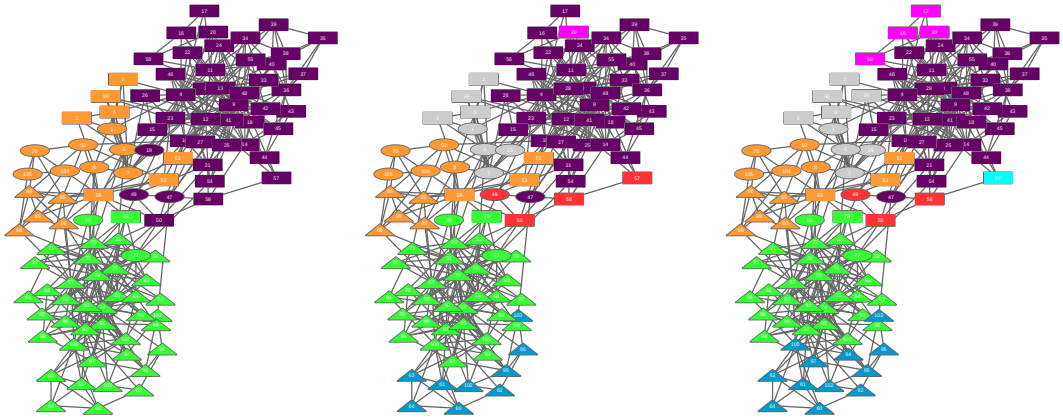
(a) Karate with $\alpha = 1$ (b) Karate with $\alpha = 2$ (c) Karate with $\alpha = 3$

(d) Dolphins with $\alpha = 1$ (e) Dolphins with $\alpha = 2$ (f) Dolphins with $\alpha = 3$

图 5.10: 参数 α 在真实网络分层社区划分上的作用。

一个控制社区尺度的参数用于多层社区检测。

更进一步，我们在单词联系网络 [64] 上比较这些方法的优劣。在这个网络中，每个节点代表一个单词，每条边代表相应的两个单词之间存在联合的关系。我们采用节点对相似性的丰富程度来评价社区检测的质量。节点对相似性的丰富程度是被分到同一个社区中的任意两个节点的平均元数据相似性除以网络中所有节点对的元数据平均相似性。从 NMF-AWL 获得的丰富程度是 2.53，高于贝叶斯非负矩阵分解的 2.47 和谱聚类的 1.2。这些结果都说明我们提出的 NMF-AWL 在大规模网络上的性能超越了其他方法。.



(a) Polbooks with $\alpha = 1$　(b) Polbooks with $\alpha = 2$　(c) Polbooks with $\alpha = 3$

图 5.11: 参数 α 在美国政治书网络分层社区划分上的作用。

5.5　小结

我们基于非负矩阵分解提出了一个全新的自动社区发现算法–自动加权低秩约束的非负矩阵分解 NMF-AWL。如下的两点使得新的方法不仅有效而且高效。一个是用来将给定网络分解为社区的加权组稀疏低秩正则项。另一个是可以从数据中通过合并小社区来自动确定正确成分个数的自适应优化算法。据我们所知，这是第一个不需要提前设定分解后矩阵的秩就能直接得到潜在的低秩表达的基于非负矩阵分解的方法。与之前通过一致加权低秩正则项来约束分解后矩阵的低秩矩阵近似方法相比，这是第一个在非负矩阵分解中引入自适应权重低秩约束正则来进行自动社区检测的算法。在人工网络和真实网络上的大量实验说明和其他四个最先进方法相比，NMF-AWL 具有优越的性能。更为重要的是，NMF-AWL 可以被进一步扩展到其他的基于非负矩阵分解的算法和应用中，如基于矩阵分解的分类和聚类算法。

第六章　总结与展望

随着复杂网络规模的增大，网络的稀疏性和噪音性就显得尤为突出，这就导致社区结构的可检测性的明显降低。为了大幅提升社区发现性能，尤其是在网络结构相对模糊的网络中，融入除拓扑结构信息外的其他附加信息是相对有效的方式。而如何融入各种形式的监督信息不仅是其中最为基本、最为重要的问题，同时也为其他信息的融入奠定坚实的理论基础。

6.1　总结

由于成对监督信息的易获取性且其可以统一的表示绝大多数其他类型的监督信息，我们在论文中对其进行了深入的讨论。为了大幅提升成对监督信息的对社区发现性能的促进作用并使其有为了广泛的实用场景，我们分别从高效的成对监督信息的融入算法、高效的成对监督信息选择算法和自动化社区发现算法三个角度开展深入的研究，主要的研究成果如下：

1. **统一的半监督社区发现框架**。我们首先对一组现有的社区发现算法给出了一种统一的解释, 即基于隐空间节点表达的聚类算法。之后基于隐空间相似性, 我们提出了一种统一的半监督社区发现框架。这种算法可以通过隐空间的图正则将成对先验信息与网络拓扑结构信息相融合, 从而提升传统单纯基于网络拓扑结构的社区发现算法的性能。提出的半监督社区发现框架可以应用于绝大多数符合我们之前统一解释的基于矩阵的社区发现算法, 如非负矩阵分解算法、谱聚类算法及他们的变种算法。与之前通过直接修改邻接矩阵将先验信息转化为拓扑结构信息的半监督社区发现算法不同，我们首次将半监督社区发现算法公式化一个整体的基于隐空间图正则的优化问题并可以权衡来自拓扑结构信息和先验信息对最终性能的贡献。

