社交媒体复杂行为分析与建模

（申请清华大学工学博士学位论文）

培 养 单 位：计 算 机 科 学 与 技 术 系
学　　科：计 算 机 科 学 与 技 术
研　究　生：蒋　朦
指 导 教 师：杨　士　强　教　授

二〇一五年五月

Uncovering and Modeling Complex Behaviors in Social Media

Dissertation Submitted to

Tsinghua University

in partial fulfillment of the requirement

for the degree of

Doctor of Philosophy

in

Computer Science and Technology

by

Meng Jiang

Dissertation Supervisor : Professor Shiqiang Yang

May, 2015

关于学位论文使用授权的说明

本人完全了解清华大学有关保留、使用学位论文的规定，即：

清华大学拥有在著作权法规定范围内学位论文的使用权，其中包括：（1）已获学位的研究生必须按学校规定提交学位论文，学校可以采用影印、缩印或其他复制手段保存研究生上交的学位论文；（2）为教学和科研目的，学校可以将公开的学位论文作为资料在图书馆、资料室等场所供校内师生阅读，或在校园网上供校内师生浏览部分内容；（3）根据《中华人民共和国学位条例暂行实施办法》，向国家图书馆报送可以公开的学位论文。

本人保证遵守上述规定。

（保密的论文在解密后应遵守此规定）

作者签名：_____ 导师签名：_____

日　　期：_____ 日　　期：_____

摘 要

社交媒体已发展成为空前庞大的互联网平台，亿万用户行为记录中蕴含着巨大的科学和市场价值。精确的行为预测和检测技术是推荐系统、个性化搜索和社会化营销等众多领域的核心，而用户行为分析与建模是预测与检测技术的基础，成为计算机科学新颖而重要的问题之一。行为分析与建模面临着行为数据的高稀疏度、海量动态、多元异构和意图复杂等挑战。传统的行为分析方法并未充分考虑用户行为的复杂特性，因此急需紧握复杂行为的潜在规律以提升行为模型的准确性。本文从用户行为的上下文关联性、跨域跨平台性和真伪性三大特性出发，采用数据挖掘技术，运用行为学、心理学等知识，分析行为规律，设计建模方法，并实现预测和检测技术。本文的主要贡献如下：

1. 提出基于社交上下文和时空上下文的采纳信息行为分析模型。 为解决采纳信息行为的高稀疏性问题，本文融合兴趣偏好和影响力的社交上下文因素分析行为产生规律并建立模型 ContextMF。实验表明，融合模型显著好于单一因素模型。另一方面，结合行为的多面性和动态性特征，本文进一步提出基于时空上下文的进化分析方法 FEMA。在大规模数据上的实验表明，该模型方法能显著提升行为预测的效果，证实增量数据处理的高效和可靠性。
2. 提出社交媒体跨域行为和跨平台行为的迁移学习算法。 社交媒体用户的复杂需求只有在多域和多平台中才能得到满足。为解决单一域或单一平台的行为数据稀疏性以及冷启动问题，本文利用社交域桥接单一平台内的多个内容域，利用重合用户特征桥接多个社交媒体平台，迁移辅助域和辅助平台的行为信息，大幅提升了目标域和目标平台的预测效果。实验表明，跨域 HybridRW 算法和跨平台 XPTrans 算法在用户行为预测中具有优异表现。
3. 提出基于同步性和密集性的可疑行为分析方法和评价指标。 欺诈、垃圾传播、“僵尸粉” 关注等可疑行为严重威胁社交媒体安全、降低用户体验。本文抓住可疑行为的同步性和密集性特征，提出快速有效的分析方法 CatchSync 和 LockInfer，成功检测出社交媒体中可疑行为、还原被扭曲的统计分布。该方法优于基于内容的传统方法，并能与之互为补充。本文进一步提出量化跨维度异常行为可疑程度的新颖概率测度指标，并给出快速检测算法 CrossSpot 检测高维真实社交媒体数据中的信息操纵行为。

关键词：复杂行为；社交媒体；行为模型；数据挖掘

Abstract

The development of social media has enabled the collection of behavioral data of unprecedented size and complexity. All social platforms have realized that great scientific and marketing values are contained in the millions of billions of behavioral records. Accurate prediction and detection of user behavior are key techniques for many social media applications, such as recommender systems, personalized search and social marketing. Behavioral analysis and modeling is the starting point of these techniques. It has been one of the most novel and important research problems in computer science. Researchers are facing a number of challenges, including high sparsity, heterogeneity and abnormality, brought by complex social media environments. Traditional behavioral models did not take complex characteristics or mechanisms of user behaviors into consideration, so they fail to provide effective prediction and detection. This thesis studies contextual, cross-domain/cross-platform and suspicious behavioral patterns, develops a series of novel data mining techniques, and provides behavioral models, behavioral prediction and detection methods. Main contributions are summarized as follows.

1. *Proposing information adopting behavioral models based on social contexts and spatial-temporal contexts.* Social contextual model (ContextMF) incorporates two factors, personal preference and interpersonal influence, to predict article sharing and message retweeting behaviors. Experiments demonstrate that this model performs much better than those models with one single factor. This thesis also proposes flexible multi-faceted evolutionary analysis (FEMA) for dynamic behavior prediction in spatial and temporal environments. Large-scale experiments show that this method can significantly improve prediction performance and speed-up incremental learning.
2. *Proposing transfer learning algorithms for cross-domain and cross-platform behaviors in social media.* Social media users perform on multiple domains and multiple platforms to fulfill their information needs. To address high sparsity and cold start problems in a single domain or a single platform, this thesis proposes to utilize the social domain to bridge multiple domains in one platform and utilize overlapped users to bridge multiple platforms. It demonstrates that knowledge transfer from auxiliary domains and auxiliary platforms can significantly improve behav-

ioral prediction performance in the target domain and target platform. Experiments on real data show that HybridRW and XPTrans algorithms provide break-through performance in cold-start users' behavioral prediction.

3. *Proposing suspicious behavioral analysis and suspiciousness metric based on synchronicity and density.* Fraudsters, spammers and zombie followers have threatened the peace and user experience in social media. This thesis captures synchronized and lockstep characteristics and proposes scalable, effective suspicious behavioral detection algorithms CatchSync and LockInfer. The algorithm catches frauds and spam, and recovers distorted degree distributions. It outperforms content-based methods and is complementary to them. Furthermore, the thesis proposes a novel metric based on probability theory to evaluate suspiciousness in multi-modal behavioral data. CrossSpot, the local search algorithm based on the metric can effectively catch information manipulating behaviors in large-scale real social media datasets.

Key words: complex behavior; social media; behavior modeling; data mining

目 录

主要符号对照表

ALS	交替最小二乘法（Alternating Least Squares）
CF	协同过滤（Collaborative Filtering）
CPU	中央处理单元（Central Processing Unit）
DCG	折扣增益值（Discounted Cumulative Gain）
DDoS	分布式服务攻击（Distributed Denial of Service）
ERR	预计排名倒数（Expected Reciprocal Rank）
Frobenius	弗罗宾尼斯范数（Frobenius Norm）
Hadamard	阿达玛乘积（$\odot$）
HITS	超链接诱导主题搜索（Hyperlink-Induced Topic Search）
HOSVD	高维奇异向量分解（High-Order Singular Value Decomposition）
Jaccard	雅卡尔相似系数
LDA	潜在狄利克雷分配模型（Latent Dirichlet Allocation）
MAE	平均绝对误差（Mean Absolute Error）
MAP	平均准确率（Mean Average Precision）
NDCG	归一化折扣增益值（Normalized Discounted Cumulative Gain）
PCA	主成分分析（Principal Component Analysis）
Pearson	皮尔森相关系数（Pearson Correlation）
RAM	随机存储器（内存）（Random Access Memory）
ROC	接收器操作特性（Receiver Operating Characteristic）
RMSE	均方根误差（Root Mean Squared Error）
SVD	奇异向量分解（Singular Value Decomposition）
TF-IDF	词频 -逆文档概率（Term Frequency-Inverse Document Frequency）
UGC	用户产生内容（User Generated Content）

第 1 章 引言

社交媒体已经成为人类日常生活中难以割舍的一部分，社交媒体中诸如“关注”和添加好友、发布和分享内容等丰富的功能强有力地满足用户沟通交流、了解消息、获取知识乃至宣传和营销等复杂需求。用户行为的复杂特性对传统的行为预测和检测技术提出了前所未有的挑战。本文重点研究在复杂的社交媒体环境下，如何实现合理高效的用户行为分析与建模方法，并在真实社交媒体数据中验证预测与检测的应用效果。本章旨在阐述研究背景，简要回顾用户行为分析与建模技术的发展，给出本文的研究问题和所面临的严峻挑战，并描述本研究工作的主要贡献以及章节安排。

1.1 研究背景

信息时代里的社交媒体服务掀起了二十一世纪初期的一场科技革命，层出不穷的社交媒体应用和其巨大的市场潜力引发创业狂潮。这些应用让生活在地球不同角落的人们通过互联网联系在一起，交际成本从几万公尺缩短到分秒之间；让人们足不出户就能讲述自己在做什么，知道朋友们在聊什么、世界上在发生什么，寻找到自己想要了解的知识和消息。这一切都源于社交媒体提供了丰富而便捷的产生行为的舞台，社交媒体服务商储藏了亿万级的用户行为记录并持续增长中。常见的社交媒体服务商包括 Facebook① 、人人网② 等联系熟识好友的社交网站，Twitter③ 、新浪微博④ 、腾讯微博⑤ 等关注名人和频道的微博网站，LinkedIn⑥ 、猎聘网⑦ 等职位招聘社交平台，Netflix⑧ 、Pinterest⑨ 、豆瓣⑩ 等专注图片、音乐、电影、书籍推荐的兴趣类网站，乃至 Amazon⑪、eBay⑫、淘宝网⑬ 等连接买家卖家的购物网站。用户可以联络朋友、分享文章，跟踪名人和偶像、转发新闻和消

① 脸谱，美国社交网络服务网站：https://www.facebook.com
② 人人，中国领先的实名制社交网络平台：https://www.renren.com
③ 推特，美国微博客服务网站：https://www.twitter.com
④ 新浪微博，新浪网推出的微博客服务网站：https://www.weibo.com
⑤ 腾讯微博，腾讯网推出的微博客服务网站：https://t.qq.com
⑥ 领英，全球职场人士沟通平台：https://www.linkedin.com
⑦ 猎聘，中高端人才求职平台：https://www.liepin.com
⑧ 奈飞，美国最大的在线 DVD 租赁公司：https://www.netflix.com
⑨ 拼趣，瀑布流行展现内容的图片社交网站：https://www.pinterest.com
⑩ 豆瓣，生活和文化为内容的社交服务：https://www.douban.com
⑪ 亚马逊，美国最大的网络电子商务公司：https://www.amazon.com
⑫ 易购，全球民众线上拍卖及购物网站：https://www.ebay.com
⑬ 淘宝，阿里巴巴集团投资创立的网络零售商圈、电子商务平台：https://www.taobao.com

息，发布个人简历、求职和招聘消息，给喜欢的图片、页面点“赞”，给音乐、电影和书籍评分等。随着数据科学技术的迅猛发展，社交媒体逐渐意识到用户行为数据蕴藏的价值。用于广告投放、内容推荐、反欺诈等应用的用户行为预测、检测技术已经为社交媒体服务商带来不菲的市场盈利，事实上用户行为的研究在国民生活的多方面都有重大意义。

第一，社交媒体用户行为的研究对互联网服务和信息系统具有巨大的市场价值。社会化推荐系统和社交媒体营销已经成为重要的盈利模式。社交媒体将用户紧密结合在一起，人们获取信息、知识乃至购物需求有了更广阔的实现方式和达成空间。精确预测点击、购买行为，准确及时检测欺诈行为等基于大规模行为数据的应用创造了巨大的市场价值和经济效益。例如在美国，Facebook 是社交电商领域的领导者：71% 的成年网民是 Facebook 用户，Facebook 上分享的电商帖子平均可以转化成 3.58 美元的销售额 ①。根据 Shopify 的统计，Polyvore 的社交推荐订单平均值为 66.75 美元，Pinterest 为 65 美元，Facebook 为 55 美元。在中国，腾讯网的社交平台 QQ 和微信将庞大的用户群与合适的内容、服务连接在一起，实现了持续创新和增长 ②。QQ 智能终端月活跃账户于 2014 年末同比增长 33% 至 5.76 亿，而整体最高同时在线账户同比增长 21% 至 2.17 亿。2014 年末，微信及 WeChat 的合并月活跃账户同比增长 41% 至 5 亿。绑定银行账户的微信支付和 QQ 钱包账户超过 1 亿。2014 年全年，腾讯总营收 789.32 亿元，同比增长 31%；净利润 238.10 亿元，同比增长 54%。

第二，社交媒体用户行为的研究对国民生产和国家安全具有重大意义。社交媒体允许个人便利和快捷地制造、发布和传播信息，这打破了数世纪以来新闻和消息由国家和相关机构垄断的传统局面。另一方面，社交媒体中更快速、容易的分享与协作能够突破传统商业模式交易成本的束缚，个体行为影响国民生产、国家安全乃至社会发展的主动权被大大增强了。比如，社交网络中的解救走失儿童运动让很多失散多年的家庭重回温暖；微博曝光贪污腐败、社会丑闻等不良事件能够协助监督政府工作、人员管理；在李光耀等政治明星离世，李娜、刘翔等体育明星退役等事件中，社交媒体让公众有了集体向英雄和偶像表达敬意的机会；马航坠毁、漳州 PX 爆炸等事件的扩散让社会看到体制漏洞，提升警觉性。总而言之，社会环境与国家管理方式不再僵化，而变得活泼而有生命力、有专家知识也富含群体智慧、感染力强且易于煽动。充分理解社交媒体用户行为特性、调用其中的潜藏力量能够有效提速国民生产效率、维系和谐稳定的国民生活环境。

① 腾讯科技 -社交电商报告收入增速达三位数：http://tech.qq.com/a/20150121/005799.htm

② 新华网 -去年净利润 238 亿元同增 54% 腾讯移动社交广告收入大增：http://news.xinhuanet.com/finance/2015-03/19/c_127596663.htm

用户行为相关研究对实际应用很有意义。同时，如图1.1所示，用户行为的分析建模也有着非常重要的科学研究价值。

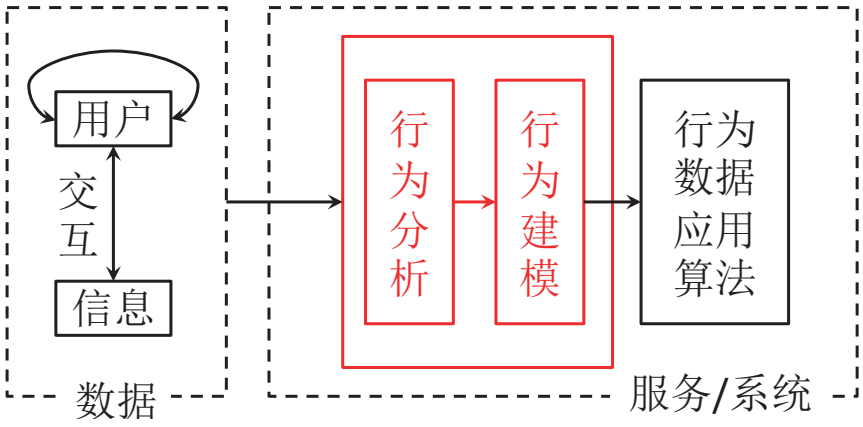


图 1.1　社交媒体中的用户行为较社交网络更丰富复杂。用户行为分析与建模是实现社交媒体服务（如社会化推荐、市场营销等）的第一步。

第一，社交媒体用户行为与传统社交网络用户行为相比更加丰富、复杂。传统的社交网络用户行为数据着重于用户之间的交互，如好友关系、关注关系、沟通频率、网络中扮演的角色等，而社交媒体的行为数据中含有更丰富的内容信息，如文本、图片、视频、超链接、表情符号等，同时这些行为也牵涉多种环境因素，如手机、平板、台式计算机等设备信息，旅馆、餐厅、超市等地理位置信息。用户在复杂环境中与内容信息的交互数据比起传统社交网络的用户行为数据更加丰富：用户数量在百万或亿级，而内容信息更是以每日上亿的数量产生和传播；交互行为作为两者的连接体，数量庞大。

第二，社交媒体用户行为的研究需要交叉学科知识的支持。依靠互联网技术搭建的社交媒体形成不同的行为产生机制，在不同的机制下产生的大量用户行为数据难以再单一依靠互联网技术分析。行为学、心理学、社会学、传播学和人类学等众多领域的知识都对理解用户行为数据、分析发现社会的运作规律具有指导意义。用户行为建模中的核心思想往往来自于这些领域的问题假设。比如，在社交媒体中，用户收到消息后为什么会采纳（即转发、分享等）或是拒绝（即忽略）；在不同社交媒体平台上，用户的行为规律是否贯穿一致；有恶意企图或是欺诈目的的用户是否会产生可疑行为，这些可疑行为与正常用户行为有何差异等。只有融合交叉学科知识的数据挖掘技术才能够合理有效地对用户行为进行分析和建模。

第三，社交媒体用户行为分析与建模是实现基于行为数据的应用算法的第一步。基于行为数据的应用算法多种多样：采纳信息行为预测算法可以为推荐系统、个性化搜索和社会化市场营销等应用提供保障，可疑行为检测算法可以为反欺诈、

反恶意、反垃圾传播和反信息操纵等安全问题提出解决方案。对行为预测来说，采纳信息行为的分析建模是技术实现的基础。比如，想要知道给定的用户是否会转发所推荐的微博消息，首先需要分析影响用户转发与否的因素，继而将这些因素用合理的数学模型表示，并通过机器学习方法进行训练，最后完成预测。再比如，想要知道用户是否会给给定电影打出高分，首先需要分析影响用户给电影评分高低的因素，再进行模型化、训练和预测。对可疑行为检测来说，分析欺诈者、垃圾传播者和僵尸粉的行为模式是基础。比如，想要检测出 Twitter 上的僵尸粉，首先需要分析产生僵尸粉并让其提升顾客粉丝数量的机制，从而知道僵尸粉的行为规律，并找到区分办法。再比如，想要检测信息操纵行为，必须预先分析操纵信息时社交媒体用户的行为特征。由此可见，用户行为分析建模是实现应用算法的第一步，是完善实际系统时不可或缺的环节。

1.2 本研究工作面临的主要挑战

社交媒体中既有的行为预测和检测技术在行为分析与建模的基础环节中仍然面临很多挑战，难以有效满足社会化推荐系统、个性化搜索、欺诈行为和信息操纵检测等应用需求。如图1.2所示，这些严峻挑战可以被归纳如下：

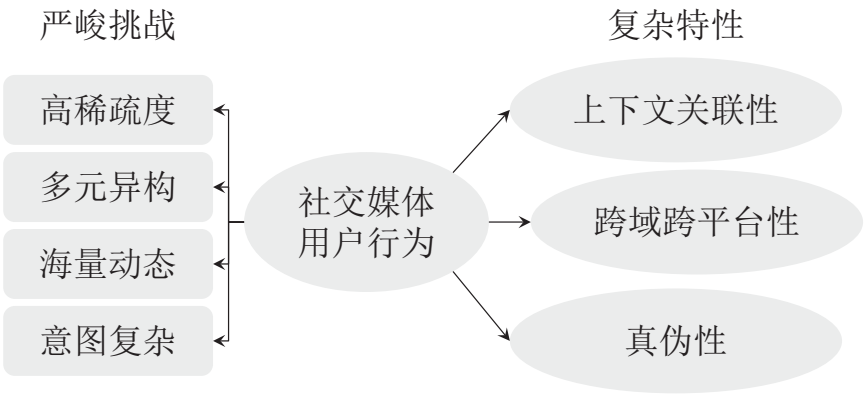


图 1.2 社交媒体用户行为分析与建模面临四大严峻挑战。只有深入分析用户行为的三大复杂特性，这些实际应用中常遇到的问题才能得以有效解决。

采纳信息行为的高稀疏度： 传统行为预测模型，如协同过滤技术等，都面临行为数据中的高稀疏度问题，如用户对电影评分的行为数据往往很稀疏，造成用户之间、电影之间关联程度估计的误差大，预测用户评分的准确程度低。比如，Netflix 拥有近百万个电影，然而每个用户会评价的电影大多不超过 200 部。由此所得到的 “用户 -电影” 评分矩阵是一个稀疏矩阵，密度往往不到百分之一。而在社交媒体中，随着用户数量增加、内容信息每日剧增，行为数据的密度会极端小，

近乎万分之一乃至十万分之一的级别。采纳信息行为的高稀疏度使得用户与内容信息的刻画非常困难，给行为预测准确度带来严峻挑战。本文从用户行为特性出发，通过深入理解行为产生规律，融合不同来源的行为数据，提升采纳信息行为预测的效果。

社交媒体用户行为的多元异构性： 社交媒体的复杂环境造成用户行为在多方面都呈现多元异构性。首先，行为权重在不同的社交媒体中不一致。比如，社交网络中用户交友行为是用二值（0 或 1）表示，兴趣网站中用户给电影、书籍评分的行为是用 1 到 5 的正整数值表示，微博网络中用户转发微博的行为可以用非负整数表示。用户行为还往往需要如无向图、有向图、二部图、带权图乃至超图的不同模型表示，需要使用（非）对称矩阵、二值矩阵、非负矩阵和非负张量等数学表示。其次，社交网络的用户节点不足以表示社交媒体的结构；异构社交媒体中依旧是以用户节点为主体，但其中还含有内容信息节点、设备节点等复杂的网络结构元素。比如，微博平台中有消息节点、社交标签节点、群组节点、视频节点等，兴趣网络中有书籍节点、电影节点和音乐节点等。社交媒体的多元异构性给用户行为建模带来巨大挑战。本文在为用户行为构建模型时，充分考虑行为的多元异构性，采用通用的矩阵、张量模型或是图模型表达用户行为。