2. **主动的半监督社区发现算法**。我们为高效的半监督社区发现提出了一个主动的链接选择框架。这个框架由链接策略和断链策略两部分组成。通过使用链接策略, 我们根据非负矩阵分解的结果交互的选择最不确定的边供人

工标注，并将对应的节点与社区核心节点相连。为了进一步挖掘加入的成对监督信息, 我们设计的断链策略在断开可能是社区之间的边。这两个策略联合的高效锐化网络邻接矩阵的块结构。之后基于标注的链接, 我们更新网络拓扑结构并运行下一轮的非负矩阵分解。和传统的通过加入随机选择的监督信息的半监督社区发现方法相比, 我们提出的框架主动的选择罪不确定的链接供标注。这显著的减少了对监督信息的需求并大幅地提升了半监督社区发现的效率。此外, 为了加速框架迭代过程中的非负矩阵分解，我们提出了一种可以提速十倍以上的非负矩阵分解初始化方法。

3. **自动化社区发现算法**。我们基于非负矩阵分解提出了一个全新的自动社区发现算法 NMF-AWL。我们首先证明了通常的低秩约束相当于使分解后矩阵的所有列以相等程度被压缩到零。因此我们先令分解后的矩阵列数远远大于可能的社区个数，并通过自动适应权重的低秩约束给每个分解后矩阵的列加入不同的压缩权重，从而使其中的一些列在矩阵分解的过程中被压缩到零来自动确定社区的个数，且最终得到的非零列可以用来检测社区结构。据我们所知，这是第一不需要提前设定分解后矩阵的秩就能直接得到潜在的低秩表达的基于非负矩阵分解的方法。与之前通过一致加权低秩正则来约束分解后矩阵的低秩矩阵近似方法相比，这是第一个在非负矩阵分解中引入自适应权重低秩约束正则来进行社区检测的算法。

6.2 展望

虽然我们对基于成对监督信息的半监督社区发现在高效性和自动性等方面进行了深入的研究，但是半监督社区发现领域还有着很多需要解决的问题。在今后的工作中我们将主要从下面的四个方面着手继续对半监督社区发现问题开展进一步的研究

1. **融入多类异构监督信息的半监督社区发现框架**。现在的绝大多数半监督算法都是单一的考虑单独节点监督信息或者成对监督信息，而能够将它们用一个统一的半监督社区发现框架进行融合并充分发挥两者的特点并的工作还很少。但在现实应用中，通常可以同时获得这两类信息，如何高效的同时嵌入这两类信息并设计相应的主动学习方法是我们今后研究工作的重点。这也会为除监督信息之外的其他多类异构信息的融合奠定基础。

2. **适用于重叠社区、层次社区的半监督社区发现框架**。现有的半监督社区发现算法大都着眼于将网络划分为单一尺度的独立社区。而在很多的真实网络中的社区结构是更加复杂的，通常都存在重叠性和分层的特点。而将现有的方法直接移植到这类问题上通常并不是十分有效的。因此我们也将研究如何设计适用于更复杂社区结构半监督算法。

3. **监督信息在提升算法速度方面的作用**。目前为止，监督信息的引入都是为了提升社区发现的精度，而很少有研究考虑如何通过嵌入监督信息来提升社区发现算法速度的研究。但当监督信息加入后，由于网络邻接矩阵的块结构变得更加清晰了，这就促使社区发现算法能够更加快速的检测网络结构。因此如何通过有效的嵌入监督信息来提升算法速度也将是我们今后的一个重要研究方向。

4. **基于深度学习的社区发现算法**。随着深度学习在包括自然语言处理和计算机视觉等领域取得了大量振奋人心的结果。但如何将其应用到社区发现中依旧是一个未知的问题。因此，我们也将在这个领域开展研究工作，并考虑如何在其中嵌入半监督信息。

参考文献

[1] Lada A Adamic and Natalie Glance. The political blogosphere and the 2004 us election: divided they blog. In *Proceedings of the 3rd international workshop on Link discovery*, pages 36–43. ACM, 2005.

[2] Y.-Y. Ahn, J. P. Bagrow, and S. Lehmann. Link communities reveal multiscale complexity in networks. *Nature*, 466:761–764, 2010.

[3] Edoardo M. Airoldi, David M. Blei, Stephen E. Fienberg, and Eric P. Xing. Mixed membership stochastic blockmodels. *J. Mach. Learn. Res.*, 9:1981–2014, June 2008.

[4] Armen E Allahverdyan, Greg Ver Steeg, and Aram Galstyan. Community detection with and without prior information. *EPL (Europhysics Letters)*, 90(1):18002, 2010.

[5] Michael J Barber. Modularity and community detection in bipartite networks. *Physical Review E*, 76(6):066102, 2007.

[6] Danielle S. Bassett, Mason A. Porter, Nicholas F. Wymbs, Scott T. Grafton, Jean M. Carlson, and Peter J. Mucha. Robust detection of dynamic community structure in networks. *Chaos: An Interdisciplinary Journal of Nonlinear Science*, 23(1):–, 2013.

[7] Mikhail Belkin, Partha Niyogi, and Vikas Sindhwani. Manifold regularization: A geometric framework for learning from labeled and unlabeled examples. *The Journal of Machine Learning Research*, 7:2399–2434, 2006.

[8] Jonathan W Berry, Bruce Hendrickson, Randall A LaViolette, and Cynthia A Phillips. Tolerating the community detection resolution limit with edge weighting. *Physical Review E*, 83(5):056119, 2011.

[9] David M. Blei, Perry R. Cook, and Matthew Hoffman. Bayesian nonparametric matrix factorization for recorded music. In Johannes Fürnkranz and Thorsten Joachims, editors, *Proceedings of the 27th International Conference on Machine Learning (ICML-10)*, pages 439–446. Omnipress, 2010.

[10] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. Fast unfolding of communities in large networks. *Journal of Statistical Mechanics: Theory and Experiment*, 2008(10):P10008, 2008.

[11] Thang Nguyen Bui and Curt Jones. Finding good approximate vertex and edge partitions is np-hard. *Information Processing Letters*, 42(3):153–159, 1992.

[12] Deng Cai, Xiaofei He, Jiawei Han, and Thomas S Huang. Graph regularized nonnegative matrix factorization for data representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 33(8):1548–1560, 2011.

[13] Emmanuel J. Candès, Xiaodong Li, Yi Ma, and John Wright. Robust principal component analysis? *J. ACM*, 58(3):11:1–11:37, June 2011.

[14] Xiaochun Cao, Liang Yang, and Xiaojie Guo. Total variation regularized rpca for irregularly moving object detection under dynamic background. 2015.

[15] Ali Taylan Cemgil. Bayesian inference for nonnegative matrix factorisation models. *Intell. Neuroscience*, 2009:4:1–4:17, January 2009.

[16] Yanhua Chen, Manjeet Rege, Ming Dong, and Jing Hua. Incorporating user provided constraints into document clustering. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 103–112. IEEE, 2007.

[17] Yanhua Chen, Manjeet Rege, Ming Dong, and Jing Hua. Non-negative matrix factorization for semi-supervised data clustering. *Knowledge and Information Systems*, 17(3):355–379, 2008.

[18] Yudong Chen, Sujay Sanghavi, and Huan Xu. Clustering sparse graphs. In *Advances in neural information processing systems*, pages 2204–2212, 2012.

[19] Jianjun Cheng, Mingwei Leng, Longjie Li, Hanhai Zhou, and Xiaoyun Chen. Active semi-supervised community detection based on must-link and cannot-link constraints. *PLoS ONE*, 9(10):e110088, 10 2014.

[20] Don Coppersmith and Shmuel Winograd. Matrix multiplication via arithmetic progressions. *Journal of symbolic computation*, 9(3):251–280, 1990.

[21] Aurelien Decelle, Florent Krzakala, Cristopher Moore, and Lenka Zdeborová. Inference and phase transitions in the detection of modules in sparse networks. *Phys. Rev. Lett.*, 107:065701, Aug 2011.

[22] James W Demmel, Michael T Heath, and Henk A Van Der Vorst. Parallel numerical linear algebra. *Acta numerica*, 2:111–197, 1993.

[23] Chris HQ Ding, Xiaofeng He, and Horst D Simon. On the equivalence of nonnegative matrix factorization and spectral clustering. In *SDM*, volume 5, pages 606–610. SIAM, 2005.

[24] Jordi Duch and Alex Arenas. Community detection in complex networks using extremal optimization. *Phys. Rev. E*, 72:027104, Aug 2005.

[25] Eric Eaton and Rachael Mansbach. A spin-glass model for semi-supervised community detection. 2012.

[26] Santo Fortunato. Community detection in graphs. *Physics Reports*, 486(3):75–174, 2010.

[27] Santo Fortunato and Marc Barthelemy. Resolution limit in community detection. *Proceedings of the National Academy of Sciences*, 104(1):36–41, 2007.

[28] Michelle Girvan and Mark EJ Newman. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821–7826, 2002.

[29] Benjamin H Good, Yves-Alexandre de Montjoye, and Aaron Clauset. Performance of modularity maximization in practical contexts. *Physical Review E*, 81(4):046106, 2010.

[30] Magnus Rudolph Hestenes and Eduard Stiefel. *Methods of conjugate gradients for solving linear systems*, volume 49. NBS, 1952.

[31] Brian Karrer and M. E. J. Newman. Stochastic blockmodels and community structure in networks. *Phys. Rev. E*, 83:016107, Jan 2011.

[32] Andrea Lancichinetti and Santo Fortunato. Benchmarks for testing community detection algorithms on directed and weighted graphs with overlapping communities. *Physical Review E*, 80(1):016118, 2009.

[33] Andrea Lancichinetti and Santo Fortunato. Community detection algorithms: a comparative analysis. *Physical review E*, 80(5):056117, 2009.

[34] Andrea Lancichinetti and Santo Fortunato. Limits of modularity maximization in community detection. *Physical review E*, 84(6):066122, 2011.

[35] Andrea Lancichinetti, Santo Fortunato, and János Kertész. Detecting the overlapping and hierarchical community structure in complex networks. *New Journal of Physics*, 11(3):033015, 2009.

[36] Andrea Lancichinetti, Santo Fortunato, and Filippo Radicchi. Benchmark graphs for testing community detection algorithms. *Physical Review E*, 78(4):046110, 2008.