用户行为数据的海量和动态性： 动态用户行为持续不断地向行为模型注入增量数据，反复处理海量动态的行为数据会造成应用系统瘫痪等异常。比如，当 Twitter 上新注册了一个用户或是新发布了一条消息，更新训练百万用户的行为模型计算复杂度很高，甚至会出现需要 25 个小时才能解决单日（24 小时）的数据分析任务。传统的用户行为模型难以有效应对这种状况，所以提出基于大规模社交媒体的增量行为数据处理模型以及在线算法尤为重要。本文在研究用户行为分析和建模时在应对增量数据的近似算法上有所突破，大幅度降低算法的时间复杂度。

用户行为意图的复杂性： 由于微博、微信等社交媒体具有传播快、范围广、影响大的特点，已经成为传播信息的重要途径。社交媒体在发挥积极作用的同时，也伴随发布隐私信息、传播谣言、信息操纵等违规现象。当全网用户在社交媒体分享、交流、互动的同时，有复杂行为意图的用户所产生的可疑行为不断地膨胀。当前迫切需要合理有效的行为分析方法和模型来净化网络空间，营造健康的社交媒体生态体系，因此，可疑行为分析技术已经成为研究者们普遍关注的热点问题。常见的可疑行为包括 “僵尸粉” 的关注行为、垃圾传播、散布谣言、信息操纵等行为。比如，一杯咖啡的钱可以买到 4000 个 Twitter 收听者；如果愿意花 3700 美元，就可以在 Instagram 上拥有 100 万好友。再比如，2011 年至 2012 年间，美国国务院为其 4 个 Facebook 主页进行了两次行销广告活动，共花费了约 63 万美元。尽管主页粉丝成功地从 10 万提升到超过 200 万，但真正关注和参与网页互动的粉丝

仅为 2%，其余都是僵尸粉 ① 。本文从可疑用户行为动机和意图出发，分析行为的产生规律，给出合理有效的解决方案，可用于多种可疑行为的检测任务。

1.3 本文的主要贡献

本文重点研究社交媒体用户行为，分析其在复杂环境中的复杂特性，如图1.2所示，包括上下文关联性，跨域跨平台性和真伪性。据此本文提出合理有效的建模方法，并应用于采纳信息行为的预测技术和可疑行为的检测技术。研究工作的重要贡献分为三点，总结如下:

1. **基于社交上下文和时空上下文的采纳信息行为分析模型：** 为解决采纳信息行为的高稀疏性问题，本文提出融合信息采纳、信息内容、社交关系和用户交互等社交上下文，进而提升行为预测效果的方案。结合社交媒体信息传播和采纳机制，本文挖掘出个人兴趣偏好和人与人之间社交影响力的两大社交上下文因素在采纳信息行为的并发作用，提出基于社交上下文的融合行为模型。实验表明，该上下文关联的融合模型显著好于基于单一因素的模型。另一方面，复杂的社交媒体环境营造出丰富的时空上下文，使得用户行为具有明显的多面性和动态性特征。本文提出用辅助信息作为灵活约束项、用高维张量刻画行为的空间多面性、用张量序列描述行为的时间维度动态性的进化分析方法。在大规模数据上实验表明，该分析方法能够利用行为的多面性和动态性特征，显著提升行为预测的准确程度，所提出的近似算法能够有效、快速地处理增量数据。这部分工作将在第3章详细描述。
2. **提出社交媒体跨域行为和跨平台行为的迁移学习算法：** 社交媒体用户在多域和多平台中满足复杂需求，例如在微博平台中，用户需要转发微博来表达自己的兴趣，需要编辑社交标签来描述自我特征，需要加入社交群组进行互动。这些用户也会注册兴趣类的评分网站，收集自己喜欢的电影、音乐和书籍。为解决单一域或单一平台的行为稀疏性以及冷启动用户（即新注册的用户）的问题，本文在单一平台利用社交域桥接单一平台内的多个内容域，重构社交媒体平台为围绕社交域的星状图，给出迁移学习思想的随机漫步算法。本文进一步提出利用重合用户特征桥接多个社交媒体平台，迁移辅助平台的行为信息，大幅提升目标平台的预测效果。实验表明，该算法在跨域行为、跨平台行为和冷启动用户行为预测中的效果比起已有算法有显著提高。这部分工作将在第4章详细介绍。

① 新华网 -美国国务院花 63 万美元在社交网络买僵尸粉：http://news.xinhuanet.com/world/2013-07/06/c_124967126.htm

3. **基于同步性和密集性的可疑行为分析方法和评价指标：**欺诈、垃圾传播、僵尸粉关注等可疑行为严重威胁社交媒体安全、降低用户体验。本文抓住可疑行为的同步性和密集性特征，提出快速有效的分析方法，成功从多个社交媒体数据中检测出可疑行为（如僵尸粉和信息操纵行为等），并还原被扭曲的统计分布（如幂律的出度分布等）。该方法优于基于内容的传统方法，并能与之互为补充。本文进一步提出量化跨维度异常行为可疑程度的新颖概率测度指标，并给出快速检测算法。实验表明，该算法能有效检测高维真实社交媒体数据中的信息操纵行为。这部分工作将在第5章详细描述。

第 2 章 研究现状与相关工作

社交媒体用户行为包括如转发微博、分享文章、电影评分等采纳信息行为和欺诈、垃圾传播和僵尸粉等可疑行为。考虑到用户行为的社交上下文和时空上下文关联性、跨域性和跨平台性以及真伪性。本章分别从采纳信息行为预测、跨域跨平台行为预测和可疑行为检测方法三个方面进行文献综述。

2.1 采纳信息行为建模和预测方法

本小节中结合常见的采纳信息行为的建模方法和预测技术，对协同过滤和社会化行为预测技术、兴趣爱好和影响力分析方法、动态分析和模型表示方法以及预测效果的评价指标进行梳理。

协同过滤行为预测技术： 协同过滤（Collaborative Filtering）是在推荐系统中已经成熟并广泛应用的技术。与传统上基于内容的过滤方法不同，协同过滤能够从用户集合中找到与给定用户有相似兴趣的用户，综合这些类似用户的信息评价，得知给定用户对信息的喜好程度。Balabanović 等提出基于内容的协同过滤方法实现文本推荐[1]。Sarwar 等提出基于项目的协同过滤推荐算法[2]。Karypis 介绍了基于项目的前 n 名推荐算法效果的评价结果[3]。Si 等提出灵活混合模型实现协同过滤[4]。进一步地，Si 等提出将协同过滤方法和基于内容的过滤方法通过混合指数分布模型融合起来[5]。Deshpande 等介绍了多种基于项目的前 n 名推荐算法[6]。于是 Herlocker 等给出一系列基于协同过滤的推荐算法的评价结果[7]。在 2005 年，Adomavicius 等给出对当时最流行的、有效的推荐算法的调研分析以及对下一代推荐系统的设想[8]。Gori 等提出基于随机漫步的推荐引擎算法 ItemRank[9]。协同过滤工作中影响最深远的要数 Koren 在 2008 年所提出的多面协同过滤模型，也就是采用矩阵因子化模型实现协同过滤算法[10]。更深一步地，Koren 等在 2009 年综述推荐系统中矩阵因子分解模型的种种技术，给出实验分析[11]。Harvey 等提出贝叶斯潜在变量模型实现协同项目的评分预测[12]。Agarwal 等提出快速的前 k 名检索模型实现推荐[13]。Zhang 等提出为低秩矩阵因子化分解模型注入诱导性约束项，实现协同过滤[14]。Fan 等提出用协同过滤因子分解法实现推荐系统[15]。2014 年，Shi 等尝试超越 “用户 -项目” 矩阵实现协同过滤技术，给出最新技术和常见挑战的调研[16]。协同过滤技术在取得不错成绩的同时，也存在以下缺点：用户与信息的交互行为数据非常稀疏；用户与信息增多，推荐系统的性能会逐步降低；冷启动问题，即如果没有对任何一个信息给出评价，就无法合理推荐，同样地，如果没

有用户对给定信息评价，信息也就无法被推荐。

社会化行为预测技术： 随着社交媒体的迅猛发展，社会化行为预测技术逐步为研究者们所重视。Ma 等提出用概率矩阵因子分解模型 SoRec 实现社会化推荐[17]。Konstas 等提出在社交网络中的协同推荐系统方案[18]。Chen 等给出超大规模中推荐信息项目的解决方法[19]。Ma 等于 2011 年提出用显式和隐式的社交关系实现社会化推荐[20]。进一步地，Ma 等提出用社交约束项提升社会化推荐系统性能[21]。Shi 等从照片共享网站中的信息实现个性化的地标推荐[22]。Noel 等提出用于社会化协同过滤新颖的目标函数[23]。Shen 等在健康领域的社交媒体中提出社会化高斯过程模型预测用户行为[24]。Zhu 等采用挖掘用户搜索行为的特征来服务查询内容推荐[25]。Liu 等提出 SoCo 系统，通过上下文关联性的推荐系统来提升社交网络的用户体验[26]。Sedhain 等分析了用户交互和活动实现社会化实体过滤推荐方法[27]。Tang 等采用局部和全局的社交上下文实现推荐系统[28]。2013 年，Tang 等总结了社会化推荐的一系列工作[29]。Qian 等提出一种个性化推荐方法来融合用户兴趣和社交圈子关系[30]。Sedhain 等设计了社会化协同过滤技术实现冷启动用户和项目推荐[31]。社会化推荐所面临的挑战比起传统推荐系统来说更加严峻，随着信息数量膨胀，社交数据的稀疏度能够接近万分之一乃至十万分之一，因此需要更出色、更合理的社交媒体用户行为建模方法完成社会化推荐。

社交用户兴趣爱好分析方法： Blei 等给出有深远影响的话题模型方法 LDA，依据文本和词汇的依赖关系，实现文本的自动话题聚类[32]。这常被用于刻画社交媒体用户的兴趣特征。Liu 等提出概率潜在兴趣偏好分析方法用于协同过滤算法[33]。Phelan 等运用 Twitter 数据实时分析用户话题偏好来推荐新闻[34]。Liu 等提出社交媒体中测量用户社会性和兴趣多样性的方法，并给出动态网络中的高效多样性排名算法[35]。Sanderson 等详细分析用户兴趣爱好和对应评价指标是否能够对齐[36]。Stefanidis 等尝试研究上下文关联的用户兴趣爱好[37]。Liu 等提出将用户兴趣通过个性化排序扩展化，提升协同过滤的效果[38]。Zhu 等为移动端用户分析上下文关联的个人兴趣偏好[39]。Narang 等在非结构化的微博数据中发现和分析话题层面的社交演变[40]。

社交影响力分析方法： Benjamin 对社会化行为的结构性进行系统分析，得出影响力会决定行为发生与否的结论[41]。Bond 等利用心理学实验证明社交关系会影响人的决定[42]。Bandura 等提出大规模通信/沟通中的社会化认知理论，发现社交关系对沟通效果的强作用[43]。Leskovec 等分析了推荐网络中的影响力模式[44]。Liu 等提出生成图模型，利用异质链路信息和文本内容刻画社交网络中节点的话题层面影响力[45]。Goyal 等提出在社交网络中学习影响力概率值的工作[46]。Huang 等给出采用人与人之间影响力的社会化推荐方案[47]。Cui 等运用概率混合矩阵分

解方法实现基于项目的社交影响力预测[48]，进一步地，Cui 等利用社交影响力预测实现用户和微博的排序[49]。Yang 等在社交网络中将好友关系和兴趣传播融合起来[50]。Yang 等提出基于社交圈子的在线社交网络推荐[51]。Huang 等从推荐系统的后验效应的角度探索社交影响力[52]。Chua 等用社交相关性为采纳信息行为设计产生式模型[53]。Cui 等利用数据驱动的方法，结合用户节点的影响力，预测信息传播的爆发可能性[54]。Cheng 等通过探寻自我一致这一行为特性的方法实现影响力最大化[55]。

社交信任关系分析方法： Massa 等设计了基于社交信任的推荐系统[56]。Jamali 等于 2009 年提出联合基于社交信任和项目的随机漫步推荐模型 TrustWalker[57]。Moghaddai 等提出在基于社交信任的推荐系统中采用反馈效果机制，称为 FeedbackTrust 方法[58]。Carminati 等提出在线社交网络中的可信任信息分享方法[59]。

用户行为的动态分析方法： 用户行为的动态特征[60] 和在线信息流的时序动态特征[61] 逐步得到重视。Sun 等在数据流和矩阵模型的基础上提出动态张量分析方法刻画数据的动态性[62]。Lin 等在 2008 年提出分析动态网络中社区发现和社区演变的方法[63]。Koren 等结合时序特征给出协同过滤的动态分析方法[64]。Kumar 等分析了在线社交网络的结构和结构演变过程[65]。Lathia 等关注于推荐系统中的时序多样性[66]。Xiang 等融合用户的长期兴趣和短期兴趣实现动态推荐[67]。Dunlavy 等运用矩阵和张量分解模型实现时序上的链路预测[68]。Yang 等从在线社交媒体中挖掘出信息传播的时序模式[69]。Radinsky 等提出在互联网中为动态行为进行建模和预测[70]。Rossi 等提出随时间演变的关系化分类以及组合方法[71]。Yu 等提出将时域上的关联性和模式上的差异性桥接起来，用于挖掘突然发生的事件[72]。Chen 等在 2013 年为推荐系统将用户随时间变化的采纳信息能力建模[73]。Radinsky 等提出网络中预测内容变化的算法[74]，进一步地，他们系统地提出网络中动态行为的学习方法、建模方法和预测方法[75]。Sun 等分析了动态星状网络中的多类信息的共同演化[76]。Wang 等采用概率生成模型分析研究主题随时间演化的特征[77]。Yuan 等针对 Twitter 用户发掘时空话题，即话题随着时间和空间变化而引起的演变[78]。Yuan 等进一步利用时间信息，实现了地理位置的推荐[79]。Zheng 等用多重相似度实现协同矩阵因子分解模型来预测药物和目标用户的交互信息[80]。Zhong 等在组合社交网络中为其动态的社交关系建模[81]，实现好友关系预测方法[82]，乃至用户行为学习和迁移[83]。行为动态性造成数据规模巨大，如何分析和使用大数据颇有挑战性[84]。

用户行为的模型表示方法： 一系列针对用户行为数据的数学表示和分析方法得到研究者们的认可，包括概率主成分分析[85]、概率矩阵因子化分解[86]、矩

阵摄动理论[87]、非负矩阵分解算法[88]、多协方差矩阵的通用成分分析[89]、非负矩阵分解的进化分析[90]、和正交的张量分解[91]。此外，Singh 提出协同矩阵因子分解方法来实现关系学习[92]。Cai 等设计了奇异值阈值算法实现矩阵填充[93]。Kong 等采用 $\mathcal{L}_{21}$ 正则项实现鲁棒的非负矩阵分解。另一方面，用户行为需要使用多维度数学模型进行表示。Sun 等提出立方奇异值分解方法实现个性化的网络搜索[94]。Cichocki 等提出非负矩阵/张量因子分解模型受约束的最小二乘法[95]。Bader 等提出 ASALSAN 方法用于语义图的动态分析[96]，给出稀疏因子化张量的快速 MATLAB 计算方法[97,98]。Huang 等检测高阶奇异值分解和 K-means 聚类算法的等价性，同步选取张量子空间和实现聚类算法[99]。Kolda 等提出可扩展的张量分解方法实现多面性数据挖掘[100]。Sun 等提出包括理论和应用的增量张量分析算法[101]。Symeonidis 等用高维张量降维方法实现标签推荐[102]。在 2009 年，Kolda 等认真分析了张量分解的方法和相关应用[103]。该工作已经得到了广泛应用，也产生了一系列的衍生工作。Grasedyck 提出张量数据的分层奇异值分解方法[104]。Rendle 等针对个性化标签推荐服务提出顺时针交互的张量因子分解模型[105]。Acar 等在数据填充问题中提出可扩展的张量因子分解方法[106]。Maruhashi 等在大规模的多元异构性网络中采用张量分析方法做模式挖掘[107]。Baskaran 等实现高效、可扩展的稀疏张量计算方法[108]。Kang 等在 2012 年提出 Gigatensor 方法，能够让可扩展的丈量分析增速 10 倍[109]。Wang 等提出多模态基于图的重排序算法，用于网络中图片搜索行为建模[110]。其他的一些行为表示方法包括 Burges 等提出用梯度下降方法来学习如何对项目排序[111]，Ng 等提出用特征向量和稳定性实现链路分析和预测[112]，Wang 等介绍了社交媒体图片的相关和差异性搜索方法[113]，以及 Ou 等提出多元异构的映射方法实现快速的相似度匹配[114]。

用户行为预测的评价指标： 常用的评价用户行为预测效果的评价指标包括如下几点：信息检索通用的预测效果评价标准[115]，准确率和召回率曲线与接收器操作特性曲线（ROC）特性[116]，ROC 的深度分析[117]，实验中的不等性和不可预测性分析[118] 和双样本测试方法[119]。Niu 等对前 k 项排序学习的方法进行了分析[120]，而 Lan 等接着对前 k 项排序算法讨论其是否对于排序来说足够[121]。

2.2　跨域跨平台的迁移学习算法

本小节中深入调研了迁移学习算法，该算法思想常用于解决利用稠密的辅助域和辅助平台行为知识，提升稀疏的目标域和目标平台行为预测效果。本小节还梳理了与迁移学习有关的常用模型方法。

跨域的迁移学习方法： Berkovsky 等提出基于协同过滤的跨域学习需要媒

介的思想[122]。Raina 等提出无标签数据中的迁移学习方法，又名自我教学学习（Self-taught learning）算法[123]。Winoto 等通过研究跨域推荐方法实现跨两个电影评分数据集的迁移学习[124]。Li 等为缓解目标域中高系数度的问题，提出跨域的协同过滤算法[125]。该方法率先提出一个问题：电影评分行为和书籍评分行为之间是否能做好迁移学习？进一步地，Li 等提出通过评分矩阵生成模型实现协同过滤的迁移学习[126]。Yang 等通过社交网络给出图片聚类的多元异构性的迁移学习方法[127]。Adams 等采用带高斯过程的概率矩阵因子化模型融合辅助信息、增进学习[128]。Cao 等在多个多元异构性的内容域中通过迁移学习实现协同链路预测[129]。在 2010 年，Pan 等调研了一系列的迁移学习方法，对其分类、解析并介绍创新性和未来方向[130]。Pan 等还提出了用于降低稀疏度的协同过滤思想的迁移学习方法[131]。Porteous 等融合辅助信息和狄利克雷混合过程的贝叶斯矩阵因子分解方法[132]。Zhuang 等从约束项角度在多个信息源中实现跨域学习[133]。2011 年，Li 给出关于跨域协同过滤的简短综述[134,135]。Pan 等通过多元异质的用户反馈、采用迁移学习实现缺失值预测[136]。Shi 等利用项目标签实现跨域的协同过滤，提升推荐效果和行为预测质量[137]。Zhu 等提出多元异构性的迁移学习方法实现图片分类[138]。Moreno 等提出 Talmud 方法实现若干个信息域之间的迁移学习[139]。Tang 等在 2012 年提出跨域协同推荐，实现了将跨域学习和协同过滤方法的融合[140]。Zhong 等分析了组合社交网络，并给出用户行为的可适应性迁移方法 ComSoc[141]。Zhou 等设计了含核的概率矩阵分解模型融合图结构信息和辅助信息[142]。Abel 等在社交网络中实现跨系统的用户建模和个性化任务[143]。Chen 等采用高维张量方法实现多域推荐，如 “用户 -标签 -电影” 等[144]。Gao 等通过聚类层次的潜在因子模型实现跨域推荐方法[145]。Hu 等通过跨域的三角化因子分解实现个性化推荐[146]。Lu 等提出可选择性的迁移学习方法实现跨域推荐[147]。Shapira 等在 Facebook 的推荐系统中比较了单域和跨域的数据特征[148]。Shi 等在分析社交标签时，给出了跨域的协同过滤方法[149]。Yang 等在主动学习的任务中给出迁移学习理论[150]。Zhao 等提出主动迁移学习方法实现跨系统的推荐算法[151]。Al-Shedivat 等提出监督迁移稀疏编码方法实现迁移学习[152]。Jing 等分析多个辅助域数据的多种组合，结合期望最大化方法训练概率矩阵分解模型，解决单一域的数据稀疏度问题[153]。Li 等用同一的特征表示描述不同域中的用户和信息内容，通过匹配方法提升推荐质量[154]。Tan 等通过混合图模型实现迁移学习方法[155]。进一步地，Tan 等提出 Multi-Transfer 方法，利用多个视图和多个信息源迁移学习目标行为[156]。Wang 等利用模型漂移实现主动迁移学习方法[157]。跨域和跨平台的迁移学习算法核心是明确利用某一个因素（如对齐用户、对齐内容等）来桥接多个内容域和行为平台。社交媒体复杂环境使得跨域和跨平台存在更多的机遇和挑战，

这些内容将在本文的研究工作中详细阐述。

高阶星状图模型、随机漫步算法和半监督学习方法： 首先，Gao 等提出二部图的同分割方法实现高阶星状的多元异构性数据同聚类[158]。进一步地，Gao 等提出基于一致性信息论的高阶星状的多元异构性数据同聚类[159]。其次，Tong 等设计了快速的含重启的随机漫步算法[160]。Avin 等在大量实验中分析了随机漫步算法在实践过程中的效果[161]。Safro 等在 2009 年提出提升随机漫步算法性能的方案[162]。Chen 等用随机漫步架构来使用负评分信息并产生合理解释[163]。最后，Liu 等基于带约束的非负矩阵分解模型实现半监督的多标签学习[164]。Li 等在 2007 年提出采用非负矩阵分解模型的半监督聚类方法[165]。Wang 等在 2008 年通过矩阵因子化分解模型实现半监督聚类[166]。Lee 等在 2010 年介绍了半监督非负矩阵分解模型[167]。