[37] Daniel D Lee and H Sebastian Seung. Learning the parts of objects by non-negative matrix factorization. *Nature*, 401(6755):788–791, 1999.

[38] Daniel D Lee and H Sebastian Seung. Algorithms for non-negative matrix factorization. In *NIPS*, pages 556–562, 2000.

[39] Mingwei Leng, Yukai Yao, Jianjun Cheng, Weiming Lv, and Xiaoyun Chen. Active semi-supervised community detection algorithm with label propa-

gation. In *Database Systems for Advanced Applications*, pages 324–338. Springer, 2013.

[40] Jure Leskovec, Kevin J Lang, Anirban Dasgupta, and Michael W Mahoney. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1):29–123, 2009.

[41] Jure Leskovec, Kevin J Lang, and Michael Mahoney. Empirical comparison of algorithms for network community detection. In *Proceedings of the 19th international conference on World wide web*, pages 631–640. ACM, 2010.

[42] Lei Li, Mei Du, Guanfeng Liu, Xuegang Hu, and Gongqing Wu. Extremal optimization-based semi-supervised algorithm with conflict pairwise constraints for community detection. In *Advances in Social Networks Analysis and Mining (ASONAM), 2014 IEEE/ACM International Conference on*, pages 180–187, Aug 2014.

[43] Tao Li, Chris Ding, and Michael I Jordan. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. In *Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on*, pages 577–582. IEEE, 2007.

[44] Zhenping Li, Shihua Zhang, Rui-Sheng Wang, Xiang-Sun Zhang, and Luonan Chen. Quantitative function for community detection. *Physical review E*, 77(3):036109, 2008.

[45] Xiao Liang, Xiang Ren, Zhengdong Zhang, and Yi Ma. Repairing sparse low-rank texture. In *Computer Vision–ECCV 2012*, pages 482–495. Springer, 2012.

[46] Chao Liu, Hung-chih Yang, Jinliang Fan, Li-Wei He, and Yi-Min Wang. Distributed nonnegative matrix factorization for web-scale dyadic data analysis on mapreduce. In *Proceedings of the 19th International Conference on World Wide Web*, pages 681–690, New York, NY, USA, 2010. ACM.

[47] Dong Liu, Hong-Yu Bai, Hui-Jia Li, and Wen-Jun Wang. Semi-supervised community detection using label propagation. *International Journal of Modern Physics B*, 28(29):1450208, 2014.

[48] Dong Liu, Dequan Duan, Shikai Sui, and Guojie Song. Effective semisupervised community detection using negative information. *Mathematical Problems in Engineering*, 2015, 2015.

[49] Dong Liu, Xiao Liu, Wenjun Wang, and Hongyu Bai. Semi-supervised community detection based on discrete potential theory. *Physica A: Statistical Mechanics and its Applications*, 416:173–182, 2014.

[50] Guangcan Liu, Zhouchen Lin, Shuicheng Yan, Ju Sun, Yong Yu, and Yi Ma. Robust recovery of subspace structures by low-rank representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(1):171–184, 2013.

[51] Guangcan Liu, Zhouchen Lin, and Yong Yu. Robust subspace segmentation by low-rank representation. In *ICML*, pages 663–670, 2010.

[52] Haifeng Liu and Zhaohui Wu. Non-negative matrix factorization with constraints. In *Twenty-Fourth AAAI Conference on Artificial Intelligence*, 2010.

[53] Hongchao Lu, Xiaopeng Zhu, Haifeng Liu, Geir Skogerbø, Jingfen Zhang, Yong Zhang, Lun Cai, Yi Zhao, Shiwei Sun, Jingyi Xu, et al. The interactome as a tree—an attempt to visualize the protein–protein interaction network in yeast. *Nucleic acids research*, 32(16):4804–4811, 2004.

[54] David Lusseau and Mark EJ Newman. Identifying the role that animals play in their social networks. *Proceedings of the Royal Society of London. Series B: Biological Sciences*, 271(Suppl 6):S477–S481, 2004.

[55] Xiaoke Ma, Lin Gao, Xuerong Yong, and Lidong Fu. Semi-supervised clustering algorithm for community structure detection in complex networks.

Physica A: Statistical Mechanics and its Applications, 389(1):187 – 197, 2010.

[56] Fragkiskos D Malliaros and Michalis Vazirgiannis. Clustering and community detection in directed networks: A survey. *Physics Reports*, 533(4):95–142, 2013.

[57] Rahul Mazumder, Trevor Hastie, and Robert Tibshirani. Spectral regularization algorithms for learning large incomplete matrices. *J. Mach. Learn. Res.*, 11:2287–2322, August 2010.

[58] Stefano Monti, Pablo Tamayo, Jill Mesirov, and Todd Golub. Consensus clustering: a resampling-based method for class discovery and visualization of gene expression microarray data. *Machine learning*, 52(1-2):91–118, 2003.

[59] Cristopher Moore, Xiaoran Yan, Yaojia Zhu, Jean-Baptiste Rouquier, and Terran Lane. Active learning for node classification in assortative and disassortative networks. In *Proceedings of the 17th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '11, pages 841–849, New York, NY, USA, 2011. ACM.

[60] M. Morup and L.K. Hansen. Tuning pruning in sparse non-negative matrix factorization. In *Signal Processing Conference, 2009 17th European*, pages 1923–1927, Aug 2009.