2.3　社交媒体可疑行为分析和检测方法

本小节中调研了实际应用中的异常检测方法，图分割算法，密集子图挖掘方法，社区结构检测方法，基于图的聚类方法，以及常用的图生成模型、真实数据集和异常检测方法。

实际应用中的异常检测方法： Tax 等利用分类器的不稳定性做异常检测[168]。Shekhar 等提出基于图的空间异常点的检测算法[169]。Noble 等给出基于图的异常检测方法[170]。Wong 等提出用于疾病突发的贝叶斯网络异常模式检测方法[171]。Gyöngyi 等采用 TrustRank 方法实现网络中垃圾信息的检测[172]。Idé 等给出计算机系统中基于特征空间的异常检测方法[173]。Liu 等从行为图中挖掘不闪现的程序错误[174]。Chau 等给出在线拍卖人网络中的欺诈个性检测方法[175]。Pandit 等进一步在在线拍卖网站中给出快速可扩展的检测欺诈行为的算法 Netprobe[176]。Willems 等设计了 CWSandBox 分析自动化的动态恶意软件[177]。Jindal 和 Liu 在 2008 年详细分析了评论网站中的垃圾传播行为，对异常行为检测领域有深远影响[178]。Benevenuto 等通过在线视频社交网络中检测垃圾传播者和内容推广者[179]。Lee 等利用社交蜜罐和机器学习方法揭示社交媒体垃圾传播者的行为规律[180]。Heard 等给出社交网络中的贝叶斯异常检测方法[181]。Stringhini 等提出在社交网络中检测垃圾散播者的解决方案[182]。Perez 等在 Twitter 上给个人资料的可疑程度做评价[183]。Xiong 等提出分层的概率模型实现异常用户组的检测[184]。Cao 等在大规模的在线社交服务中提供假冒账号的检测服务[185]。Hu 等在健康医疗应用中提出上下文关联性的异常检测方法[186]。Mukherjee 等基于异常检测方法探索消费者评论中的虚假消费者群体[187]。Song 等通过分析行为的配合关系，捕捉以组的形式

操纵市场的行为[188]。Akoglu 等从在线评论中利用网络效应检测欺诈性评论[189]。Beutel 等在 Facebook 的“喜欢”页面行为数据中，提出 CopyCatch 方法检测密集性的组攻击[190]。Hu 等在微博网络中检测社交化垃圾传播者[191]。Aggarwal 等探讨了 Twitter 中的粉丝用户究竟是正常的还是购买所得的[192]。De Cristofaro 等利用蜜罐技术来理解 Facebook 上“喜欢”页面行为的欺诈性[193]。Mao 等在电话网络中检测可疑的网络行为模式[194]。Shah 等在 Twitter 网络中提出 fBox 方法检测可疑的链路行为[195]。Yu 等在社交媒体分析方法里提出 Glad 方法检测可疑的群体行为[196]。

（二部）图分割算法和相关的异常检测方法： Karypis 等提出非结构化的图分割算法 METIS[197] 以及快速高质量的多层图分割架构[198]。Ding 等提出从网络图结构中分离不相连和近乎不相连的谱方法[199]。Chakrabarti 提出无参数的图分割和异常检测算法 Autopart[200]。在 2006 年，Cook 等总结了大多数的图数据挖掘方法[201]。Dhillion 等在 2007 年进一步提出不采用特征向量就可以实现带权图分割[202]。Eberle 等在图数据中检测结构化异常，运用在图结构的系统中[203]。Moonesinghe 等采用随机漫步算法实现基于图的异常检测架构 OutRank[204]。Akoglu 等提出 OddBall 方法在带权图中观察到异常现象[205]。在 2010 年，Aggarwal 等总结了管理和挖掘图数据的技术和方法[206]。Feng 等用基于压缩方法实现图挖掘算法来分析图结构[207]。2014 年 Akoglu 等调研了一系列基于图的异常检测以及异常描述方法[208]。Koutra 等提出 VoG 算法为大规模图提供可理解的摘要信息[209]。另一方面也存在一系列的基于二部图研究工作：Dhillon 等提出实现文档同聚类任务的二部图谱分割算法[210]。Zha 等提出二部图分割和数据聚类算法的共通性[211]。Sun 等在二部图中实现邻居形成和异常检测的分析方法[212]。Feng 等给出摘要方法实现二部图挖掘的思想[213]。Fang 等通过稀疏正则化和核扩展方法实现基于图的学习[214]。

密集子图和近似闭环的挖掘方法： Asahiro 等提出寻找密集子图的贪心算法[215]，而 Charikar 提出了密集子图检测的贪心的近似算法[216]。Yan 等提出频繁的闭环图模式 CloseGraph[217]。Pei 等于 2005 年提出革命性的跨图近似闭环的检测方法[218]。在 2009 年，Jiang 等基于 2005 年 Pei 等的工作提出频繁的跨图近似闭环挖掘方法[219]。Andersen 提出了在局部数据中寻找密集子图的算法[220]。Lahiri 等提出在动态网络中的周期性子图挖掘方法[221]。Lee 等调研了一系列的密集子图发掘工作[222]。Miller 等利用带 $\mathcal{L}_1$ 正则项的特征向量实现子图检测方法[223]。Zou 等在不确定的图数据中挖掘频繁子图模式[224]。Giatsidis 等提出 D-cores 方法评价有向图中的协同关系[225]。Bahmani 等基于 MapReduce 系统在流数据上检测最为密集的子图结果[226]。Chen 等探讨了密集子图抽取和社区检测之间的关联性[227]。

Tsourakakis 提出通过质量保障约束项抽取优化的近似闭环，使得这种子图比起最密集子图更有意义[228]。2015 年，Balalau 等采用最大整体密度和有限重合寻找密集子图[229]。

社区结构的检测方法： Clauset 等给出初步的大规模网络中检测社区结构方法[230]。Newman 采用社交关系矩阵的特征向量发掘网络中的社区结构[231]。Brown 等给出在线社交网络的社区概念[232]。Chi 等通过社区因子化分析博客空间的结构性和时序性[233]。Rosvall 等提出复杂网络中解决社区结构发现的信息论架构[234]。Wakita 等设计了在超大规模社交网络中寻找社区结构的方法[235]。Leskovec 等探讨了在大规模的社交网络和信息网络中的社区结构统计属性[236]。Satuluri 等基于随机流给出用于社区发现的可扩展图聚类方法[237]。Fortunato 在 2010 年总结了在图数据中检测社区的一系列方法[238]。Prakash 等提出用特征子空间的形状分析和切割出大规模图中的社区，方法名为 EigenSpokes[239]。Wu 等通过分析邻接矩阵的特征空间里正交线形，将社区从图中分割出来[240]。

基于图的（谱）聚类方法： Ng 等提出谱聚类方法，已经被广泛用于社区发现中[241]。Wang 等给出多类别内部关联数据的聚类方法 ReCoM[242]。Schaeffer 在 2007 年总结了一系列图聚类算法的工作[243]。Zhou 等在多元异构性网络中实现作者和文档的同排序[244]。Chen 等于 2009 年给出多维度架构 OLAP 实现图数据分析[245]。Fu 等提出贝叶斯重合子空间的聚类方法[246]。Huang 等在摄动数据中给出谱聚类方案[247]。Kriegel 等介绍了高维度数据聚类的多种方法，包括子空间聚类、基于模式聚类和相关度聚类[248]。Müller 等评价高维度数据中的子空间映射形成的聚类效果[249]。Yan 等设计了快速近似算法实现谱聚类[250]。Gunnemann 等解释了子空间聚类在密集子图挖掘中的作用[251]。Trappey 等采用专利重合行为进行专利聚类[252]。Wauthier 等通过迭代削减不确定性实现活跃的谱聚类[253]。Shen 等通过谱方法将社区结构检测出来，并给出深入分析[254]。Ren 等在论文引用的网络结构中利用聚类方法建模[255]。Günnemann 等通过特征向量，将子空间模式和子图聚类方法高效结合起来[256]。

常用的图生成模型和真实数据集： 研究者们提出了很多种随机图的生成模型方法，包括随机图生成方法[257]、网络拓扑图结构的幂律分布[258]、幂律分布的随机图生成方法[259]、给定度数的随机图中节点之间平均距离[260]、复杂网络机制的统计分析[261]、闭环模式的生成方法[262]、社交网络中指数分布的随机图模型[263]、社交网络的随机分布分析[264]、以及社交网络的 Kronecker 图模型化方法[265] 等。在基于图的异常检测研究工作中，一个经典问题是 Zarankiewicz 问题，其近似解决方案在上世纪被提出[266,267]，近年里 Babai 等关于此问题给出了基于图的谱分析结果[268]。此外，Broder 等在网络的分布中观察到的尖峰往往意味着可疑的图结

构[269]。常用的基于图的真实数据集包括美国专利引用数据集[270]、支撑向量数据描述[271]、网络包的数据集[272]、美国医疗系统医生乱开药方并获取药物金钱利益的数据集[273]、电信通讯的欺诈行为数据集[274]、Twitter 社交网络数据集[275]、社交网路数据分析的简介[276]、和微软学术搜索数据集[277]。

常用的异常检测方法： 首先，研究者们提出一系列异常结构特征提取方法，包括带权中位数过滤的尖峰检测方法[278]，奇异值分解方法[279–281]，PageRank 值[282]，超链接环境下的 HITS 值（枢纽度和权威度）[283] 和超链接数据中的知识挖掘方法[284]。Chapelle 等通过等级相关性给出期望排序值[285]。第二，最短描述长度和密度比例估计等方法常用于实际应用，包括最短数据描述方法[286]、基于最小描述长度的子结构发掘[287,288]；Hido 等借助密度比例估计实现异常检测[289]，Sugiyama 等也曾尝试借助密度比例估计的方法在高维度空间中降低数据维度[290]，Hido 等采用直接密度比例估计方法统计性地实现异常检测[291]。Chandola 等在 2009 年调研了一系列的异常检测方法[292]。Song 等在同年提出了关系化的新颖性检测方法[293]。Sotiris 等通过贝叶斯支撑向量机实现异常检测方法[294]。Muandet 等提出单类支撑测量机实现异常组的检测[295]。

2.4　本章小结

本章对于社交媒体用户行为的相关研究工作进行了梳理，通过与之比较可知，本文的不同之处主要在于以下几点：

1. *本文着重于分析采纳信息行为的内在规律。* 社会化行为预测技术往往停留在表面现象，也就是只考虑协同特性、信息内容或是社交关系知识，但社交媒体的复杂环境形成的社交上下文关联机制才能反映社交媒体中用户行为产生规律：当用户接收到一则消息，会从个人兴趣偏好和人与人之间社交影响力的两大社交上下文因素来决定是否采纳该信息。另一方面，采纳信息行为是受到时空环境约束的，复杂的空间因素造成行为的多面性，需要采用高维度模型来描述，复杂的时间因素造成行为的动态性，需要用序列或数据流来描述。所以本文结合时空上下文为用户行为重新构造模型。
2. *本文着重于分析桥接多域和多平台行为的纽带。* 在单一平台的社交媒体中，桥接微博域、社交标签域、视频域和群组域等信息域的自然纽带是社交域：因为是用户转发微博、编辑社交标签、观看视频、加入群组等行为将多个信息域联系在一起。传统的跨域迁移学习方法并没有考虑到这一纽带特性。而这就要求重构社交媒体为一个以社交域为中心的星状图，详细工作会在本文中给出。多个平台的纽带也并没有在传统工作中讨论过，经过数据技术分

析，本文发现充分利用平台之间的重合用户能够将多个平台的行为数据联系起来，提升目标平台上的行为预测效果。本文深入分析桥接多域、多平台的纽带，给出跨域、跨平台的架构，并在真实数据集中验证所提出的迁移学习算法的有效性。

3. 本文着重于分析可疑行为的意图动机和产生规律。传统的可疑行为检测方法往往关注内容特征，比如微博文本中是否有超链接、是否有广告词汇等，而这些只是表面现象，可疑行为的实质是其不良意图和动机带来的产生规律。另一方面，可疑行为对社交媒体中富含关系的图结构产生的扭曲作用往往难以通过已有的社区结构检测、谱聚类、图分割、密集子图挖掘等方法找到。本文着力于分析欺诈、垃圾传播、僵尸粉关注等可疑行为的同步性和密集性特征，提出快速有效的检测方法，用以在社交媒体数据中检测僵尸粉和信息操纵行为。

通过分析社交媒体用户行为的内在规律，由此建立行为模型，实现的行为预测和检测技术，与传统技术相比较，具有准确率高、运行速度快、可解释性强等多个优势，突出了本研究工作的独特性。

第 3 章　上下文关联性的采纳信息行为建模

本章从上下文关联性的角度介绍社交媒体用户采纳信息行为的建模方法。首先介绍基于社交上下文的行为预测模型，接着介绍基于时空上下文的行为模式分析方法，第三小节介绍性能评测结果，并在最后小结本章内容。

3.1　基于社交上下文的行为预测模型

本节介绍社交媒体用户采纳信息行为的社交上下文关联性，并给出行为预测模型。内容包括引言、相关工作、采纳信息行为的社交上下文因素分析以及基于社交上下文的离线处理和在线增量模型。

3.1.1　本节引言

社交媒体亿万级的用户每天产生着大量信息，因而迫切需要高效准确地帮助用户找到有用信息的社会化推荐系统。传统的协同过滤技术并没有考虑到社交关系在社会化推荐中的作用，无法提供准确的推荐结果[8]。研究者开发出利用社交关系来正则化用户特征空间的社会化推荐系统[17,21]。然而，社交上下文信息，如用户交互、社交关系、采纳信息行为和信息内容，并没有得到充分利用。将丰富的社交上下文信息通过社交上下文因素融合为一个统一的社会化推荐框架是颇具挑战性的。

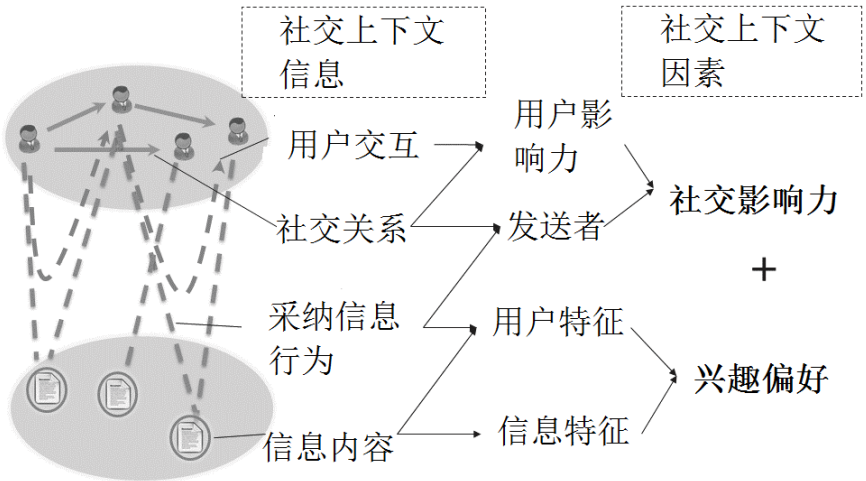


图 3.1　社会化行为预测模型学习上下文信息并融合成两大社交上下文因素，即个人兴趣爱好和社交影响力。该模型分析了社交媒体中用户行为形成的规律。

图3.1展示了如何从社交媒体的丰富链接性数据中利用所有的上下文信息进行推荐。每当收到一条社交媒体信息，用户会首先检查信息内容是否符合其兴趣爱好。例如，在 Twitter 上用户每收到一则来自朋友的微博，通常会阅读内容看看是否有趣。由此预测算法需要获取信息内容和采纳信息行为的知识。同时，用户还会在意发来消息的人是谁，看看这个人是否是熟悉的朋友或者是名人。如果超过一个朋友给他发来同样的内容，他可能会更关注这条消息。这样的知识可以从社交关系和交互讯息中获得。这两方面对于一个用户决定是否采纳信息（分享文章或是转发微博）都非常重要。上述可以总结为两大社交上下文因素，个人兴趣爱好和社交影响力。

心理学和社会学研究已证明个人兴趣爱好和社交影响力会影响人们在采纳信息时的决定。和纯粹由个人兴趣驱动、独立去做决定相比，个人很容易某种程度上受到某些人行为的影响[42]。基于网络的评价音乐实验也证实了人与人之间的影响力被融合进个人兴趣爱好后共同做决定，这会使得人类行为更加复杂，因而预测难度也会大大增加[118]。所以，只有当个人兴趣爱好和社交影响力被合理地融合进推荐系统中，才能降低行为不确定性，提升预测效果。

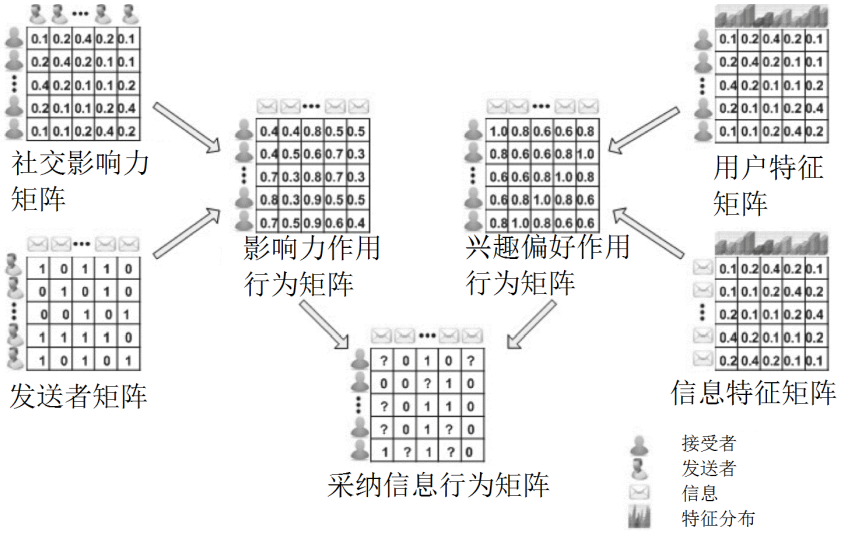


图 3.2　基于概率矩阵分解的社交上下文推荐模型：融合兴趣爱好和社交影响力。

为了解决上述问题，如图3.2所示，本工作中提出了一个全新的社交上下文行为预测模型（即推荐框架）ContextMF。该框架基于概率矩阵分解模型而设计，融合了个人兴趣爱好和社交影响力两大因素来提升社会化推荐的准确度。具体来说，ContextMF 首先将采纳信息行为矩阵分解为两大块，兴趣爱好行为矩阵和社交影响力行为矩阵。而这两大矩阵可以由三个矩阵得到，分别是用户兴趣特征矩阵、信息话题特征矩阵和影响力矩阵。模型从采纳信息数据和用户交互数据能够部分观测到用户的兴趣爱好和社交影响力，利用观测数据可以计算这三个矩阵。本工

作进一步设计了运行时间与推荐用户和信息数量成正比的可扩展算法。该算法可以在大规模离线处理的基础上处理增量数据，实现大型社交应用的实时推荐。

工作中在两大真实的社交媒体数据集上评测算法效果，一是中国流行的社交网站人人网，二是中国最大的微博平台之一腾讯微博。这两大数据集代表了两种经典的社交网络结构：一是双向社交关系（须验证的好友），二是单项社交关系（粉丝和被关注的人）。实验证明社交上下文推荐模型能够在社交数据上有效提升推荐系统性能。ContextMF 比起过去的模型算法有很大的效果提升。由此证明了兴趣爱好和影响力两大社交上下文因素的实际应用意义。

3.1.2　相关工作

本节综述了传统的行为预测算法。基于内容的过滤算法和协同过滤算法已经被广泛用于帮助用户找到最有价值信息的系统中。随着社交媒体的涌现，研究者们利用社交关系知识设计出了基于信任和基于影响力的方法。模型方面，矩阵分解方法因其在大规模数据集上的高效被用于社会化推荐中。然而，了解社交用户行为形成的潜在机制，从用户采纳信息的行为动机角度充分利用社交上下文信息，实现高效预测，是很有价值的。

基于内容的过滤技术介绍了通过信息内容对信息进行排序的基本想法。因为如 LDA（Latent Dirichlet Allocation，潜在狄利克雷分配模型）[32] 等话题聚类技术的出现，基于内容的方式方法[1,34,37,39] 根据用户所收到的内容与其喜好的话题分布的匹配程度对信息进行排序。但是这些侧重于个人层面的解决方案并不能够完整地从采纳信息行为数据中学习用户行为模式。

被广泛应用的协同过滤算法包括基于记忆和基于模型两种。基于记忆的方法根据采纳信息行为来计算用户相似度[2,5,11,19]。基于模型的方法则是从行为数据中学习用户行为的模式特征。研究者提出一种基于模型的三层（用户 -兴趣 -信息）协同过滤算法来实现个性化推荐系统[38]。然而，协同过滤只是利用采纳行为数据，但不能充分利用社交关系、交互频率和信息内容。

近年来，研究者提出矩阵分解模型来解决这类问题[11,14,90,296]。矩阵模型聚焦在将用户的采纳信息行为用低维度特征向量来表征[26,28,29,114]。另一方面发现社交影响力是决定社交媒体动态性的原动力[24,44,46,47]，基于影响力的推荐算法把人与人之间的交互信息融合到系统中[48,49,53]。基于信任的方法通过分析用户采纳信息的直接或间接信任程度来描述信任网络关系并实现推荐[56–59]。研究者提出了一种把用户品味和信任联系起来的概率因子分析模型来实现推荐[17]。另一个社交正则化的矩阵分解模型利用社交关系强度来提升推荐效果[21]。然而这些工作都只关注了用户相互之间的关系，而忽略了用户本身对于内容的期待。他们没有完整地从

上下文因素的两大层面理解采纳信息行为。如何理解用户采纳信息的动机、有效整合来提升推荐准确度依旧值得研究。

心理学和社会学的研究者们通过社会认知理论证明交互系统存在两个途径[43]。一条直接途径是了解参与者所喜欢的内容，另一条社会化途径是发现参与者的决定受到他们的朋友影响。另一个研究表明相近的观点，如认知、感受、品味、兴趣以及人与人的关系形成社会行为和交互的结构[41]。在社交媒体中，这两个因素恰恰就是个人兴趣爱好和社交影响力。所以需要基于社交上下文的推荐模型，该模型能够分析用户行为动机和社交应用机制来改善社会化推荐系统。本文融合兴趣爱好和社交影响为一体来模型化社会化行为。