[61] Morten Morup and Lars Kai Hansen. Automatic relevance determination for multi-way models. *Journal of Chemometrics*, 23(7-8):352–363, 2009.

[62] Raj Rao Nadakuditi and M. E. J. Newman. Graph spectra and the detectability of community structure in networks. *Phys. Rev. Lett.*, 108:188701, May 2012.

[63] Keisuke Nakata and Tsuyoshi Murata. *Complex Networks VI: Proceedings of the 6th Workshop on Complex Networks CompleNet 2015*, chapter Fast

Optimization of Hamiltonian for Constrained Community Detection, pages 79–89. Springer International Publishing, Cham, 2015.

[64] D. L. Nelson, McEvoy, C. L., and T. A. Schreiber. The University of South Florida word association, rhyme, and word fragment norms, 1998.

[65] M. E. J. Newman. Spectral methods for community detection and graph partitioning. *Phys. Rev. E*, 88:042822, Oct 2013.

[66] Mark EJ Newman. Detecting community structure in networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 38(2):321–330, 2004.

[67] Mark EJ Newman. Finding community structure in networks using the eigenvectors of matrices. *Phys. Rev. E*, 74(3):036104, 2006.

[68] Mark EJ Newman. Modularity and community structure in networks. *Proceedings of the National Academy of Sciences*, 103(23):8577–8582, 2006.

[69] MEJ Newman. Spectral methods for network community detection and graph partitioning. *arXiv preprint arXiv:1307.7729*, 2013.

[70] Gergely Palla, Imre Derényi, Illés Farkas, and Tamás Vicsek. Uncovering the overlapping community structure of complex networks in nature and society. *Nature*, 435(7043):814–818, 2005.

[71] Pascal Pons and Matthieu Latapy. Post-processing hierarchical community structures: Quality improvements and multi-scale view. *Theoretical Computer Science*, 412(8):892–900, 2011.

[72] Ioannis Psorakis, Stephen Roberts, Mark Ebden, and Ben Sheldon. Overlapping community detection using bayesian non-negative matrix factorization. *Physical Review E*, 83(6):066114, 2011.

[73] Guo-Jun Qi, Charu C Aggarwal, and Thomas Huang. Community detection with edge content in social media networks. In *Data Engineer-*

ing (ICDE), 2012 IEEE 28th International Conference on, pages 534–545. IEEE, 2012.

[74] Filippo Radicchi. Detectability of communities in heterogeneous networks. *Phys. Rev. E*, 88:010801, Jul 2013.

[75] Filippo Radicchi. Driving interconnected networks to supercriticality. *Phys. Rev. X*, 4:021014, Apr 2014.

[76] Filippo Radicchi. A paradox in community detection. *EPL (Europhysics Letters)*, 106(3):38001, 2014.

[77] Jörg Reichardt and Stefan Bornholdt. Statistical mechanics of community detection. *Physical Review E*, 74(1):016110, 2006.

[78] Yiye Ruan, David Fuhry, and Srinivasan Parthasarathy. Efficient community detection in large networks using content and links. In *Proceedings of the 22nd international conference on World Wide Web*, pages 1089–1098. International World Wide Web Conferences Steering Committee, 2013.

[79] M. N. Schmidt and M. Mørup. Infinite non-negative matrix factorization. In *European Signal Processing Conference (EUSIPCO)*, Richard Petersens Plads, Building 321, DK-2800 Kgs. Lyngby, aug 2010. Informatics and Mathematical Modelling, Technical University of Denmark, DTU.

[80] Hua-Wei Shen and Xue-Qi Cheng. Spectral methods for the detection of network community structure: a comparative analysis. *Journal of Statistical Mechanics: Theory and Experiment*, 2010(10):P10020, 2010.

[81] Hua-Wei Shen, Xue-Qi Cheng, and Bin-Xing Fang. Covariance, correlation matrix, and the multiscale community structure of networks. *Physical Review E*, 82(1):016114, 2010.

[82] Huawei Shen, Xueqi Cheng, Kai Cai, and Mao-Bin Hu. Detect overlapping and hierarchical community structure in networks. *Physica A: Statistical Mechanics and its Applications*, 388(8):1706–1712, 2009.

[83] Lixin Shi, Yuhang Zhao, and Jie Tang. Batch mode active learning for networked data. *ACM Trans. Intell. Syst. Technol.*, 3(2):33:1–33:25, February 2012.

[84] Xiaohua Shi, Hongtao Lu, Yangchen He, and Shan He. Community detection in social network with pairwisely constrained symmetric non-negative matrix factorization. In *Proceedings of the 2015 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining 2015*, ASONAM '15, pages 541–546, New York, NY, USA, 2015. ACM.

[85] Paris Smaragdis and Judith C Brown. Non-negative matrix factorization for polyphonic music transcription. In *Applications of Signal Processing to Audio and Acoustics, 2003 IEEE Workshop on.*, pages 177–180. IEEE, 2003.

[86] Alexander Strehl and Joydeep Ghosh. Cluster ensembles—a knowledge reuse framework for combining multiple partitions. *The Journal of Machine Learning Research*, 3:583–617, 2003.

[87] V. Y. F. Tan and C. Fevotte. Automatic relevance determination in non-negative matrix factorization with the /spl beta/-divergence. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 35(7):1592–1605, 2013.