3.1.3　采纳信息行为的社交上下文因素分析

本节在大规模真实社交媒体数据集中证实社交上下文因素（包括兴趣爱好和社交影响）在社会化推荐中的存在和重要性。给定一则信息，用户采纳信息的行为一方面依赖于个人兴趣爱好，也就是要了解用户是否喜欢这则信息；另一方面，社交影响力则表达出用户是否与发来信息的人（如 Twitter 上所关注的发微博的人）有密切联系。应用 LDA 技术从社交信息内容（如文章、微博等）中提取它们的话题分布。根据用户历史行为，简单获得用户 u 对于消息 a 的兴趣符合程度：

$$P_u(a) = T_a \cdot (\frac{1}{|A(u,a)|} \sum_{a' \in A(u,a)} T_{a'}) \tag{3-1}$$

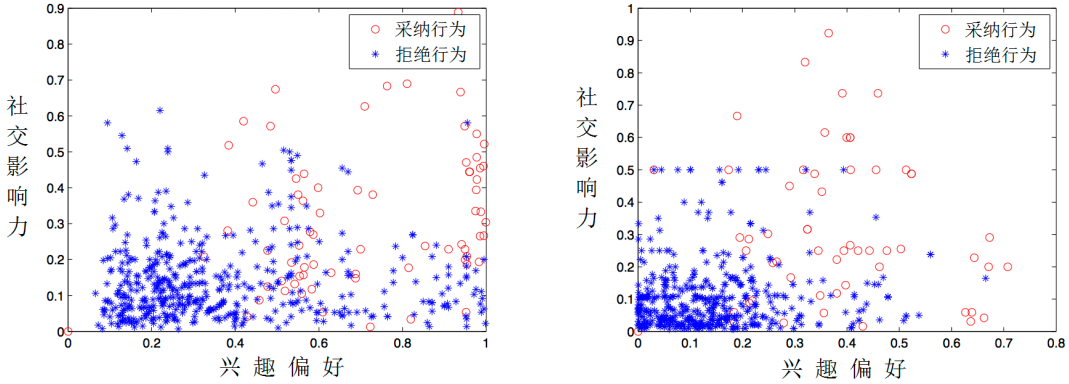
其中 $A(u,a)$ 是除去 a 以外用户 u 所采纳的信息集合，T_a 是信息 a 的话题分布。用 u 的朋友所发来的消息被采纳的比率来表征他们之间社交影响力的强度：

$$I_u(a) = \frac{1}{|V(u,a)|} \sum_{v \in V(u,a)} \frac{|\mathcal{S}(u,v) \cap \mathcal{A}(u)|}{|\mathcal{S}(u,v)|} \tag{3-2}$$

其中 $V(u,a)$ 是将信息 a 发送给用户 u 的用户集合，$\mathcal{S}(u,v)$ 是从用户 v 发送给用户 u 的消息集合，$\mathcal{A}(u)$ 是用户 u 所采纳的信息集合。

工作中把信息根据用户行为分为被采纳的和被拒绝的两种。在图3.3中画出每对用户 u 和信息 a 的个人兴趣爱好 $P_u(a)$ 和社交影响力 $I_u(a)$，可以发现用户无论在类似 Facebook 还是类似 Twitter 的社交媒体中都采纳既具有高的兴趣爱好值，又具有高的影响力值的信息。

为了验证兴趣爱好和社交影响力不仅仅是有效的而且还是互补的社交上下文因素，计算这两个因素在社会化推荐中的相关度。简单地用 P 和 I 来表示一个用户采纳信息的兴趣爱好值和影响力，它们的皮尔森相关系数（Pearson correlation）



人人网用户分享文章行为分布　　腾讯微博用户转发微博行为分布

图 3.3　两大社交上下文在人人网和腾讯微博用户行为上的分布：采纳信息行为往往都比起拒绝信息行为有更强的兴趣爱好和社交影响力。

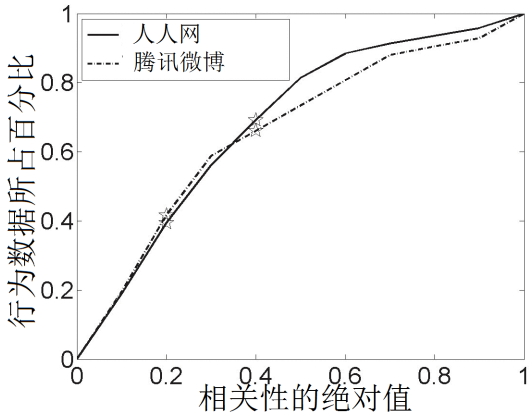


图 3.4　兴趣爱好和社交影响力这两大上下文因素几乎没有关联性：他们的绝对值在超过 40% 的用户上不到 0.2，在超过 70% 的用户上不到 0.4。

可以定义为：

$$\rho_{P,I} = \frac{cov(P,I)}{\sigma_P \sigma_I} = \frac{E[(P-\mu_P)(I-\mu_I)]}{\sigma_P \sigma_I} \tag{3-3}$$

相关系数是 1 则表示正线性相关，-1 表示负线性相关，而 0 表示没有关联。图3.4说明在超过 40% 的用户上具有不到 0.2 的相关性绝对值，在超过 70% 的用户上具有不到 0.4 的相关性绝对值。所以兴趣爱好和社交影响力可以在社会化推荐中看作互补的两大社交上下文因素。

3.1.4　基于社交上下文的采纳信息行为模型

下面介绍基于矩阵分解的社交上下文推荐模型 ContextMF 的细节。首先定义社会化推荐问题。假如存在 M 个用户（用 u_i 表示第 i 个用户），存在 N 条信息（用 p_j 表示第 j 条信息）。采用表示信息采纳矩阵为 $\mathbf{R} \in \{0,1\}^{M \times N}$，其中第 (i,j) 个

矩阵元素为

$$R_{ij} = \begin{cases} 1 & \text{如果用户 } u_i \text{ 采纳信息 } p_j \\ 0 & \text{否则} \end{cases}$$

那么社会化推荐问题可以定义为基于观测到的矩阵和其他因素，来预测信息采纳矩阵 $\mathbf{R}$ 中未被观测的矩阵元素。

在 ContextMF 模型中，假设社交媒体用户是否采纳一则信息受到三方面约束：（1）信息内容：这个信息讲述了什么；（2）用户信息交互：用户喜欢什么样的内容；（3）社交关系和用户交互：发来消息的是谁。用 $\mathbf{U} \in \mathbb{R}^{k \times M}$ 表示用户特征矩阵，用 $\mathbf{V} \in \mathbb{R}^{k \times N}$ 表示消息特征矩阵。$\mathbf{S} \in \mathbb{R}^{M \times M}$ 是社交影响力矩阵，其中矩阵元素 S_{ij} 表示用户 u_i 对 u_j 的影响力。要注意的是，只有在 Facebook 和人人网上用户 u_i 是用户 u_j 的好友，或者在 Twitter 和腾讯微博上用户 u_i 关注用户 u_j，那么，$S_{ij} > 0$。$\mathbf{G} \in \mathbb{R}^{N \times M}$ 是消息发送者的矩阵，其中矩阵元素 $G_{ij} = 1$ 表示用户 u_j 发送消息 p_i，反之则为 0。基于上述表示和用户只会收到来自他们在社交网络上朋友的消息的假设（也就是 $G_{ii} = 0$），可以表述社会化推荐问题为找到合适的矩阵 $\mathbf{U}$，$\mathbf{V}$ 和 $\mathbf{S}$，使得 $((\mathbf{S}\mathbf{G}^\top) \odot (\mathbf{U}^\top \mathbf{V}))$ 能够很好地、避免过拟合地近似观测矩阵 $\mathbf{R}$，其中 $\odot$ 是 Hadamard 积。

输入信息包括消息内容，用户采纳信息的行为和用户交互行为，由此产生消息内容表示，用户兴趣爱好表示和社交影响力表示。由下方表达式计算人与人之间的兴趣偏好相似度为矩阵为 $\mathbf{W} \in \mathbb{R}^{M \times M}$，消息内容相似度矩阵 $\mathbf{C} \in \mathbb{R}^{N \times N}$ 和社交交互矩阵 $\mathbf{F} \in \mathbb{R}^{M \times M}$：

$$W_{i,j} = \frac{\sum\limits_{a \in \mathcal{A}(u_i)} P_{u_i}(a)}{|\mathcal{A}(u_i)|} \cdot \frac{\sum\limits_{a' \in \mathcal{A}(u_j)} P_{u_j}(a')}{|\mathcal{A}(u_j)|} \tag{3-4}$$

$$C_{i,j} = T_{a_i} \cdot T_{a_j} \tag{3-5}$$

$$F_{i,j} = \frac{|\mathcal{S}(u_i, u_j) \cap \mathcal{A}(u_i)|}{|\mathcal{S}(u_i, u_j)|} \tag{3-6}$$

虽然相似度矩阵 $\mathbf{W}$ 和 $\mathbf{C}$ 的准确度受到 LDA 技术在数据中效果影响，但对于实验中不同的社会化推荐算法，采用同样知识矩阵学习和预测行为是公平的。

假设特征空间中的用户和信息相似度与观测数据中的相似度是一致的，利用观测矩阵信息，采用如下方法正则化三个特征空间中的矩阵：

- 用户在特征空间 $\mathbf{U}$ 上的相似度源自兴趣爱好相似度矩阵 $\mathbf{W}$；
- 信息在特征空间 $\mathbf{V}$ 上的相似度源自内容相似度矩阵 $\mathbf{C}$；
- 用户的影响力矩阵 $\mathbf{S}$ 源自社交交互有效频率矩阵 $\mathbf{F}$；
- 用户特征矩阵 $\mathbf{U}$ 和信息特征矩阵 $\mathbf{V}$ 乘积表示兴趣偏好程度；

- 通过兴趣爱好和影响力分别得到的行为预测矩阵之间的 Hadamard 积与实际采纳行为的概率矩阵成正比。

由于模型在测试集合上的预测效果通常用均方根误差（Root Mean Squared Error，简称 RMSE）来评价，采用带高斯噪声的概率线性模型。这里定义观测矩阵 $\mathbf{R}$ 的条件分布为

$$P(\mathbf{R}|\mathbf{S},\mathbf{U},\mathbf{V},\sigma_R^2)=\prod_{i=1}^{M}\prod_{j=1}^{N}\mathcal{N}(\mathbf{R}_{ij}|\mathbf{S}_i\mathbf{G}_j^\top\odot\mathbf{U}_i^\top\mathbf{V}_j,\sigma_R^2) \tag{3-7}$$

融合社交上下文因素并定义后验分布为

$$\begin{aligned}
& P(\mathbf{S},\mathbf{U},\mathbf{V}|\mathbf{R},\mathbf{G},\mathbf{W},\mathbf{C},\mathbf{F},\Omega)\\
= \;& \frac{P(\mathbf{R},\mathbf{W},\mathbf{C},\mathbf{F},\mathbf{G}|\mathbf{S},\mathbf{U},\mathbf{V},\Omega)P(\mathbf{S},\mathbf{U},\mathbf{V}|\Omega)}{P(\mathbf{R},\mathbf{G},\mathbf{W},\mathbf{C},\mathbf{F},\Omega)}\\
\propto \;& P(\mathbf{R}|\mathbf{S},\mathbf{U},\mathbf{V},\Omega)P(\mathbf{W}|\mathbf{U},\Omega)P(\mathbf{C}|\mathbf{V},\Omega)P(\mathbf{F}|\mathbf{S},\Omega)P(\mathbf{S}|\Omega)P(\mathbf{U}|\Omega)P(\mathbf{V}|\Omega)\\
= \;& \prod_{i,j}\mathcal{N}(R_{ij}|\mathbf{S}_i\mathbf{G}_j^\top\odot\mathbf{U}_i^\top\mathbf{V}_j,\sigma_R^2)\prod_{p,q}\mathcal{N}(W_{pq}|\mathbf{U}_p^\top\mathbf{U}_q,\sigma_W^2)\prod_{m,n}\mathcal{N}(C_{mn}|\mathbf{V}_m^\top\mathbf{V}_n,\sigma_C^2)\\
& \prod_{s,t}\mathcal{N}(F_{st}|S_{st},\sigma_F^2)\prod_{x}\mathcal{N}(\mathbf{S}_x|0,\sigma_S^2)\prod_{y}\mathcal{N}(\mathbf{U}_y|0,\sigma_U^2)\prod_{z}\mathcal{N}(\mathbf{V}_z|0,\sigma_V^2)
\end{aligned} \tag{3-8}$$

其中 Ω 表示特征空间向量和观测矩阵中平均值为 0 的球状高斯分布。那么

$$\begin{aligned}
& \ln P(\mathbf{S},\mathbf{U},\mathbf{V}|\mathbf{R},\mathbf{G},\mathbf{M},\mathbf{C},\mathbf{F},\Omega)\\
\propto \;& -\frac{1}{2\sigma_R^2}\sum_{i,j}(R_{ij}-\mathbf{S}_i\mathbf{G}_j^\top\odot\mathbf{U}_i^\top\mathbf{V}_j)^2-\frac{1}{2\sigma_W^2}\sum_{p,q}(W_{pq}-\mathbf{U}_p^\top\mathbf{U}_q)^2-\frac{1}{2\sigma_C^2}\sum_{m,n}(C_{mn}-\mathbf{V}_m^\top\mathbf{V}_n)^2\\
& -\frac{1}{2\sigma_F^2}\sum_{s,t}(F_{st}-S_{st})^2-\frac{1}{2\sigma_S^2}\sum_{x}\mathbf{S}_x^\top\mathbf{S}_x-\frac{1}{2\sigma_U^2}\sum_{y}\mathbf{U}_y^\top\mathbf{U}_y-\frac{1}{2\sigma_V^2}\sum_{z}\mathbf{V}_z^\top\mathbf{V}_z
\end{aligned} \tag{3-9}$$

最大化这一后验分布等价于最小化混合平方正则项构成的均方误差函数：

$$\begin{aligned}
\mathcal{J} \;=\; & \|\mathbf{R}-\mathbf{S}\mathbf{G}^\top\odot\mathbf{U}^\top\mathbf{V}\|_F^2+\alpha\|\mathbf{W}-\mathbf{U}^\top\mathbf{U}\|_F^2\\
& +\beta\|\mathbf{C}-\mathbf{V}^\top\mathbf{V}\|_F^2+\gamma\|\mathbf{F}-\mathbf{S}\|_F^2+\delta\|\mathbf{S}\|_F^2+\eta\|\mathbf{U}\|_F^2+\lambda\|\mathbf{V}\|_F^2
\end{aligned} \tag{3-10}$$

其中 $\alpha=\frac{\sigma_R^2}{\sigma_W^2}$，$\beta=\frac{\sigma_R^2}{\sigma_C^2}$，$\gamma=\frac{\sigma_R^2}{\sigma_F^2}$，$\delta=\frac{\sigma_R^2}{\sigma_S^2}$，$\eta=\frac{\sigma_R^2}{\sigma_U^2}$，$\lambda=\frac{\sigma_R^2}{\sigma_V^2}$，并且 $\|.\|_F$ 是 Frobenius 范数。采用梯度下降法来解得最佳条件。用随机值初始化 $\mathbf{S}$，$\mathbf{U}$ 和 $\mathbf{V}$，固定另外两个参数，对每一个参数逐步求取最优值直到收敛。很明显地，目标函数的下限是 0，并且梯度下降算法单调降低目标函数值，所以，这一算法一定会最终收敛。本文中使用的梯度下降算法中，目标函数中每一个变量的梯度为

$$\frac{\partial\mathcal{J}}{\partial\mathbf{S}} = -2(\mathbf{R}-\mathbf{S}\mathbf{G}^\top\odot\mathbf{U}^\top\mathbf{V})\mathbf{G}-2\gamma(\mathbf{F}-\mathbf{S})+2\delta\mathbf{S} \tag{3-11}$$

$$\frac{\partial\mathcal{J}}{\partial\mathbf{U}} = -2\mathbf{V}(\mathbf{R}-\mathbf{S}\mathbf{G}^\top\odot\mathbf{U}^\top\mathbf{V})^\top-4\alpha\mathbf{U}(\mathbf{W}-\mathbf{U}^\top\mathbf{U})+2\eta\mathbf{U} \tag{3-12}$$

$$\frac{\partial \mathcal{J}}{\partial \mathbf{V}} = -2\mathbf{U}(\mathbf{R} - \mathbf{S}\mathbf{G}^\top \odot \mathbf{U}^\top\mathbf{V}) - 4\beta\mathbf{V}(\mathbf{C} - \mathbf{V}^\top\mathbf{V}) + 2\lambda\mathbf{V} \tag{3-13}$$

由此，算法1中展示了基于梯度下降的社交上下文模型。在每一个迭代过程中，目标函数值 $\mathcal{J}$ 沿着梯度方向下降最快，那么目标函数值序列 $(\mathcal{J}^{(t)})$ 收敛到最小值。

Algorithm 1 社交上下文推荐模型 ContextMF

Require: $0 < \alpha_S^{(t)}, \alpha_U^{(t)}, \alpha_V^{(t)} < 1, t = 0$. 初始化 $\mathcal{J}^{(0)} = \mathcal{J}(\mathbf{S}^{(0)}, \mathbf{U}^{(0)}, \mathbf{V}^{(0)})$.

Ensure: $\mathcal{J}^{(0)} \geq 0, \mathcal{J}^{(t+1)} < \mathcal{J}^{(t)}$

for $t = 1, 2, \cdots$ **do**

计算 $\frac{\partial \mathcal{J}}{\partial \mathbf{S}}^{(t-1)}, \frac{\partial \mathcal{J}}{\partial \mathbf{U}}^{(t-1)}, \frac{\partial \mathcal{J}}{\partial \mathbf{V}}^{(t-1)}$

$\mathbf{S}^{(t)} \leftarrow \mathbf{S}^{(t-1)} - \alpha_S^{(t-1)} \cdot \frac{\partial \mathcal{J}}{\partial \mathbf{S}}^{(t-1)}$

$\mathbf{U}^{(t)} \leftarrow \mathbf{U}^{(t-1)} - \alpha_U^{(t-1)} \cdot \frac{\partial \mathcal{J}}{\partial \mathbf{U}}^{(t-1)}$

$\mathbf{V}^{(t)} \leftarrow \mathbf{V}^{(t-1)} - \alpha_V^{(t-1)} \cdot \frac{\partial \mathcal{J}}{\partial \mathbf{V}}^{(t-1)}$

$\mathcal{J}^{(t)} \leftarrow \mathcal{J}(\mathbf{S}^{(t)}, \mathbf{U}^{(t)}, \mathbf{V}^{(t)})$

end for

对于真实在线系统来说，把社交上下文推荐模型拓展为可以处理增量数据的模型是势在必行的。但这需要回答这样两个问题：一是该如何通过社交关系和历史行为来把信息推荐给新用户，二是该如何通过信息内容和历史数据来给用户推荐新信息。本工作中通过解决上述问题，开发处理在线增量数据的社交上下文推荐模型Δ ContextMF。该模型能够学习增量数据与离线值 **U**，**V** 和 **S** 的关系来更新影响力矩阵和用户、信息的特征矩阵。

首先是为 ΔM 个新用户推荐信息。假设知道 M 个用户和 ΔM 个新用户之间的交互和关系，旨在学习影响力矩阵的增量 $\Delta\mathbf{S} \in \mathbb{R}^{\Delta M \times M}$ 和新用户特征矩阵的增量 $\Delta\mathbf{U} \in \mathbb{R}^{k \times \Delta M}$。给定增量交互矩阵为 $\Delta\mathbf{F} \in \mathbb{R}^{\Delta M \times M}$，增量用户相似度矩阵为 $\Delta\mathbf{W} \in \mathbb{R}^{\Delta M \times M}$，可以得到增量相关的目标函数 $\mathcal{J}_{\Delta S}$ 和 $\mathcal{J}_{\Delta U}$，以及梯度值 $\Delta\mathbf{S}$ 和 $\Delta\mathbf{U}$。在如下的目标函数中忽略过小的高维度项。

$$\mathcal{J}_{\Delta S} = \|\Delta\mathbf{F} - \Delta\mathbf{S}\|_F^2 \quad ; \quad \frac{\partial \mathcal{J}}{\partial \Delta\mathbf{S}} = -2\Delta\mathbf{F} + \mathcal{O}(\Delta\mathbf{S}) \tag{3-14}$$

$$\mathcal{J}_{\Delta U} = \|\Delta\mathbf{W} - \Delta\mathbf{U}^\top\mathbf{U}\|_F^2 \quad ; \quad \frac{\partial \mathcal{J}}{\partial \Delta\mathbf{U}} = -2\mathbf{U}\Delta\mathbf{W}^\top + \mathcal{O}(\Delta\mathbf{U}) \tag{3-15}$$

所以，采纳信息行为的预测矩阵 $\Delta\mathbf{R} \in \mathbb{R}^{M \times \Delta N}$ 可以计算为 $\Delta\mathbf{R} = \Delta\mathbf{S}\mathbf{G}^\top \odot \Delta\mathbf{U}^\top\mathbf{V}$。

第二是要推荐 ΔN 个新信息给用户。定义增量信息发送者矩阵为 $\Delta\mathbf{G} \in \mathbb{R}^{\Delta N \times M}$，增量信息间相似度矩阵为 $\Delta\mathbf{C} \in \mathbb{R}^{\Delta N \times N}$，也就是通过信息内容的话题分布计算信息之间话题层面的相似度。那么得出目标函数 $\mathcal{J}_{\Delta V}$ 以及相关梯度来计算

信息特征矩阵的增量 $\Delta\mathbf{V} \in \mathbb{R}^{k\times\Delta N}$。

$$\mathcal{J}_{\Delta V} = \|\Delta\mathbf{C} - \Delta\mathbf{V}^{\top}\mathbf{V}\|_F^2 \quad ; \quad \frac{\partial\mathcal{J}}{\partial\Delta\mathbf{V}} = -2\mathbf{V}\Delta\mathbf{C}^{\top} + O(\Delta\mathbf{V}) \tag{3-16}$$

所以采纳信息行为的预测矩阵 $\Delta\mathbf{R} \in \mathbb{R}^{M\times\Delta N}$ 可以计算为 $\Delta\mathbf{R} = \mathbf{S}\Delta\mathbf{G}^{\top} \odot \mathbf{U}^{\top}\Delta\mathbf{V}$。

表 3.1　算法复杂度对比（假定 $M \gg \Delta M$，$N \gg \Delta N$）

	在线增量处理Δ ContextMF	离线处理 ContextMF
ΔM 个新用户	$O(k^2L\Delta MM)$	$O(k^2LM(M+N))$
ΔN 个新信息	$O(k^2L\Delta NN)$	$O(k^2LN(M+N))$

离线推荐模型 ContextMF 是将新用户和新信息融合进历史数据中。如果要推荐信息给新的 ΔM 用户，ContextMF 每次迭代需要 $O(k^2(M+\Delta M)^2)$ 的时间（复杂度）来更新矩阵 $\mathbf{S}$ 和 $\mathbf{U}$，还需要 $O(k^2(M+\Delta M)N)$ 的时间来更新矩阵 $\mathbf{V}$。然而，Δ ContextMF 只需要 $O(k^2\Delta M(M+\Delta M))$ 的时间来计算矩阵 $\Delta\mathbf{S}$ 和 $\Delta\mathbf{U}$。处理新信息的推荐方法也是相似的。所以如表3.1所示，Δ ContextMF 不仅在时间复杂度也在内存节省量上要离线算法 ContextMF 出色，其中 L 是算法的迭代次数。增量处理算法的复杂度正比于用户和信息的数量，而传统算法往往需要平方的时间复杂度[17,21]。在后续性能评测中会展示增量处理模型Δ ContextMF 的推荐效果，证实算法可以用于处理在线增量数据。

3.2　基于时空上下文的行为模式发现方法

本节介绍社交媒体用户行为的时空上下文关联性，并给出行为模式的发现方法。内容包括引言、相关工作、分析行为模式的时空上下文关联性，并给出基于时空上下文的进化分析方法。

3.2.1　本节引言

科学家们从文化、政治乃至心理等各种角度研究人类行为，尝试寻找在个体行为和社交行为中统一的模式，并给予解释。人类行为其实是多种相关联因素交织形成的这一观点已经被广泛接受。诸如物理环境、社交互动、社会身份、个体性格和利益等多面性因素影响了用户行为是否发生。例如，当研究人员改换研究机构的时候，他会开始与新的合作人员展开科研工作，加入新的工程项目，乃至研究新的问题。由于多面性因素的影响，用户行为的方式非常复杂，所以很难精确地总结出这些因素是什么，又是怎么交互在一起的。例外，有心理学研究证实人类的行为会随着内部因素（个性等）和外部因素（环境等）的改变而变化，最

终形成时域上的不同的动态行为特征[60]。例如说，在上世纪 90 年代早期，许多研究者在研究数据库系统和查询处理。在 90 年代晚期，随着多种数据收集方法涌现，未标注的数据量在增加，研究者开始研究非监督的聚类算法和模式挖掘问题。在本世纪初，随着 Facebook 和 Twitter 变得越来越火爆，研究者们开始研究社交网络和社区发现的问题。由此可见，人类行为模式随着不同的地点、不同的时间、不同的环境而发生变化。这种复杂的、动态的特点给理解和预测人类行为带来很大的挑战，缺乏足够的研究工作来利用行为的多面性和动态性，即时空上下文信息，实现行为建模。

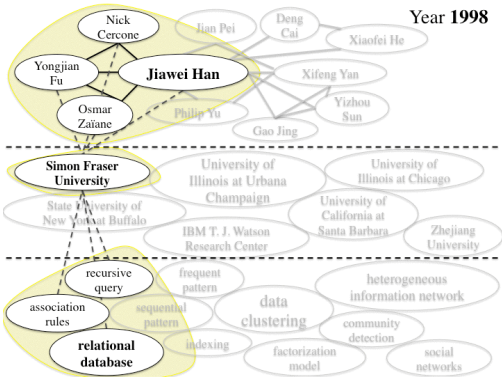
一些传统的数据分析方法已经被用于发现人类行为模式。研究者提出用三维分析法来处理包含用户、查询和网页的点击数据[94]，或构建三维分解模型来为用户、音乐标签和音乐的表现音乐行为建模[144]。然而这种静态的人类行为分析方法并不能够学习行为的动态性信息，捕捉动态行为特征。有一些工作是用时间序列的模型来表示和预测网页搜索行为和内容变化[75]。也有用会话节点来捕捉在标记论文行为时的短期动态兴趣[67]。然而，这种表示方法并不能够很好地处理多面性信息，或是充分描述人类行为的空间复杂特征。目前缺乏多面动态行为模式方面的研究，如何准确地预测人类行为还需要进一步分析。

在从多面性和动态性信息中学习人类行为的过程中存在两个主要的挑战：

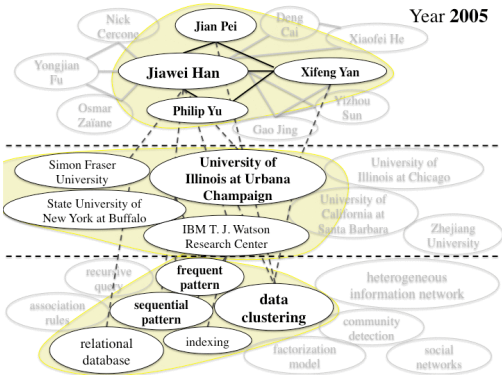
- 高稀疏度。真实系统中的多面性数据是非常稀疏的，例如，在研究者 -研究机构 -科研问题的数据里，研究者并不会在太多的研究机构工作，也不会研究太多的科研问题。更重要的是，如果把时序信息也放进来，多面性的行为信息会更加稀疏。
- 高复杂性。新的多面性人类行为持续不断地产生，形成了动态的行为特征。持续不断产生的数据有着高量、高维度和高稀疏度的问题，给建模、分析带来高计算复杂度的问题。动态张量分析的方法被提出来缩小张量分解时的存储空间和缩短更新分解矩阵的时间[101]。但是这还是需要很多时间来重新计算特征值和特征向量。在为人类行为建模和做预测时，快速处理增量数据的问题依旧非常严峻。

为了解决上述问题，本工作提出了基于张量分解的动态框架得到的“灵活多面动态分析方法”（Flexible Evolutionary Multi-faceted Analysis，简称 FEMA）来预测动态多面的人类行为，从中挖掘时空上下文关联的行为模式。FEMA 采用了灵活的正则项来缓解数据的高稀疏度问题。为了快速处理高维度的张量序列，提出了用稀疏的增量数据快速更新张量的近似算法，从理论上证实了近似算法损失的上界。工作中在两个真实数据集上评测 FEMA 算法，一是从微软学术搜索的发表文章数据，二是从中国的 Twitter 型社交网络腾讯微博得到的微博数据。FEMA

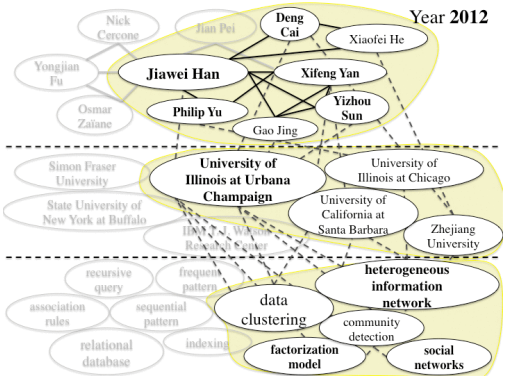
使用多面信息能给出相对 30.8% 的准确度提升，而使用灵活正则项能够再次给出 17.4% 的准确度提升。另外，FEMA 可以把运行时间从小时级降低到分钟级，这对于实时系统来说是有用的。



(a) 当 Prof. Han 在西蒙弗雷泽大学研究数据库系统



(b) 当 Prof. Han 在 UIUC 研究模式提取和聚类方法



(c) 当 Prof. Han 在 UIUC 研究社交网络和信息网络的数据挖掘

图 3.5　多面动态行为的模式发现：韩家炜教授（Prof. Han）和他的研究组是如何改变着学术研究方向？科研行为包括了用户、研究机构和关键词三层，并且随着时间变化。

图3.5 中展示了 FEMA 挖掘出的学术科研行为的动态模式。可以看到韩家炜教授和他的科研小组在研究数据库系统、聚类算法到社交网络和信息网络的变化。这样的模式很清楚地展现了科研机构的变化和韩教授论文合作者的关系形成了发

表论文的多面动态行为，可以对科研行为的多面性和动态性有更好的认识和理解。本章的主要贡献点如下：

- 由人类行为心理学研究得到启发，提出如何融合时间维度和多面分析方法预测时空上下文关联的行为和模式挖掘。这对于如网页搜索、推荐系统等应用都有重大意义。
- 基于张量分解模型提出了 FEMA 方法来预测动态多面行为。这个模型融入了灵活的正则项来削减数据高稀疏度的问题，还给出近似算法来快速处理人类行为的动态增量。
- FEMA 算法快速有效，并给出了理论的证明：这个算法相比于已经存在的多面张量算法的立方级复杂度，FEMA 算法是接近平方级的。
- 在学术科研和社交网络数据集上进行充分实验来预测人类行为。实验结果表明 FEMA 算法能够在两个数据集上都比其他方法要好。更重要的是还证实了近似算法会降低计算时间，同时准确度损失是很小的。

3.2.2　相关工作

相关的研究工作分为三类：行为建模，行为动力学和张量分解模型。矩阵分解模型被广泛应用在人类行为建模和预测应用中。这些模型往往具有两种元素，比如用户 -项目（电影或音乐）的评价行为[11]，药品 -病人的作用[80]。元素个数超过两个的情况下，研究者们用高维度张量模型实现行为建模，例如网页搜索[94]，图片和书籍的标签[102,105]，和推荐系统[40,144,146]。这些工作从静态角度分析了人类行为模式，但是并不能够捕捉行为的动态性。

还有一些研究工作是用时间信息来理解不同应用中的过去用户行为来预测未来的行为。应用包括用户的社区发现[63]，推荐系统[49,73]，研究主题发现[77]，语义图模型[76,96] 和在线媒体的话题演变[40,45,54,78]。研究者把用户兴趣分为长期和短期，并利用时间因素来描述它们的不同[67]。和这个工作不同，本文关注于用户群体的动态性行为，而不是单个用户的行为演变。此外，研究者选择依赖于时间的合适模型来学习用户行为模式[75]。利用时间信息合理发现人类行为的潜在动态机制是重要的、有挑战性的研究工作。

研究人员已经开发出很多相关的模型方法，包括矩阵分解模型[89,90]，张量分析方法[89,95,101,109]，张量分解模型[68,91,103,280] 以及可扩展方法[100,106,108]。本工作在方法层面关注如何用矩阵和张量理论在张量分解模型中高效处理增量数据[87]。

3.2.3　行为模式的时空上下文关联性分析

本节给出两种不同类型的人类行为，包括 “学术研究发表论文” 的行为和 “在微博中提及某人” 的行为，接着给出要研究的问题定义和任务。

学术研究数据往往会关注发掘学术行为的动态模式。这里把数据集表示为一组值对 (a, f, k, t)，用来表示在时间 t（$t = 1, \ldots, T$），研究人员 a 在研究机构 f（大学，研究中心等）发表了一篇关于一个关键字 k 的论文。把数据模型化为三维张量序列 $\mathcal{X}_\mathbf{t} \in \mathbb{R}^{n^{(a)} \times n^{(f)} \times n^{(k)}}$，其中 $n^{(a)}$ 是研究人员的数量，$n^{(f)}$ 是研究机构的数量，和 $n^{(k)}$ 是关键字的数量。$\mathcal{X}_\mathbf{t}(a, f, k)$ 是所有的值对 (a, f, k, t') 的数量（$t' \leq t$）。目标是把张量序列分解为

$$\mathcal{X}_\mathbf{t} \approx \mathcal{Y}_\mathbf{t} \times_{(a)} \mathbf{A_t} \times_{(f)} \mathbf{F_t} \times_{(k)} \mathbf{K_t} \tag{3-17}$$

其中

- $\mathcal{Y}_\mathbf{t} \in \mathbb{R}^{r^{(a)} \times r^{(f)} \times r^{(k)}}$ 是核张量序列，编码动态行为模式，描述研究人员、研究机构和关键字组之间的关系。$\mathcal{Y}_\mathbf{t}(j^{(a)}, j^{(f)}, j^{(k)})$ 表示在时间 t 之前来自第 $j^{(a)}$ 组研究人员，在第 $j^{(f)}$ 组研究机构，发表第 $j^{(k)}$ 组关键字的行为发生的概率。
- $\mathbf{A_t} \in \mathbb{R}^{n^{(a)} \times r^{(a)}}$ 是研究人员在时间 t 之前的映射矩阵。$\mathbf{A}(i^{(a)}, j^{(a)})$ 表示了第 $i^{(a)}$ 个研究人员归属于第 $j^{(a)}$ 个研究人员组的概率。
- $\mathbf{F_t} \in \mathbb{R}^{n^{(f)} \times r^{(f)}}$ 是研究机构在时间 t 之前的映射矩阵。$\mathbf{F_t}(i^{(f)}, j^{(f)})$ 表示了第 $i^{(f)}$ 个研究机构归属于第 $j^{(f)}$ 个研究机构组的概率。
- $\mathbf{K_t} \in \mathbb{R}^{n^{(k)} \times r^{(k)}}$ 是关键字在时间 t 之前的映射矩阵。$\mathbf{K_t}(i^{(k)}, j^{(k)})$ 表示了第 $i^{(k)}$ 个关键字归属于第 $j^{(k)}$ 个关键字组的概率。

要解决张量分解模型中的稀疏度问题的关键是从研究人员的合作关系、研究机构的地理位置或是关键字的语义信息中学习辅助知识，用灵活正则项表示。正则项可以被描述成拉普拉斯矩阵 $\mathbf{L^{(a)}}$，$\mathbf{L^{(f)}}$ 和 $\mathbf{L^{(k)}}$，其中第 (i, j) 个矩阵元素表示第 i 个和第 j 个实体（研究人员、研究机构和关键字）之间的相似度。这样的数值可以用研究人员合作论文的数量或是研究机构的地理距离来表示。现在问题是给定张量序列和限制项，如何分解出核张量序列和映射矩阵。要注意的是相对于张量的大规模来比，相邻时间里的改变是非常小的。用 $\Delta\mathcal{X}_\mathbf{t}$ 来定义在时刻 t 的稀疏增量，也就是对于 $1 \leq t < T$，$\Delta\mathcal{X}_\mathbf{t} = \mathcal{X}_{\mathbf{t+1}} - \mathcal{X}_\mathbf{t}$。于是问题可以被转化为两步：

- 给定第一个张量 $\mathcal{X}_\mathbf{1}$ 和限制项 $\mathbf{L^{(a)}}$，$\mathbf{L^{(f)}}$ 和 $\mathbf{L^{(k)}}$，找到映射矩阵 $\mathbf{A_1}$，$\mathbf{F_1}$ 和 $\mathbf{K_1}$ 以及第一个核张量 $\mathcal{Y}_\mathbf{1}$。
- 在时刻 t $(1 \leq t < T)$ 给定张量 $\mathcal{X}_\mathbf{t}$，张量增量 $\Delta\mathcal{X}_\mathbf{t}$，原先的映射矩阵 $\mathbf{A_t}$，$\mathbf{F_t}$ 和 $\mathbf{K_t}$，以及限制项 $\mathbf{L^{(a)}}$，$\mathbf{L^{(f)}}$ 和 $\mathbf{L^{(k)}}$，找到新的映射矩阵 $\mathbf{A_{t+1}}$，$\mathbf{F_{t+1}}$ 和 $\mathbf{K_{t+1}}$ 以

及新的核张量 $\mathcal{Y}_{\mathbf{t+1}}$.

接下来介绍微博数据集如何建模来捕捉微博中提及行为的动态模式。数据集可以用一系列的值对 (s,d,w,t) 来表示一个 Twitter 用户 s（源用户）在时间 t 的微博内容里用 "@d" 的格式来提及某个用户 d (目标用户)，微博中包含词汇 w $(t=1,\ldots,T)$，于是用户 d 会在 "被提及" 的功能里看到这条消息。和给学术研究的行为建模相似，用三维张量序列来为数据建模：$\mathcal{X}_{\mathbf{t}} \in \mathbb{R}^{n^{(s)}\times n^{(d)}\times n^{(w)}}$，其中 $n^{(s)}$ 是源用户的数量，$n^{(d)}$ 是目标用户的数量，$n^{(w)}$ 是词汇数量。$\mathcal{X}_{\mathbf{t}}(s,d,w)$ 描述了值对 (s,d,w,t') 的数量（$t' \le t$）。这里任务是分解张量：

$$\mathcal{X}_{\mathbf{t}} \approx \mathcal{Y}_{\mathbf{t}} \times_{(s)} \mathbf{S_t} \times_{(d)} \mathbf{D_t} \times_{(w)} \mathbf{W_t} \tag{3-18}$$

其中 $\mathcal{Y}_{\mathbf{t}} \in \mathbb{R}^{r^{(s)}\times r^{(d)}\times r^{(w)}}$ 是核张量序列；$\mathbf{S_t} \in \mathbb{R}^{n^{(s)}\times r^{(s)}}$ 是源用户的映射矩阵；$\mathbf{D_t} \in \mathbb{R}^{n^{(d)}\times r^{(d)}}$ 是目标用户的映射矩阵；$\mathbf{W_t} \in \mathbb{R}^{n^{(w)}\times r^{(w)}}$ 是微博词汇的映射矩阵。

要解决数据稀疏度的问题，可以用社交媒体用户之间的社交关系（共同好友数量）和词汇的语义信息来编码成拉普拉斯矩阵 $\mathbf{L^{(s)}}$，$\mathbf{L^{(d)}}$，$\mathbf{L^{(w)}}$ 并作为灵活正则项。研究问题可以总结为以下两步：

- 给定第一个张量 $\mathcal{X}_1$，和限制项 $\mathbf{L^{(s)}}$，$\mathbf{L^{(d)}}$ 和 $\mathbf{L^{(w)}}$，找到映射矩阵 $\mathbf{S_1}$，$\mathbf{D_1}$ 和 $\mathbf{W_1}$ 以及第一个核心张量 $\mathcal{Y}_1$.
- 在时刻 t（$1 \le t < T$），给定张量 $\mathcal{X}_{\mathbf{t}}$，张量增量 $\Delta\mathcal{X}_{\mathbf{t}}$，存在的映射矩阵 $\mathbf{S_t}$，$\mathbf{D_t}$ 和 $\mathbf{W_t}$，以及限制项 $\mathbf{L^{(s)}}$，$\mathbf{L^{(d)}}$ 和 $\mathbf{L^{(w)}}$，找到新的映射矩阵 $\mathbf{S_{t+1}}$，$\mathbf{D_{t+1}}$ 和 $\mathbf{W_{t+1}}$ 以及新的核张量 $\mathcal{Y}_{\mathbf{t+1}}$。

上述问题和过去的模式挖掘算法不同。首先，要融合多面性信息和限制项到统一的框架中；第二，与原有工作中分解单一张量不同的是，要用动态分析方法高效地处理稀疏增量。如果扩展问题描述从 3 维到 M 维，给出通用定义。

定义 3.1 (灵活多面性动态分析)：(1) **初始化:** 给定第一个 M 维度的张量 $\mathcal{X}_1 \in \mathbb{R}^{n^{(1)}\times\ldots\times n^{(M)}}$，和限制项 $\mathbf{L^{(m)}}|_{m=1}^{M} \in \mathbb{R}^{n^{(m)}\times n^{(m)}}$，找到第一个映射矩阵 $\mathbf{A_1^{(m)}}|_{m=1}^{M} \in \mathbb{R}^{n^{(m)}\times r^{(m)}}$ 找到第一个核张量 $\mathcal{Y}_1 \in \mathbb{R}^{r^{(1)}\times\ldots\times r^{(M)}}$。(2) **动态分析：** 在时刻 t（$1 \le t < T$），给定张量 $\mathcal{X}_{\mathbf{t}} \in \mathbb{R}^{n^{(1)}\times\ldots\times n^{(M)}}$，和增量张量 $\Delta\mathcal{X}_{\mathbf{t}}$，已经算出的映射矩阵 $\mathbf{A_t^{(m)}}|_{m=1}^{M}$，和限制项 $\mathbf{L^{(m)}}|_{m=1}^{M}$，找到新的映射矩阵 $\mathbf{A_{t+1}^{(m)}}|_{m=1}^{M}$ 和新的核张量 $\mathcal{Y}_{\mathbf{t+1}}$.