[88] Wei Tang, Zhengdong Lu, and Inderjit S Dhillon. Clustering with multiple graphs. In *Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on*, pages 1016–1021. IEEE, 2009.

[89] Amanda L Traud, Peter J Mucha, and Mason A Porter. Social structure of facebook networks. *Physica A: Statistical Mechanics and its Applications*, 391(16):4165–4180, 2012.

[90] Stijn Marinus van Dongen. *Graph clustering by flow simulation.* PhD thesis, Dutch National Research Institute for Mathematics and Computer Science, University of Utrecht, Netherlands, 2000.

[91] Greg Ver Steeg, Aram Galstyan, and Armen E Allahverdyan. Statistical mechanics of semi-supervised clustering in sparse graphs. *Journal of Statistical Mechanics: Theory and Experiment*, 2011(08):P08009, 2011.

[92] Dingding Wang, Tao Li, Shenghuo Zhu, and Chris Ding. Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In *Proceedings of the 31st annual international ACM SIGIR*, pages 307–314. ACM, 2008.

[93] Fei Wang, Tao Li, Xin Wang, Shenghuo Zhu, and Chris Ding. Community discovery using nonnegative matrix factorization. *Data Min. Knowl. Discov.*, 22(3):493–521, May 2011.

[94] Fei Wang, Tao Li, and Changshui Zhang. Semi-supervised clustering via matrix factorization. In *SDM*, pages 1–12. SIAM, 2008.

[95] Fei Wang, Hanghang Tong, and Ching-Yung Lin. Towards evolutionary nonnegative matrix factorization. In *AAAI*, volume 11, pages 501–506, 2011.

[96] Rui-Sheng Wang, Shihua Zhang, Yong Wang, Xiang-Sun Zhang, and Luonan Chen. Clustering complex networks and biological networks by nonnegative matrix factorization with various similarity measures. *Neurocomputing*, 72(1):134–141, 2008.

[97] Yu-Xiong Wang and Yu-Jin Zhang. Nonnegative matrix factorization: A comprehensive review. *Knowledge and Data Engineering, IEEE Transactions on*, 25(6):1336–1353, June 2013.

[98] Zhaoxian Wang, Wenjun Wang, Guixiang Xue, Pengfei Jiao, and Xuewei Li. *Advances in Swarm and Computational Intelligence: 6th International Conference, ICSI 2015 held in conjunction with the Second BRICS Congress, CCI 2015, Beijing, June 25-28, 2015, Proceedings, Part II*, chapter Semi-supervised Community Detection Framework Based on Nonnegative Factorization Using Individual Labels, pages 349–359. Springer International Publishing, Cham, 2015.

[99] Stanley Wasserman. *Social network analysis: Methods and applications*, volume 8. Cambridge university press, 1994.

[100] Jierui Xie, Stephen Kelley, and Boleslaw K Szymanski. Overlapping community detection in networks: The state-of-the-art and comparative study. *ACM Computing Surveys (CSUR)*, 45(4):43, 2013.

[101] Jierui Xie, Stephen Kelley, and Boleslaw K Szymanski. Overlapping community detection in networks: The state-of-the-art and comparative study. *Acm computing surveys (csur)*, 45(4):43, 2013.

[102] Wei Xu, Xin Liu, and Yihong Gong. Document clustering based on non-negative matrix factorization. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pages 267–273. ACM, 2003.

[103] Jaewon Yang and Jure Leskovec. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, pages 587–596. ACM, 2013.

[104] Jaewon Yang and Jure Leskovec. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems*, 42(1):181–213, 2015.

[105] Jaewon Yang, Julian McAuley, and Jure Leskovec. Community detection in networks with node attributes. *arXiv preprint arXiv:1401.7267*, 2014.

[106] Tianbao Yang, Rong Jin, Yun Chi, and Shenghuo Zhu. A bayesian framework for community detection integrating content and link. In *Proceedings of the Twenty-Fifth Conference on Uncertainty in Artificial Intelligence*, pages 615–622. AUAI Press, 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 ACM SIGKDD '09*, pages 927–936, 2009.

[108] Zhirong Yang, Zhanxing Zhu, and Erkki Oja. Automatic rank determination in projective nonnegative matrix factorization. 6365:514–521, 2010.

[109] W Zachary. An information flow modelfor conflict and fission in small groups1. *Journal of anthropological research*, 33(4):452–473, 1977.

[110] Pan Zhang, Cristopher Moore, and Lenka Zdeborová. Phase transitions in semisupervised clustering of sparse networks. *Physical Review E*, 90(5):052802, 2014.

[111] Shihua Zhang, Rui-Sheng Wang, and Xiang-Sun Zhang. Uncovering fuzzy community structure in complex networks. *Physical Review E*, 76(4):046103, 2007.

[112] Yu Zhang and Dit-Yan Yeung. Overlapping community detection via bounded nonnegative matrix tri-factorization. In *Proceedings of the 18th ACM SIGKDD '12*, pages 606–614, 2012.

[113] Zhengdong Zhang, Xiao Liang, Arvind Ganesh, and Yi Ma. Tilt: transform invariant low-rank textures. In *Computer Vision–ACCV 2010*, pages 314–328. Springer, 2010.