3.2.4　基于时空上下文的进化分析方法

本节根据灵活多面性动态分析问题的两步骤给出解决方法，讨论运算效率和近似效果。融合多面性信息和限制项到张量分解模型中前，定义 $\mu^{(m)}$ 为第 m 维的

拉普拉斯矩阵 $\mathbf{L}^{(\mathbf{m})}$ 的权重，并定义第 m 维在时刻 $t=1$ 的协方差矩阵为

$$\mathbf{C}_{\mathbf{1}}^{(\mathbf{m})} = \mathbf{X}_{\mathbf{1}}^{(\mathbf{m})}\mathbf{X}_{\mathbf{1}}^{(\mathbf{m})\mathbf{T}} + \mu^{(m)}\mathbf{L}^{(\mathbf{m})} \tag{3-19}$$

其中 $\mathbf{X}_{\mathbf{1}}^{(\mathbf{m})} \in \mathbb{R}^{n^{(m)}\times\prod_{i\neq m} n^{(i)}}$ 是张量 $\mathcal{X}_{\mathbf{1}}$ 在第 m 维的分解矩阵。映射矩阵 $\mathbf{A}_{\mathbf{1}}^{(\mathbf{m})}|_{m=1}^{M}$ 用对角化得到：协方差矩阵 $\mathbf{C}_{\mathbf{1}}^{(\mathbf{m})}|_{m=1}^{M}$ 的前 $r^{(m)}$ 个特征向量。算法2给出了伪代码。

Algorithm 2 FEMA 算法的初始化

Require: $\mathcal{X}_{\mathbf{1}}, \mathbf{L}^{(\mathbf{m})}|_{m=1}^{M}$

  for $m = 1, \ldots, M$ **do**

    计算协方差矩阵 $\mathbf{C}_{\mathbf{1}}^{(\mathbf{m})}$;

    $\lambda_{\mathbf{1}}^{(\mathbf{m})}$、$\mathbf{A}_{\mathbf{1}}^{(\mathbf{m})}$ 是 $\mathbf{C}_{\mathbf{1}}^{(\mathbf{m})}$ 的前 $r^{(m)}$ 个特征值、特征向量

  end for

  $\mathcal{Y}_{\mathbf{1}} = \mathcal{X}_{\mathbf{1}} \prod_{m=1}^{M} \times_{(m)} \mathbf{A}_{\mathbf{1}}^{(\mathbf{m})\mathbf{T}}$;

  return $\mathbf{A}_{\mathbf{1}}^{(\mathbf{m})}|_{m=1}^{M}, \lambda_{\mathbf{1}}^{(\mathbf{m})}|_{m=1}^{M}, \mathcal{Y}_{\mathbf{1}}$

基于张量摄动（tensor perturbation）理论设计根据张量的变化调整映射矩阵的高效算法。定义 $\mathbf{X}_{\mathbf{t}}^{(\mathbf{m})} \in \mathbb{R}^{n^{(m)}\times\prod_{i\neq m} n^{(i)}}$ 为张量 $\mathcal{X}_{\mathbf{t}}$ 在第 m 维度上的矩阵化结果。定义协方差矩阵为 $\mathbf{C}_{\mathbf{t}}^{(\mathbf{m})} = \mathbf{X}_{\mathbf{t}}^{(\mathbf{m})}\mathbf{X}_{\mathbf{t}}^{(\mathbf{m})^\top} + \mu^{(m)}\mathbf{L}^{(\mathbf{m})}$，并定义 $(\lambda_{t,i}^{(m)}, \mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})})$ 为协方差矩阵 $\mathbf{C}_{\mathbf{t}}^{(\mathbf{m})}$ 的一对特征值和特征向量对。其中向量 $\mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})}$ 就是映射矩阵 $\mathbf{A}_{\mathbf{t}}^{(\mathbf{m})}$ 的第 i 列。可以重写 $(\lambda_{t+1,i}^{(m)}, \mathbf{a}_{\mathbf{t+1,i}}^{(\mathbf{m})})$ 为

$$\lambda_{t+1,i}^{(m)} = \lambda_{t,i}^{(m)} + \Delta\lambda_{t,i}^{(m)} \tag{3-20}$$

$$\mathbf{a}_{\mathbf{t+1,i}}^{(\mathbf{m})} = \mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})} + \Delta\mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})} \tag{3-21}$$

简化上述表达，在所有项和公式中忽略 "t"，可得

$$\begin{aligned}&[(\mathbf{X}^{(\mathbf{m})} + \Delta\mathbf{X}^{(\mathbf{m})})(\mathbf{X}^{(\mathbf{m})} + \Delta\mathbf{X}^{(\mathbf{m})})^\top + \mu^{(m)}\mathbf{L}^{(\mathbf{m})}] \cdot (\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}) \\ &= (\lambda_i^{(m)} + \Delta\lambda_i^{(m)})(\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})})\end{aligned} \tag{3-22}$$

问题转化为如何分别动态地改变 $\Delta\lambda_i^{(m)}$ 和 $\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$。由上式可知

$$\begin{aligned}&\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})^\top}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})^\top}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^\top}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^\top}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} \\ &+\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})^\top}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})^\top}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \Delta\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^\top}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} \\ &+\Delta\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^\top}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \mu^{(m)}\mathbf{L}^{(\mathbf{m})}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \mu^{(m)}\mathbf{L}^{(\mathbf{m})}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} \\ &= \lambda_i^{(m)}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \lambda_i^{(m)}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \Delta\lambda_i^{(m)}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})} + \Delta\lambda_i^{(m)}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}\end{aligned} \tag{3-23}$$

近似算法中关注一阶项，忽略高阶摄动项，比如 $\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^\top}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$，$\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})^\top}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$，$\Delta\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^\top}\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$ 和 $\Delta\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^\top}\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$。使用事实

$(\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top + \mu^{(m)}\mathbf{L^{(m)}})\mathbf{a_i^{(m)}} = \lambda_i^{(m)}\mathbf{a_i^{(m)}}$，可得

$$\begin{aligned}&\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} + (\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}} + \mu^{(m)}\mathbf{L^{(m)}}\Delta\mathbf{a_i^{(m)}}\\&= \lambda_i^{(m)}\Delta\mathbf{a_i^{(m)}} + \Delta\lambda_i^{(m)}\mathbf{a_i^{(m)}}\end{aligned} \tag{3-24}$$

在上式两段乘以 $\mathbf{a_i^{(m)}}^\top$ 由于 $\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top$ 和 $\mathbf{L^{(m)}}$ 的对称性得知

$$\Delta\lambda_i^{(m)} = \mathbf{a_i^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}} \tag{3-25}$$

因为每一对特征向量正交，用已有的特征向量推导特征向量在子空间中的变化 $\Delta\mathbf{a_i^{(m)}}$，即

$$\Delta\mathbf{a_i^{(m)}} \approx \sum_{j=1}^{r^{(m)}}\alpha_{ij}\mathbf{a_j^{(m)}} \tag{3-26}$$

其中 $\{\alpha_{ij}\}$ 是要决定的小值常数。代入后得

$$\begin{aligned}&(\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top + \mu^{(m)}\mathbf{L^{(m)}})\sum_{j=1}^{r^{(m)}}\alpha_{ij}\mathbf{a_j^{(m)}} + (\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}}\\&= \lambda_i^{(m)}\sum_{j=1}^{r^{(m)}}\alpha_{ij}\mathbf{a_j^{(m)}} + \Delta\lambda_i^{(m)}\mathbf{a_i^{(m)}}\end{aligned} \tag{3-27}$$

这等价于

$$\begin{aligned}&\sum_{j=1}^{r^{(m)}}\lambda_j^{(m)}\alpha_{ij}\mathbf{a_j^{(m)}} + \mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}}\\&= \lambda_i^{(m)}\sum_{j=1}^{r^{(m)}}\alpha_{ij}\mathbf{a_j^{(m)}} + \Delta\lambda_i^{(m)}\mathbf{a_i^{(m)}}\end{aligned} \tag{3-28}$$

在上述公式两端乘以 $\mathbf{a_k^{(m)}}^\top$ 得到

$$\lambda_k^{(m)}\alpha_{ik} + \mathbf{a_k^{(m)}}^\top\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} + \mathbf{a_k^{(m)}}^\top\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}} = \lambda_i^{(m)}\alpha_{ik} \tag{3-29}$$

所以，

$$\alpha_{ik} = \frac{\mathbf{a_k^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}}}{\lambda_i^{(m)} - \lambda_k^{(m)}} \tag{3-30}$$

为了求得 α_{ii}，可知

$$(\mathbf{a_i^{(m)}} + \Delta\mathbf{a_i^{(m)}})^\top(\mathbf{a_i^{(m)}} + \Delta\mathbf{a_i^{(m)}}) = 1 \tag{3-31}$$

$$\Longleftrightarrow 1 + 2\mathbf{a_i^{(m)}}^\top\Delta\mathbf{a_i^{(m)}} + O(\|\Delta\mathbf{a_i^{(m)}}\|^2) = 1 \tag{3-32}$$

舍去高阶项，可知 $\alpha_{ii}=0$。所以，

$$\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}=\sum_{j\neq i}\frac{\mathbf{a}_{\mathbf{j}}^{(\mathbf{m})^{\top}}(\mathbf{X}^{(\mathbf{m})}\Delta\mathbf{X}^{(\mathbf{m})^{\top}}+\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{X}^{(\mathbf{m})^{\top}})\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}}{\lambda_i^{(m)}-\lambda_j^{(m)}}\mathbf{a}_{\mathbf{j}}^{(\mathbf{m})} \tag{3-33}$$

注意到的是正则项 $\mathbf{L}^{(\mathbf{m})}$ 并不出现在特征值和特征向量的更新函数中，也就是说只需要学习一次复杂的限制项。

Algorithm 3 FEMA 算法的动态分析

Require: $\mathcal{X}_{\mathbf{t}}, \Delta\mathcal{X}_{\mathbf{t}}, \mathbf{A}_{\mathbf{t}}^{(\mathbf{m})}|_{m=1}^{M}, \lambda_{\mathbf{t}}^{(\mathbf{m})}|_{m=1}^{M}$

  for $m=1,\ldots,M$ **do**

    for $i=1,\ldots,r^{(m)}$ **do**

      计算 $\Delta\lambda_{t,i}^{(m)}$，然后计算 $\lambda_{t+1,i}^{(m)}=\lambda_{t,i}^{(m)}+\Delta\lambda_{t,i}^{(m)}$;

      计算 $\Delta\mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})}$，然后计算 $\mathbf{a}_{\mathbf{t+1,i}}^{(\mathbf{m})}=\mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})}+\Delta\mathbf{a}_{\mathbf{t,i}}^{(\mathbf{m})}$ 和 $\mathbf{A}_{\mathbf{t+1}}^{(\mathbf{m})}=\{\mathbf{a}_{\mathbf{t+1,i}}^{(\mathbf{m})}\}$;

    end for

  end for

  $\mathcal{Y}_{\mathbf{t+1}}=(\mathcal{X}_{\mathbf{t}}+\Delta\mathcal{X}_{\mathbf{t}})\prod_{m=1}^{M}\times_{(m)}\mathbf{A}_{\mathbf{t+1}}^{(\mathbf{m})\mathbf{T}}$;

  return $\mathbf{A}_{\mathbf{t+1}}^{(\mathbf{m})}|_{m=1}^{M}, \lambda_{\mathbf{t+1}}^{(\mathbf{m})}|_{m=1}^{M}, \mathcal{Y}_{\mathbf{t+1}}$

接下来分析算法3的计算复杂度。定义 $D^{(m)}$ 为第 m 维的每个点的特征数量。由于张量数据通常极端稀疏，所以 $D^{(m)}\leq E\ll\prod_{m'\neq m}n^{(m')}$，其中 E 是张量里的非负值的数量。为了计算第 m 维的特征值和特征向量的增量，需要计算 $\mathbf{v}_{\mathbf{i}}^{(\mathbf{m})}$，需要时间复杂度 $O(n^{(m)}D^{(m)})$。由于 $\Delta\mathbf{X}^{(\mathbf{m})}$ 是稀疏的，$\Delta\mathbf{X}^{(\mathbf{m})}\mathbf{v}_{\mathbf{i}}^{(\mathbf{m})}$ 只需要常数时间复杂度 $O(D^{(m)})$. 所以，计算 $\Delta\lambda_i^{(m)}$ 和 $\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$ 的时候，需要 $O(r^{(m)}n^{(m)}D^{(m)}+r^{(m)}D^{(m)})$ 的时间，而更新 T 次特征值和特征向量需要 $O(T\sum_{m=1}^{M}r^{(m)}(n^{(m)}+1)D^{(m)})$ 时间。通过与重新计算 $\mathcal{X}_{\mathbf{t+1}}$ 的特征值的比较，消耗 $O(T\sum_{m=1}^{M}(D^{(m)}(n^{(m)})^2+(n^{(m)})^3))$ 时间，那么增量算法需要时间少很多。

下面证明两个定理：$\Delta\lambda_i^{(m)}$ 和 $\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$ 的上界。这两个定理证明了 $\Delta\lambda_i^{(m)}$ 和 $\Delta\mathbf{a}_{\mathbf{i}}^{(\mathbf{m})}$ 与 $\Delta\mathbf{X}^{(\mathbf{m})}$ 的范数相关。由于近似算法舍弃高阶项，当这些项相对较小的时候，FEMA 算法准确性得到保证。

定理 3.1：　特征值的标准差的范数 $|\Delta\lambda_i^{(m)}|, (\forall i=1,\ldots,r^{(m)})$, 满足不等式

$$|\Delta\lambda_i^{(m)}|\leq 2(\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})^{\top}}\mathbf{X}^{(\mathbf{m})}})^{\frac{1}{2}}\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2 \tag{3-34}$$

其中 $\lambda^{\max}_{\mathbf{X}^{(\mathbf{m})^{\top}}\mathbf{X}^{(\mathbf{m})}}$ 是 $\mathbf{X}^{(\mathbf{m})^{\top}}\mathbf{X}^{(\mathbf{m})}$ 的最大特征值。$\|\Delta\mathbf{X}^{(\mathbf{m})}\|_2$ 是矩阵 $\Delta\mathbf{X}^{(\mathbf{m})}$ 的 2-范数。

证明 根据上式可知

$$|\Delta\lambda_i^{(m)}| = |\mathbf{a_i^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}}| \tag{3-35}$$

根据 Cauchy-Schwarz 不等式得到

$$\begin{aligned}&|\mathbf{a_i^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}}|\\&\leq 2\|\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}}\|_2\|\mathbf{a_i^{(m)}}\|_2 = 2\|\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}}\|_2\end{aligned} \tag{3-36}$$

其中首先利用 $\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top$ 的对称性，接着利用 $\|\mathbf{a_i^{(m)}}\| = 1$。根据矩阵 2-范数的定义知道

$$\|\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\|_2 = \sup_{\|\mathbf{w}\|_2=1}\|\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{w}\|_2 \tag{3-37}$$

所以

$$\begin{aligned}&|\mathbf{a_i^{(m)}}^\top(\mathbf{X^{(m)}}\Delta\mathbf{X^{(m)}}^\top + \Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top)\mathbf{a_i^{(m)}}|\\&\leq 2\|\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\|_2 \leq 2\|\mathbf{X^{(m)}}\|_2\|\Delta\mathbf{X^{(m)}}\|_2 = 2(\lambda^{\max}_{\mathbf{X^{(m)}}^\top\mathbf{X^{(m)}}})^{\frac{1}{2}}\|\Delta\mathbf{X^{(m)}}\|_2\end{aligned} \tag{3-38}$$

□

定理 3.2： 特征向量的标准差的范数 $|\Delta\mathbf{a_i^{(m)}}|, (\forall i = 1, \ldots, r^{(m)})$, 满足不等式

$$|\Delta\mathbf{a_i^{(m)}}| \leq 2\|\Delta\mathbf{X^{(m)}}\|_2\sum_{j\neq i}\frac{(\lambda^{\max}_{\mathbf{X^{(m)}}^\top\mathbf{X^{(m)}}})^{\frac{1}{2}}}{|\lambda_i^{(m)} - \lambda_j^{(m)}|} \tag{3-39}$$

其中 $\lambda^{\max}_{\mathbf{X^{(m)}}^\top\mathbf{X^{(m)}}}$ 是 $\mathbf{X^{(m)}}^\top\mathbf{X^{(m)}}$ 的最大的特征值，$\|\Delta\mathbf{X^{(m)}}\|_2$ 是 $\Delta\mathbf{X^{(m)}}$ 的 2-范数。

证明 于是有

$$\begin{aligned}|\Delta\mathbf{a_i^{(m)}}| &= 2\left|\sum_{j\neq i}\frac{\mathbf{a_j^{(m)}}^\top\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}}}{\lambda_i^{(m)} - \lambda_j^{(m)}}\mathbf{a_j^{(m)}}\right| \leq 2\sum_{j\neq i}\left\|\frac{\mathbf{a_j^{(m)}}^\top\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}}}{\lambda_i^{(m)} - \lambda_j^{(m)}}\mathbf{a_j^{(m)}}\right\|\\&\leq 2\sum_{j\neq i}\frac{\|\mathbf{a_j^{(m)}}\|}{|\lambda_i^{(m)} - \lambda_j^{(m)}|}\|\mathbf{a_j^{(m)}}^\top\Delta\mathbf{X^{(m)}}\mathbf{X^{(m)}}^\top\mathbf{a_i^{(m)}}\|\\&\leq 2\|\Delta\mathbf{X^{(m)}}\|_2\sum_{j\neq i}\frac{(\lambda^{\max}_{\mathbf{X^{(m)}}^\top\mathbf{X^{(m)}}})^{\frac{1}{2}}}{|\lambda_i^{(m)} - \lambda_j^{(m)}|}\end{aligned} \tag{3-40}$$

□

3.3　性能评测

本节从以下两个方面评测本章所提出工作的性能：一是社交媒体采纳信息行为的预测效果，二是时空环境下行为预测效果和模式发现效果。

3.3.1　社交媒体中采纳信息行为预测性能

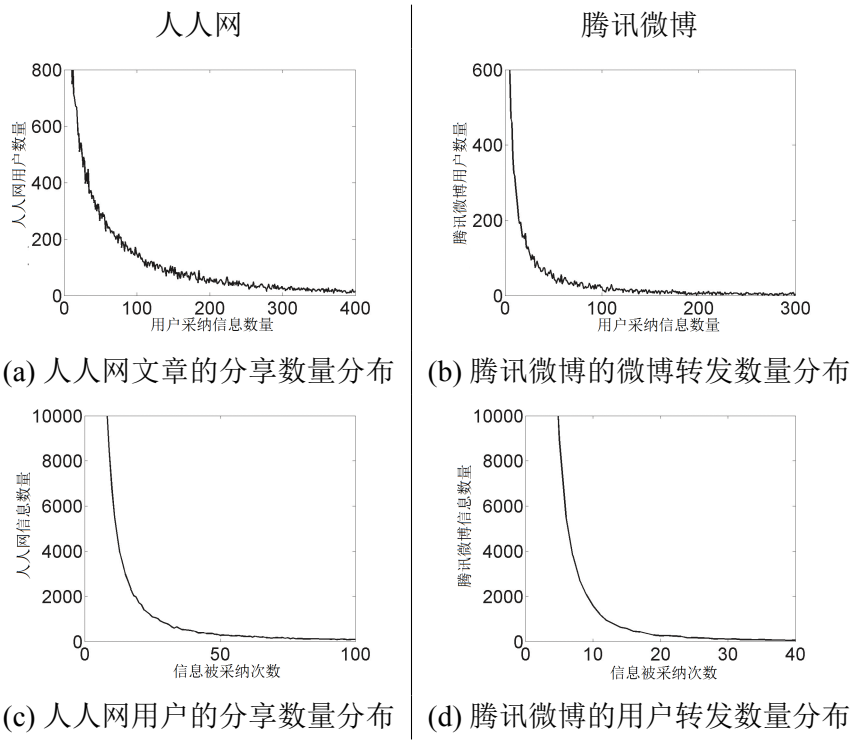
本节将给出基于人人网和腾讯微博两大社交媒体数据的实验结果，验证 ContextMF 的有效性和高效性。表3.2总结了人人网和腾讯微博的实验数据特性。人人网数据集的密度为 0.59%，腾讯微博的密度为 0.09%。数据的高稀疏度问题在社交媒体中极其严重。工作中从一个典型的允许用户发布个人信息、增加好友的社交网络人人网中爬取数据。人人网用户一个常用操作就是分享博客文章、照片和视频的超链接信息（本文中统称为信息）。如果一个用户分享了某个信息，该行为就会作为新鲜事实时发布在他好友的时间线上。抓取了接近 100 万人人网用户在 2007 年 2 月到 2009 年 12 月之间的社交关系和分享行为。同时从允许用户互相关注和获取信息的腾讯微博上也爬取了数据。类似 Twitter，腾讯微博允许用户通过转发微博来传播信息。在 2011 年 1 月抓取了超过 16 万用户的原创微博、转发微博和用户之间的关注关系。

表 3.2　社交媒体数据集概况

	人人网数据	腾讯微博数据
用户数量 (M)	939,363	163,661
信息（文章、微博）数量 (N)	1,625,689	529,615
采纳信息的行为数量	5,829,368	1,566,609

观察人人网和腾讯微博数据的统计特征来进一步证明数据的高稀疏性。表3.6(a) 和 (b) 绘制了消息的分享或是转发数量分布，也就是为用户 u_j 计算 $\sum_i G_{ij}$。表3.6(c) 和 (d) 绘制了用户分享或是转发消息数量的分布，也就是为信息 p_i 计算 $\sum_j G_{ij}$。可以看到这四个表格都存在着长尾分布现象，反映出社交媒体中大部分用户的采纳信息行为都非常稀疏。

工作中参照两种经典的推荐任务来设计实验[7]，一是预测用户的采纳信息与否的行为，二是对用户收到的信息按照采纳概率排序。第一个任务要求推荐系统在给定特定的用户和特定的信息的情况下预测是否会发生采纳行为。因此一种合适的评价指标是预测错误率（越小越好）。第二个任务关注实际推荐效果，即给出对用户收到的信息排序后的列表，同样要能预测用户采纳所收到信息的概率。用不同的排序式评测标准来评价算法有多成功地把用户最喜欢的信息放到推荐列表



(a) 人人网文章的分享数量分布　(b) 腾讯微博的微博转发数量分布

(c) 人人网用户的分享数量分布　(d) 腾讯微博的用户转发数量分布

图 3.6　人人网和腾讯微博的长尾数据分布，包括 (a-b) 消息采纳数量的分布和 (c-d) 用户采纳信息数量的分布。

的前头（越高越好）。

推荐社交信息（文章、微博等）不同于传统的不含时间信息的数据集，实验要考虑时间设置。首先因为用户会在不同的上下文情况下对于同样的信息做出不同的决定，也就是说，一个用户在时间 t_1 收到一则消息后做出的采纳还是拒绝的决定，与其在时间 t_2 收到该信息后做出的决定是可能不同的，因为他的好友或者是所关注的人会在时间间隔 $\Delta t = t_2 - t_1$ 里分享或者转发这个信息。第二，实验设置要求既有正向的用户采纳信息的数据，又有负向的用户拒绝信息的数据。根据用户采纳行为记录很容易得到正向数据，但负向数据是很难直接获取的。但可知有两个情况下是用户不会采纳某条信息：一是用户不在线，也就不会看到这条消息；二是用户看过消息，但是拒绝采纳消息。只有后面这一类的不采纳行为可以被看做负向数据。根据上面的原因可以设计 “在线会话” 来描述用户在线使用社交媒体。假设在线会话中会阅读所有来自他社交关系发来的信息。给定一个用户，一个合理的在线会话会有这么三个特性。首先，一个会话的长度应该在 Δt_{max}（默认为 5 分钟）以内。第二，会话中用户会收到来自好友的至少 n_{min}（默认为 15）条消息。第三，会话中用户采纳（转发或者分享）不止 2 条信息。这里称 Δt_{max} 和 n_{min} 为在线会话参数。那么在定义的在线会话中，用户会在很短的时间内收到一组以两个正向采纳行为作为头和尾的信息。图3.7中展现了合理的和不合理的在线

会话之间的区别，以及如何用在线会话来构造测试数据集的。在这些测试项上比较所有的基线算法以及提出的 ContextMF 算法。虽然在线会话的设计并不能保证是完美的，但对于所有的推荐算法来说在实验设置上是公平的。从而在两个推荐任务中证明 ContextMF 算法的优势。

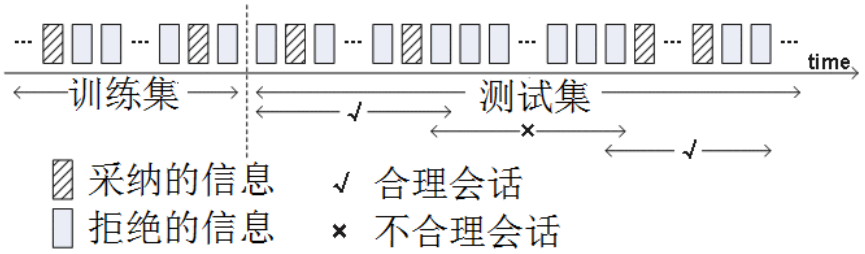


图 3.7　实验设置中合理的在线会话用作测试集：首先把数据集划分为训练和测试集，接着从测试集中用在线会话参数选出在线会话，最后在至少有两个采纳行为的在线会话上测试算法效果。

实验中实现了如下的算法作为基线与 ContextMF 模型做比较：

- 基于内容的过滤算法 ContentBased[1]：该算法根据用户采纳过的信息内容给其推荐相似内容的信息。它只考虑了用户在内容上偏好，但没有使用社交关系和交互信息。
- 基于项目的协同过滤算法 ItemCF[2]：标准的基于项目的协同过滤算法假设用户会与有同样行为的用户继续产生同样地采纳行为。这只是用了用户与信息的交互行为信息。
- 信任反馈算法 FeedbackTrust[58]：该算法是在基于信任的标准算法[56] 上添加反馈机制。该算法能够准确的根据用户相关性计算用户之间的信任程度，但这只利用了用户之间的交互信息。
- 基于影响力的推荐算法 InfluenceBased[47]：该方法用梯度下降方法从社交关系中估计用户的影响力。这也只利用了用户之间的交互信息，而没有去分析用户的个人兴趣偏好。
- 基于矩阵分解的社交推荐算法 SoRec[17]：该方法同时分析社交关系和用户对信息的交互数据，使用矩阵分解模型抽取出用户和信息的特征向量。但是用户与用户的交互信息并没有被考虑。
- 社交正则化的推荐算法 SoReg[21]：该方法使用社交正则化来设计一个矩阵分解的目标函数来约束用户特征。它没有考虑到真实反映用户个人兴趣爱好的信息内容。用户和信息的特征需要用信息内容做约束。

同样地还实现了不同版本的 ContextMF 算法。

- 仅用影响力的上下文推荐算法 InfluenceMF：该方法只采用社交影响力的社

交上下文因素，被调整后的最小化目标函数是

$$\mathcal{J} = \|\mathbf{R} - \mathbf{S}\mathbf{G}^\top\|_F^2 + \gamma\|\mathbf{S} - \mathbf{F}\|_F^2 + \delta\|\mathbf{S}\|_F^2 \tag{3-41}$$

- 仅用兴趣爱好的上下文推荐算法 PreferenceMF：该方法只采用兴趣爱好的社交上下文因素，调整后的目标函数是

$$\mathcal{J} = \|\mathbf{R} - \mathbf{U}^\top\mathbf{V}\|_F^2 + \alpha\|\mathbf{W} - \mathbf{U}^\top\mathbf{U}\|_F^2 + \beta\|\mathbf{C} - \mathbf{V}^\top\mathbf{V}\|_F^2 + \eta\|\mathbf{U}\|_F^2 + \lambda\|\mathbf{V}\|_F^2 \tag{3-42}$$

实验中用如下四种经典的评价标准来评估推荐算法的效果：(1) 预测错误率：对于第一个任务来说，算法预测用户行为有多准确；(2) 前 K 个信息的推荐效果：对于第二个任务来说，算法在给出最优的 K 个推荐信息时有多成功；(3) 排名评价系数：对于第二个任务来说，算法在信息做排名时有多好；(4) 稳定系数：梯度下降算法对于同样地数据做 100 次试验的结果有多稳定。要想比较和其他算法在预测质量上的差别，用两个常用的错误率评价标准：平均绝对误差（Mean Absolute Error，简称 MAE）和均方根误差（RMSE），其中 MAE 定义为

$$MAE = \frac{1}{|\mathcal{R}|}\sum_{R_{ij}\in\mathcal{R}}|\mathbf{R_{ij}} - \mathbf{S_i}\mathbf{G_j}^\top \odot \mathbf{U_i}^\top\mathbf{V_j}| \tag{3-43}$$

其如果第 i 个用户采纳第 j 个信息，那么 R_{ij} 为 1，否则为 0。RMSE 定义为

$$RMSE = \sqrt{\frac{1}{|\mathcal{R}|}\sum_{R_{ij}\in\mathcal{R}}(\mathbf{R_{ij}} - \mathbf{S_i}\mathbf{G_j}^\top \odot \mathbf{U_i}^\top\mathbf{V_j})^2} \tag{3-44}$$

因此，更小的 MAE 或者 RMSE 值表示有更好的预测效果。前 K 个信息的推荐效果和预测准确度相比，也是很重要的，因为网页中能够展现的推荐信息是有限的，在实际应用中推荐前 K 个信息是有意义的。在线会话中，每一个算法能够给出一个 K 个推荐信息的列表。这里用 Precision@K[2,13] 和 NDCG@K[111] 来评价推荐效果。前 K 个项目的准确率（Precision@K）定义为所推荐的 K 个信息中被采纳的信息百分比。所以一个高的 Precision@K 意味着更好的推荐效果。NDCG 是归一化后的 DCG（Discounted Cumulative Gain，折扣增益值）。DCG 是被排序信息相关性的权重和，给定会话下前 K 个项目的 DCG 是

$$DCG@K = \sum_{r=1}^{K}\frac{y(r)}{log(r+1)} \tag{3-45}$$

其中排名 r 的信息的相关性是 $y(r)$，对数化折扣为 $\frac{1}{log(1+r)}$。相关性 $y(r)$ 是从信息排名映射到一个有限集合 $\mathcal{Y} = \{0,1\}$，其中 1 表示采纳行为，也就是第 r 个信息被用户分享和转发，0 表示拒绝行为。理想的 DCG 值也就是 IDCG 可以简单为 DCG 结果的最大值。NDCG 是用 IDCG 归一化 DCG 后的结果，所以取值在区间 $[0,1]$

内。一个更大的 NDCG@K 意味着算法在推荐前 K 个消息时更准确。另外还采用两套排名评价系数：(1) $\hat{\tau}$ 和 $\hat{\rho}$，分别是 Kendall 和 Spearman 用于评价排名准确程度的系数。(2) ERR，是 Sanderson 给出的最佳的对用户行为做排名的指标[36]。这几个评价系数都是越大说明推荐效果越好。排名评价系数 $\hat{\tau}$ 和 $\hat{\rho}$ 是直觉的统计指标，也就是计算模型给出的排名中有多少排名数据是乱序的，定义

$$T = \sum_{r<s} I(y(r) > y(s)) \tag{3-46}$$

其中 (r, s) 是每一对推荐信息的顺序，$I(x)$ 是如果 x 是真的返回 1，假的返回 0 的映射函数。$y(r)$ 定义为相关性函数。顺序变换的权重和可以定义为：

$$R = \sum_{r<s} (s - r) \cdot I(y(r) > y(s)) \tag{3-47}$$

这两个指标可以线性转换 [-1,1] 的范围内，其中 1 表示模型完美的推荐效果，-1 表示模型最差的推荐效果（也就是完全相反的排序）。由此定义非参的相关性系数（n 是信息数量）：

$$\hat{\tau} = 1 - \frac{4T}{n(n-1)} \tag{3-48}$$

$$\hat{\rho} = 1 - \frac{12R}{n(n-1)(n+1)} \tag{3-49}$$

Chapelle 提出 ERR（Expected Reciprocal Rank）来评价相关性是否正确，并证实了 ERR 比 DCG 在评价用户满意度上效果更好[285]，ERR 如下计算：

$$ERR = \sum_{r=1}^{n} \frac{1}{r} \prod_{i=1}^{r-1} (1 - y(i)) y(r) \tag{3-50}$$

从上式知道 ERR 值在区间 $[0, 1]$ 中。如果 ERR 值越大，那么算法返回的是一个更好的消息排序。如果是算法对于被采纳的信息比起拒绝的信息能给出更大差别的值，那么推荐系统的效果就更好。为了证实算法是否有较强的区分度，可以进行一组 T 测试来比较正确预测和错误预测之间的数值差别。当 T 测试值越大，当用户采纳行为发生时，预测就越准确。

模型中的参数是非常重要的，但设置并不困难。工作中调整 ContextMF 模型中的参数以及所有基线算法的参数使它们都达到最佳推荐效果。在这样的条件下，实验设置和参数设置对实验结果的比较来说都是公平的。下面介绍如何容易地、自动地设置为模型设置合适的参数。模型的权衡参数包括 α，β，γ，δ，η 和 λ，它们能够调整目标参数中每一项的权重：(1) α 和 β 使用话题层面的用户相似度和信息相似度来正则化用户和信息的特征空间，决定了个人兴趣偏好这一上下文因素

在模型中的权重；(2) γ 使用用户交互频率来正则化社交影响力，决定了社交影响力这一上下文因素在模型中的权重；(3) δ，η 和 λ 分析社交影响力矩阵 **S** 的规模；用户特征矩阵 **U** 的规模和信息特征矩阵 **V** 的规模来调整模型的目标函数。用控制变量法来使得这些参数达到最佳效果。图3.8展示了达到最佳预测效果（最低的 RMSE）时，参数们都不是太大也不是太小。这里给出用一套自动的参数设定方法来设置该社会化推荐模型 ContextMF 的参数。

$$\alpha \leftarrow 10^{-2} \times \frac{\|\mathbf{R} - \mathbf{S}\mathbf{G}^{\top} \odot \mathbf{U}^{\top}\mathbf{V}\|_F^2}{\|\mathbf{W} - \mathbf{U}^{\top}\mathbf{U}\|_F^2} \propto 10^{-2} \times \frac{N}{M} \tag{3-51}$$

$$\beta \leftarrow 10^{-2} \times \frac{M}{N}, \gamma \leftarrow 10^{-2} \times \frac{N}{M} \tag{3-52}$$

$$\delta \leftarrow 10^{-4} \times \frac{\|\mathbf{R} - \mathbf{S}\mathbf{G}^{\top} \odot \mathbf{U}^{\top}\mathbf{V}\|_F^2}{\|\mathbf{S}\|_F^2} \propto 10^{-4} \times \frac{N}{M} \tag{3-53}$$

$$\mu \leftarrow 10^{-4} \times \frac{N}{k}, \lambda \leftarrow 10^{-4} \times \frac{M}{k} \tag{3-54}$$

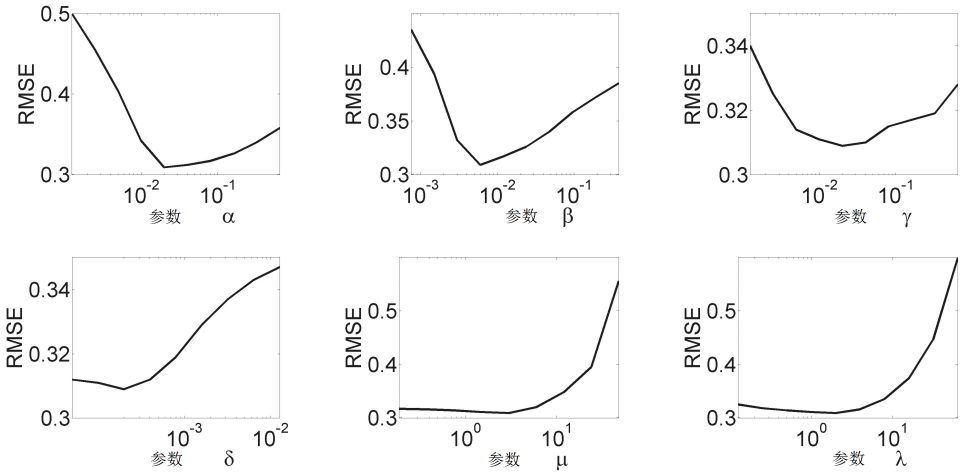
其中 M 和 N 是用户数量和信息数量，k 是特征空间的特征数量。

下面是训练矩阵 **U** 和 **V** 来寻找最合适的特征数量 k。如果 k 太小，推荐系统就无法很好地区分用户和信息之间的差别。如果 k 太大，用户和信息有太过独立，而且计算复杂度非常高。因此要在人人网和腾讯微博数据集上让 k 从 3 到 80 变化来做实验寻找最合适的 k 值。图3.9中发现随着特征数量 k 的增加，RMSE 值逐渐减小。很显然地看到当 $k \geq 60$ 时，RMSE 减小得逐渐变慢。考虑到推荐效果和运行效率，选择 $k = 60$ 作为默认的特征空间大小。

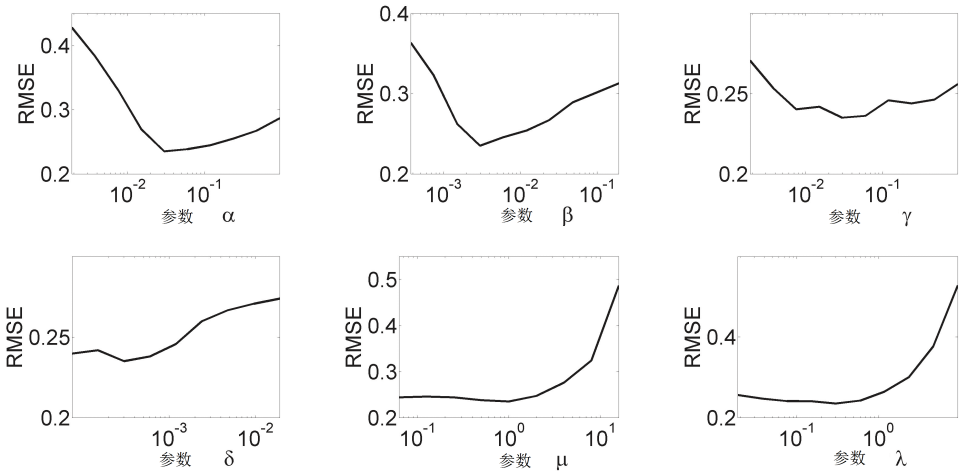
从图3.10中观察到 RMSE 和目标函数值 $\mathcal{J}$ 逐渐随着迭代次数的增加而减小。通过有效地整合社交上下文正则项，ContextMF 能够成功避免在梯度下降方法中常见的过拟合问题。在两大社交媒体数据集中，为了达到收敛的效果和可接受的运行时间，算法运行 60 次迭代过程效果很好。根据上述方法也为 PreferenceMF 和 InfluenceMF 的模型调整参数，另外实验中搜索最好的配置帮助其他基线算法在真实数据集上达到最好的效果，也确保后续的比较结果是公平的。

图3.11中展示了 100 次重复实验后 RMSE 的标准差 (σ_{RMSE}) 随着在线会话的时间窗大小 Δt_{max} 和会话中用户采纳行为数量 n_{min} 的变化趋势。如果 Δt_{max} 太大，用户可能已经离开会话，状态变成离线；如果值太小，用户可能并不足够活跃。如果 n_{min} 太小，用户在会话中不够活跃；如果值太大，数据集就有过多的信息被选作测试，缺乏训练信息。因此选择当 σ_{RMSE} 达到最小时的参数值。实验中设定 Δt_{max}=5 分钟，n_{min}=15。

实验中用预测错误率（MAE 和 RMSE）、排序系数（$\hat{\tau}$，$\hat{\rho}$ 和 ERR）和显著性指标（T 测试系数）来评测 ContextMF 模型和基线算法。首先，如表3.3所示，基



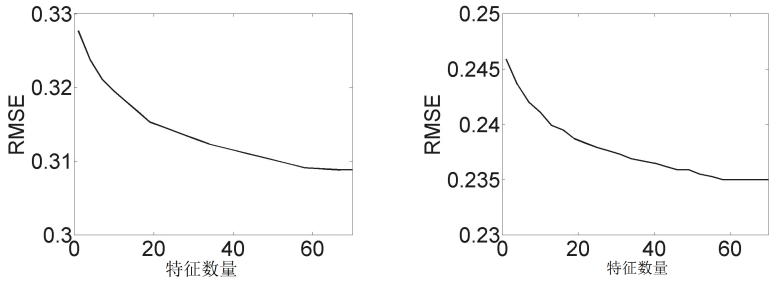
(a) 人人网数据集的权衡参数



(b) 腾讯微博数据集的权衡参数

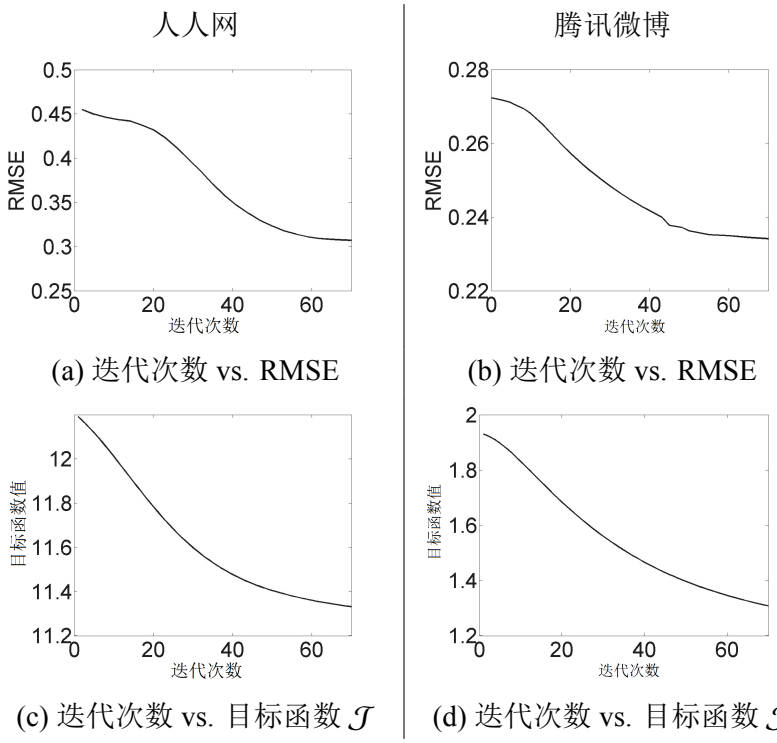
图 3.8　调整 α，β，γ，δ，η 和 λ 参数来使得模型达到最有效果。对于不同的推荐算法，都采用同样的训练数据。

于矩阵分解模型的社交上下文推荐模型能够比其他基线算法推荐效果更好。在人人网和腾讯微博数据集上，降低 MAE 程度达到 19.1% 和 12.8%，把 RMSE 分别降低了 24.2% 和 20.7%，比起时下最好的社会化推荐算法 SoReg 提升 ERR 指标分别达到 19.7% 和 11.4%。ContextMF 模型能够比起 PreferenceMF 和 InfluenceMF 的推荐准确率都有大幅提升：在人人网和腾讯微博数据集上能够分别降低 MAE 达到 25.2% 和 39.7%，降低 RMSE 达到 21.7% 和 31.5%，提升 Kendall 排序系数分别为 12.1% 和 2.27%，提升 Spearman 排序系数分别为 12.2% 和 6.04%，提升 ERR 指标分别达到 46.5% 和 31.6%。所有这些实验结果证明了一个同时考虑两大社交上下文因素（个人兴趣爱好和社交影响）的社会化推荐模型能够比起只考虑其中一个的模型要更好。另外用 T 测试结果，也就是预测出采纳行为和拒绝行为的概率值比例值，称为显著性；与基线算法比较 ContextMF 算法的显著性。ContextMF