[114] Zhong-Yuan Zhang. Community structure detection in complex networks with partial background information. *EPL (Europhysics Letters)*, 101(4):48005, 2013.

[115] Zhong-Yuan Zhang, Kai-Di Sun, and Si-Qi Wang. Enhanced community structure detection in complex networks with partial background information. *Scientific reports*, 3, 2013.

[116] Zhong-Yuan Zhang, Yong Wang, and Yong-Yeol Ahn. Overlapping community detection in complex networks using symmetric binary matrix factorization. *Phys. Rev. E*, 87:062803, Jun 2013.

[117] Ruicong Zhi, Markus Flierl, Qiuqi Ruan, and W Bastiaan Kleijn. Graph-preserving sparse nonnegative matrix factorization with application to fa-

cial expression recognition. *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, 41(1):38–52, 2011.

[118] Mingjun Zhong and Mark Girolami. Reversible jump mcmc for non-negative matrix factorization. In *International Conference on Artificial Intelligence and Statistics*, pages 663–670, 2009.

[119] Mingjun Zhong and Mark Girolami. Reversible jump mcmc for non-negative matrix factorization. In David V. Dyk and Max Welling, editors, *Proceedings of the Twelfth International Conference on Artificial Intelligence and Statistics (AISTATS-09)*, volume 5, pages 663–670. Journal of Machine Learning Research - Proceedings Track, 2009.

[120] Xiaowei Zhou, Can Yang, and Weichuan Yu. Automatic mitral leaflet tracking in echocardiography by outlier detection in the low-rank representation. In *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pages 972–979. IEEE, 2012.

[121] Xiaowei Zhou, Can Yang, and Weichuan Yu. Moving object detection by detecting contiguous outliers in the low-rank representation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 35(3):597–610, 2013.

[122] Xiaojin Zhu, Zoubin Ghahramani, John Lafferty, et al. Semi-supervised learning using gaussian fields and harmonic functions. In *ICML*, volume 3, pages 912–919, 2003.

[123] Yan Zhu, Liping Jing, and Jian Yu. Text clustering via constrained non-negative matrix factorization. In *Data Mining (ICDM), 2011 IEEE 11th International Conference on*, pages 1278–1283. IEEE, 2011.

发表文章目录

[1] **Liang Yang**, Dongxiao He, Xiao Wang, Xiaochun Cao, Weixiong Zhang. ”Modularity based Community Detection with Deep Learning”. Accepted in International Joint Conference on Artificial Intelligence 2016. (CCF A 类)

[2] **Liang Yang**, Xiaochun Cao, Di Jin, Xiao Wang, Dan Meng. ”A Unified Semi-Supervised Community Detection Framework Using Latent Space Graph Regularization”. IEEE Transactions on Cybernetics, 2015, 45(11): 2585-2598. (CCF B 类，JCR 2 区)

[3] **Liang Yang**, Di Jin, Xiao Wang, Xiaochun Cao. ”Active Link Selection for Efficient Semi-supervised Community Detection”. Scientific Reports, 2015, 5, 9039. (JCR 2 区)

[4] Xiaochun Cao, **Liang Yang**, Xiaojie Guo. ”Total Variation Regularized RPCA for Irregularly Moving Object Detection under Dynamic Background”. IEEE Transactions on Cybernetics, 2016, 46(4): 1014-1027. (CCF B 类，JCR 2 区)

[5] Menghao Li, **Liang Yang**, Zimu Yuan, Rui Zhang, Rui Xue. ”An Approach for Mitigating Potential Threats in Practical SSO Systems”. In Proceedings of Inscrypt 2015.

[6] Xiaojie Guo, Xinggang Wang, **Liang Yang**, Xiaochun Cao, Yi Ma. ”Robust Foreground Detection Using Smoothness and Arbitrariness Constraints”. In Computer Vision–ECCV 2014, pp. 535-550. Springer International Publishing, 2014. (CCF B 类)

[7] Chuan Wang, Hua Zhang, **Liang Yang**, Si Liu, Xiaochun Cao. ”Deep People Counting in Extremely Dense Crowds”. In Proceedings of the 23rd Annual ACM Conference on Multimedia Conference, pp. 1299-1302. ACM, 2015. (CCF A 类)

[8] Xiao Wang, Di Jin, Xiaochun Cao, **Liang Yang**, Weixiong Zhang. ”Semantic community identification in large attribute networks”. In Proceedings of AAAI Conference on Artificial Intelligence, pp. 130-136. 2015. (CCF A 类)

[9] **Liang Yang**, Di Jin, Xiao Wang, Xiaochun Cao, Weixiong Zhang. ”Autonomous Community Detection via Adaptively Weighted Low-rank Approximation in Nonnegative Matrix Factorization”. Submitted to IEEE Transactions on Cybernetics.

[10] **Liang Yang**, Dongxiao He, Xiao Wang, Rui Wang, Xiaochun Cao, Weixiong Zhang. ”A Unified Semi- and Un-supervised Community Detection Algorithm by Joint Topology Refinement and Label Propagation”. Submitted to Scientific Reports.

[11] Xiao Wang, Di Jin, Xiaochun Cao, **Liang Yang**, Weixiong Zhang. ”Non-topological information improves community detection via semi-supervised and active learning”. Submitted to IEEE Transactions on Cybernetics.

简　　历

基本情况

杨亮，男，天津市人，1981 年 12 月出生，未婚，中国科学院信息工程研究所在读博士研究生

教育状况

2001 年 9 月 – 2004 年 7 月：南开大学 数学科学学院，理学硕士，信息与计算科学专业

2004 年 9 月 – 2007 年 7 月：南开大学 数学科学学院，理学硕士，计算数学专业，并行计算方向，导师：邓越凡

2013 年 9 月 – 至今：中国科学院信息工程研究所信息安全国家重点实验室，工学博士，计算机软件与理论，社区发现方向，导师：操晓春

工作经历

2007 年 7 月 – 2009 年 5 月：深圳市腾讯计算机系统有限公司，数据平台部，研发工程师

2009 年 9 月 – 2010 年 5 月：北京百度网讯科技有限公司，商务搜索部，研发工程师

2010 年 5 月 – 至今：天津商业大学，信息工程学院，讲师

研究兴趣

社区发现、机器学习、计算机视觉

基金项目

国家自然科学基金青年项目，61503281，基于主动异构监督的重叠社区发现及其模型选择方法研究，2016/01– 2018/12 ，23.4 万，主持

国家自然科学基金青年项目，61402467，航拍视频全景拼接、前景提取及压缩技术研究，2014/01–2016/12，27 万元，第一参与人

联系方式

通讯地址：北京市海淀区闵庄路甲 89 号中国科学院信息工程研究所

邮编：100089

E-mail：yangliang@vip.qq.com

个人主页：http://yangliang.github.io/

致　　谢

四年前，当我三十岁的时候，我决定要读博，而且是到中科院读博。最要好的朋友跟我说:“何必呢”。家人跟我说:“你有很多更重要的事情要做”。我所就职的高校的同行跟我说:“可能这个事情没看上去那么重要”。

转眼间，三年博士生活就要结束了，这也是我人生中最后一个学生生涯。回想起在信息工程研究所这三年来的学习生涯不禁思绪万千。在通往摘取人生最高学历的道路上，有太多值得感谢的人，正是有了这些人，我才能顺利走到今天。值此论文完成之际，谨在此向多年来给予我关心和帮助的老师、同学、朋友和家人表示衷心的感谢!

首先，最应该感谢的是我的博士导师操晓春研究员。我还记得第一次发邮件给操老师，希望申请他的博士生的时候，操老师立即给我回电话说:“如果你只想混个博士学位，就不要来读我的博士”。我还记得 12 年 9 月在信工所 3107 参加的第一次组会，操老师对每个公式的推导到严格要求。虽然之后的研究方向和操老师的研究方向不太一致，但是在跟操老师的学习过程中体会到了应该如何更好地思考问题，研究问题，解决问题。操老师渊博的学识水平与严谨的学术态度深深的影响着我，使我受益无穷。借此论文完成之际向操老师表达最崇高的敬意，最诚挚的谢意和最美好的祝福。

其次，我应该感谢天津大学计算机科学与技术学院的金弟副教授。正是金老师将我引向了社区发现及复杂网络分析的研究道路。在我最无助的时候为我指明了适合我的研究方向。从第一篇论文时的完全不知道如何组织文章结构，到现在可以独立的完成一篇论文，再到能否顺利申请到国家自然科学基金青年项目，金老师在科研的各个环节都给予了莫大的帮助。不仅仅是在学术上，在生活上金老师也给与了我很大的帮助。同时还要感谢美国圣路易斯华盛顿大学的 Weixiong Zhang 教授。虽然与其还未曾谋面，但在论文完成阶段 Zhang 老师在无论是整体立意还是细节的语言把握上都给予了我很大的帮助。

此外，我还有感谢我的师兄王啸。正是他把我引荐给了金弟老师，并在我最初开展社区发现研究的时候给了我很大的帮助。从需要在哪些数据集上进行试验到要和哪些方法进行对比，再到需要比较哪些性能度量标准。值此王啸师兄要去清华大学从事博士后研究工作之际，愿其能在学术上更上一层楼。

在博士期间，最值得我还念的，就是每月四次的讨论班，通过讨论班，我们共同学习了大量前沿的科研成果，也在头脑风暴中讨论出了很多的想法。在此，要感谢参加讨论版的各位老师和同学，包括郭晓杰、刘偲、许倩倩、张华、张玮、郭园方、韦星星、张长青、杜玲、马坷、任文琦、刘桐、王川、吴庆波、张三义和陈志勇。有了你们的共同讨论才有了丰富精彩的三年时光。

在我读博士期间，也要感谢我就职的学校天津商业大学的领导和老师们所给予的帮助，正式你们让我有了更多的时间专心从事科研工作。其中包括信息工程学院原书记刘振航，书记王艳军，院长潘旭华，副院长陈雷，计算机系系主任侯立坤，电子商务系系主任高岩，研究生院主任张立毅。

最后，也是最为重要的就是要感谢我的家人。感谢你们宽容一个 30 岁的人还能那么任性的去做自己想做的事情，感谢你们替我承担起了很多家庭的责任，感谢你们在我最为无助的时候给予我莫大的安慰和鼓励。我也要感谢离我远去的她，在我什么都没有的时候还能依然在我身边。我还要感谢未来的她，感谢你在耐心的等我，我这就要去和你一起共筑精彩的生活。

谨把本文献给我最敬爱的父母！

杨亮

2016 年 *3* 月

于北京