(a) 人人网　　(b) 腾讯微博

图 3.9　在两大社交数据，RMSE 随着特征数量 k 的增加而减小，收敛在 $k = 60$。



(a) 迭代次数 vs. RMSE　　(b) 迭代次数 vs. RMSE

(c) 迭代次数 vs. 目标函数 $\mathcal{J}$　　(d) 迭代次数 vs. 目标函数 $\mathcal{J}$

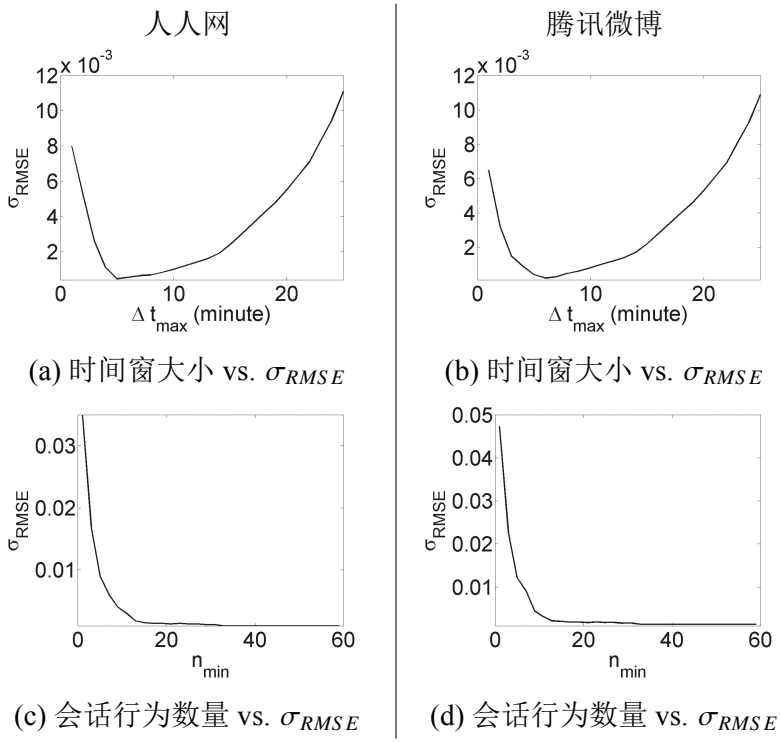
图 3.10　在人人网和腾讯微博上 RMSE 值和目标函数值 $\mathcal{J}$ 随着迭代次数的增加而减小并收敛到大约 60。

算法能够给出最好的 T 测试结果，也就是比最好的基线算法分别高 1.78 倍和 1.26 倍，因此社交上下文推荐模型有更好的显著性。值得强调的是：(1) PreferenceMF 和 InfluenceMF 比起 SoRec 要效果好，证实了兴趣偏好和社交影响力的有效性；(2) ContextMF 比起 PreferenceMF 和 InfluenceMF 在推荐效果上的大幅提升证实了完整融合所有社交上下文信息（从兴趣偏好和社交影响力两大层面上）的重要性；(3) ContextMF 比起 SoReg 要好很多，证明了从用户采纳信息行为动机融合两大社交上下文因素的重要性。

第二，如图3.12所示，实验中比较 ContextMF 模型和其他基线算法在推荐 K 个信息的效果（Precision@K 和 NDCG@K）上的差别。随着推荐信息数量 K 的降

表 3.3　社会化推荐算法在两大社交媒体数据集上的效果

方法	预测错误率		排序评分			显著性测试		
	MAE	**RMSE**	$\hat{\tau}$	$\hat{\rho}$	**ERR**	采纳	拒绝	**T 测试**
人人网数据集								
ContentBased[1]	0.384	0.477	0.541	0.540	0.325	0.702	0.665	1.06
ItemCF[2]	0.360	0.451	0.590	0.599	0.397	0.360	0.268	1.34
FeedbackTrust[58]	0.376	0.468	0.543	0.547	0.378	0.363	0.343	1.06
InfluenceBased[47]	0.386	0.469	0.539	0.545	0.365	0.641	0.590	1.09
SoRec[17]	0.328	0.413	0.617	0.620	0.452	0.473	0.347	1.37
SoReg[21]	0.299	0.354	0.709	0.714	0.561	0.523	0.336	1.56
InfluenceMF	0.310	0.377	0.686	0.701	0.477	0.351	0.213	1.65
PreferenceMF	0.303	0.376	0.694	0.704	0.465	0.132	0.056	2.38
ContextMF	**0.242**	**0.309**	**0.778**	**0.790**	**0.699**	0.456	0.107	**4.24**
腾讯微博数据集								
ContentBased[1]	0.258	0.364	0.773	0.778	0.476	0.417	0.276	1.51
ItemCF[2]	0.238	0.337	0.787	0.805	0.544	0.637	0.244	2.62
FeedbackTrust[58]	0.283	0.389	0.709	0.712	0.492	0.792	0.610	1.30
InfluenceBased[47]	0.265	0.381	0.716	0.728	0.491	0.800	0.392	2.04
SoRec[17]	0.226	0.333	0.797	0.806	0.555	0.495	0.058	8.53
SoReg[21]	0.200	0.296	0.839	0.842	0.667	0.552	0.060	9.17
InfluenceMF	0.218	0.321	0.818	0.82	0.572	0.522	0.062	8.42
PreferenceMF	0.211	0.309	0.838	0.845	0.568	0.576	0.052	11.1
ContextMF	**0.151**	**0.235**	**0.857**	**0.896**	**0.753**	0.812	0.058	**14.0**



(a) 时间窗大小 vs. σ_{RMSE}　(b) 时间窗大小 vs. σ_{RMSE}

(c) 会话行为数量 vs. σ_{RMSE}　(d) 会话行为数量 vs. σ_{RMSE}

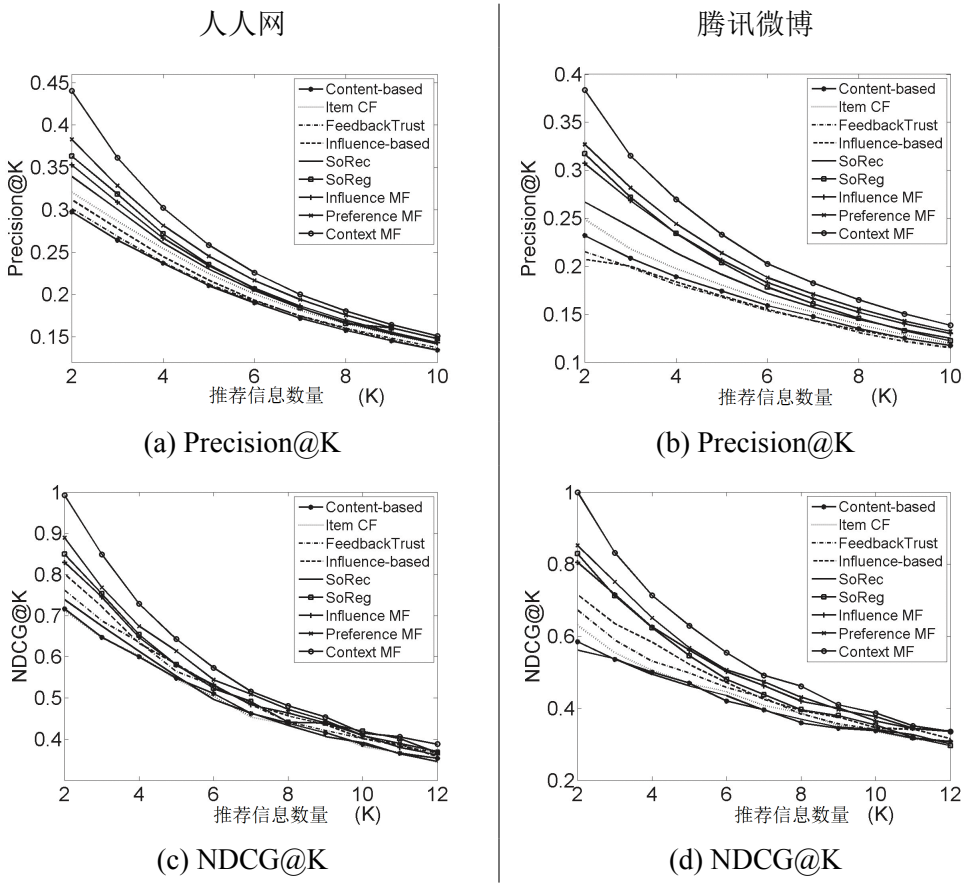
图 3.11　选择合适的会话参数 Δt_{max} 和 n_{min} 来确定在线会话。

低，推荐效果会逐步增加，但提升程度会逐步降低。与最好的基线算法 SoReg 相比，在人人网数据集上，Precision@5 提升 21.7%，Precision@10 提升了 10.8%；相似地，在腾讯微博数据集上，Precision@5 提升 12.3%，Precision@10 提升了 6.85%。此外还测算了 NDCG@K 的数值：在人人网和腾讯微博数据集上 NDCF@5 分别提升了 4.7% 和 10.8%。当 K 很小的时候 ContextMF 模型的优势很明显。因为用户采纳信息的行为非常稀疏，所以当 K 非常大的时候是很难区分出效果很好的算法的。于是 ContextMF 算法和基线算法都在 K 变得大时收敛到同一个结果。

第三，重复 100 次实验以检验模型的稳定性。如表3.4所示，MAE 和 RMSE 的低标准差展现了 ContextMF 算法不仅仅在两大社交媒体数据集上效果好，而且运行结果并没有很大的起伏。

表 3.4　在两大社交媒体数据集上推荐效果稳定性比较

	MAE	**RMSE**	$\hat{\tau}$	$\hat{\rho}$	**ERR**	**T 测试**
人人网数据集						
$\bar{x}$	0.2416	0.3086	0.7783	0.7897	0.6987	4.2437
σ	0.0001	0.0001	0.0006	0.0006	0.0008	0.6
腾讯微博数据集						
$\bar{x}$	0.1514	0.2348	0.8571	0.8686	0.7529	13.989
σ	0.0001	0.0002	0.0002	0.0001	0.001	0.8



(a) Precision@K　(b) Precision@K

(c) NDCG@K　(d) NDCG@K

图 3.12　在人人网和腾讯微博上的推荐 K 个信息效果评测：比起基线算法，(a-b) Precision@5 分别提升了 21.7% 和 12.3%；(c-d) NDCG@5 分别提升 4.7% 和 10.8%。

第四，分析 ContextMF 模型处理增量数据的能力。如表3.5所示，通过不同的组合得到 8 份数据集：(1) 人人网还是腾讯微博；(2) 新用户还是新信息；(3) 新增 ΔM 个用户和新增 ΔN 个信息，其中 $\Delta M, \Delta N \in \{1,000, 10,000\}$。例如，在数据集 R$\Delta$M1000 中，首先从 M 个人人用户中随机选取 $\Delta M = 1000$ 个用户作为新用户。用数据中存留的 $M_0 = M - \Delta M$ 个用户的行为数据来训练社交影响力矩阵 $\mathbf{S}$ 和用户以及信息的特征矩阵 $\mathbf{U}$ 和 $\mathbf{V}$。接着，用在线增量模型Δ ContextMF 来与离线处理模型 ContextMF 以及基线算法 SoReg 做比较，并通过解决用户冷启动问题和信息冷启动问题来比较它们的效果。表3.5还展示了在线增量模型Δ ContextMF 的运行时间比起离线处理模型 ContextMF 要少很多：能够从小时级减小到秒级。改善效率可能会降低有效性，因为高阶项在Δ ContextMF 中被忽略了。然而Δ ContextMF 的 RMSE 比起 ContextMF 紧紧增大了 2.33%（越大越差），所以是适用的。另一方面Δ ContextMF 还是比 SoReg 要好的，因为这一在线处理方法还是充分利用了社交上下文信息：在人人网数据上，Δ ContextMF 的 RMSE 比起 SoReg 要小 18.5%（越小越好）；在腾讯微博上，Δ ContextMF 的 RMSE 要小 16.9%。另外，Δ ContextMF 的 ERR 在人人网和腾讯微博上也比 SoReg 要提升

表 3.5　离线处理模型 ContextMF 和在线增量处理模型Δ ContextMF 的推荐效果比较

数据集	RMSE (越小越好)			ERR (越大越好)			时间开销	
	SoReg	在线	离线	SoReg	在线	离线	在线	离线
R∆M1000	0.342	0.263	0.257	0.555	0.610	0.636	172s	41.7h
R∆M10000	0.502	0.464	0.444	0.481	0.542	0.559	1610s	41.7h
T∆M1000	0.168	0.122	0.105	0.652	0.764	0.783	54.2s	2.42h
T∆M10000	0.342	0.333	0.317	0.534	0.611	0.651	531s	2.42h
R∆N1000	0.335	0.276	0.276	0.570	0.663	0.680	97.3s	41.7h
R∆N10000	0.546	0.478	0.465	0.514	0.587	0.609	941s	41.7h
T∆N1000	0.218	0.192	0.173	0.726	0.824	0.864	17.8s	2.42h
T∆N10000	0.427	0.376	0.355	0.658	0.720	0.751	160s	2.42h

表 3.6　图3.13中消息的内容和话题分布

ID	话题 t_3	话题 t_8	内容
p_1	0.00	0.86	爱 Java，爱代码！
p_2	0.00	0.72	Have you ever read this? The Zen of Python by Tim Peter
p_3	0.02	0.91	想招网站开发程序员：1. 会写 java，2. 会写 HTML, CSS...
p_4	0.65	0.09	爱以笑生，以吻增情，以泪结束。
p_5	0.12	0.68	我遇到... Exception in thread main me.love.NoGirlFriendError
p_6	0.71	0.00	我想你，可我错过了你。
p_7	0.68	0.00	如果你选择离开我，请不要来安慰我

11.7% 和 11.9%。这证明了在真实社交媒体数据上，社交上下文推荐的在线增量算法Δ ContextMF 能够快速处理增量数据，并保持很高的准确度。

最后，实验给出真实样例来证明利用社交关系、信息内容、用户交互和采纳信息行为这些社交上下文信息是非常重要的。图3.13和表3.6中给出了在腾讯微博上的样例。这里用户 u_1 关注了 u_2 和 u_3，所以会从他们那里收到消息。在时刻 t 之前，用户 u_1 从用户 u_3 处转发了 18 条微博，而只从 u_2 处转发了 3 条信息。ContextMF 模型会去学习他们之间的社交影响力：用户 u_1 更容易和 u_3 互动，而 PreferenceMF 是无法做到的。另外，用户 u_1 转发了消息 p_1，... 和 p_4：p_1，p_2 和 p_3 都在第 8 个话题上有很高的分布值，因为内容上大多关于程序语言，代码和计算机工程；而微博 p_4 是关于爱和生活的，所以在第 3 个话题上又很高的分布值。ContextMF 能够学习个人兴趣爱好这一社交上下文因素：u_1 会喜欢这几个特定话题内容，但 InfluenceMF 方法是做不到的。这样，在时刻 t 下用户 u_1 从用户 u_2 处收到消息 p_5 和 p_6，从用户 u_3 处收到消息 p_7。ContextMF 关注给用户 u_1 推荐的效果，也就是看推荐之后他是否有采纳这些信息。

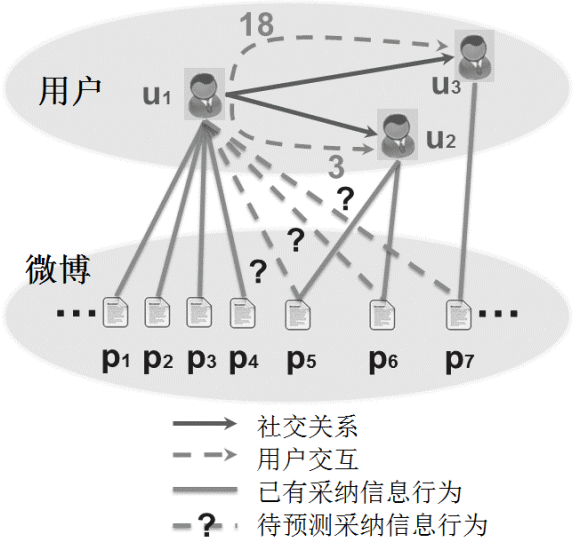


图 3.13　腾讯微博上的社交推荐样例：用户 u_1 关注了用户 u_2 和 u_3，所以能够获取他们的信息；在时间 t 之前，用户 u_1 转发了用户 u_3 的 18 条消息，但只转发了用户 u_2 的 3 条消息；用户 u_1 转发了消息 p_1，…，p_4。在时刻 t，用户 u_1 从用户 u_2 处收到信息 p_5 和 p_6，从用户 u_3 处收到信息 p_7。在这样的条件下，任务是预测 u_1 是否会转发这些信息。

表 3.7　给用户 u_1 推荐信息 p_5，p_6 和 p_7 的概率值

	$\mathcal{R}(u_1,p_5)$	$\mathcal{R}(u_1,p_6)$	$\mathcal{R}(u_1,p_7)$
真实值	1	0	1
ContextMF	**0.884**	**0.112**	**0.845**
PreferenceMF	0.901	**0.354**	**0.323**
InfluenceMF	**0.190**	0.094	0.854

表3.7中给出不同算法在预测用户 u_1 到微博 p_5，p_6 和 p_7 的链接权重，也就是转发事件发生的概率。帖子 p_5 关于程序语言，所以用户 u_1 会喜欢。从 u_1 的行为历史来看，ContextMF 认为转发概率为 88.4%，而 InfluenceMF 只知道用户 u_1 不会从 u_2 处转发很多信息，所以给出的转发概率只有 19.0%。微博 p_6 和 p_7 有相似的话题分布，然而 p_7 是从 u_3 转发而来，而 p_6 是从 u_2 转发而来，所以对 u_1 的影响是 p_7 会更大，因为 u_3 带来的影响更大（u_1 已经转发了他的 18 条微博）。ContextMF 和 InfluenceMF 都预测 u_1 会喜欢 p_7，但 PreferenceMF 并不能够。ContextMF 的预测结果比起 PreferenceMF 和 InfluenceMF 都更接近与真实值，因为 ContextMF 能够把所有的社交上下文信息融入到单一模型中。

3.3.2　时空环境下行为预测性能和模式发现效果

本节中通过实验在行为预测任务上验证 FEMA 算法的有效性，效率和鲁棒性。同时给出了动态行为模式上的有趣发现，强调 FEMA 模式发现有效性。

	MAE 数据		微博数据
研究人员	7,777	源用户	6,200
研究机构	651	目标用户	1,813
关键字	4,566	微博词汇	6,435
时间	32 年	时间	43 天
合作关系	98,671	社交关系	465,438
论文数量	171,519	微博数量	519,624

表 3.8　数据集各项统计值

实验中使用了以下两个数据集：

- MAS 数据[277]：这是微软学术搜索数据库中的公开数据集，其中包含三个文件。第一个文件是 25 万研究者的个人属性，包括姓名和研究机构。第二个文件包含 250 万的论文，包括论文标题、年份和关键字。第三个文件包括研究者和论文之间的关系。首先把这三个文件通过研究者、研究机构、关键字和论文年份相关联，然后预处理数据保证研究者、研究机构、关键字都至少出现 10 次。于是有在 32 年（1980 年到 2012 年）里的 7,777 个研究人员，651 个研究机构和 4,566 个关键字。每一年的张量平均密度小于 $3 \times 10^{-5}\%$，而研究人员合作的数据密度大约有 0.2%。
- Weibo 数据: 腾讯微博是中国最大的微博平台之一。微博用户通过在微博中添加 “@ 姓名” 可以提及他们想提及的人。数据集中包含两个文件。第一个文件包含发布时间、用户和微博内容。从内容中可以提取出被提及的目标用户。第二个文件包括了社交网络信息，是源用户和目标用户之间的关注关系。在预处理之后有 519,624 条记录（源用户，目标用户，微博词汇，时间），其中包含了 6,200 个源用户，1,813 个目标用户，6,435 个微博词汇和 43 天的时间刻度（从 2011 年 11 月 9 日到 2011 年 12 月 21 日）。每周张量中的平均密度为 $2 \times 10^{-5}\%$，而社交关系矩阵的密度为 0.7%.

表3.8总结了学术和微博数据集的特点。这里使用这两个数据集来测试两种不同的行为预测任务。这两个任务都希望能从新得到的行为数据预测出未来的行为。

- 2W (Who-What and Who-Whom) 预测：预测给定研究人员 u 是否发表含有关键字 v 的论文，预测给定源用户 u 是否会在他们的微博中提及目标用户 v，不论 u 在哪个研究机构，也不论微博内容是什么；
- 3W (Who-Where-What and Who-Whom-What) 预测：目标是给定研究人员 u 是否在给定的研究机构 w 中发表含有关键字 v 的论文，预测给定源用户 u 是否会在含有词汇 w 的微博中提及目标用户 v。

图3.14展示如何用数据集来做实验的。两个数据集都被分为三部分：用于初始化

的训练数据，用于动态分析的训练数据，和用于测试的数据。用最初的 30% 行为数据初始化，然后每次增加 5% 数据，预测后接的 20% 行为；这样的测试可以进行 10 次。换句话说，是做 $T = 10$ 次推荐效果的测试，每次训练数据占总数据的比例为 $\alpha_t = 35\%, 40\%$ to 80%，其中 $t = 1, 2, \ldots, T$。

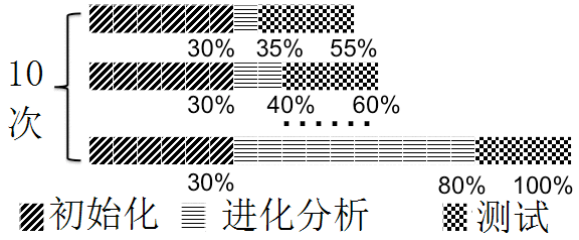


图 3.14　实验设置：用最初的 30% 行为数据初始化，然后每次增加 5% 数据并进行 10 次预测紧接的 20% 行为。

通过和下面的算法比较来评价有效性和效率。实验中实现了三种 FEMA 的配置：

- FEMA：该模型用研究人员 -研究机构 -关键字的张量做学术研究行为建模，并用合作关系做正则化约束。相似的，用源用户 -目标用户 -微博词汇的张量做微博提及行为建模，并用社交关系做正则化约束。
- EMA：该模型使用多维张量做行为建模，但并不使用正则化约束。也就是不用合作关系和社交关系信息。
- EA：用研究人员 -关键字和源用户 -目标用户矩阵来做行为建模，但不用多维信息。所以这个方法只用于 2W 预测。

此外还实现了下面的先进算法，并作比较：

- CP[103] (CANDECOMP/PARAFAC)：对更新的张量每次分解为多个秩为 1 的张量的和。这个方法需要每个维度的组的大小 $r^{(1)} = r^{(2)} = r^{(3)} = R$.
- HOSVD[280] (High-Order SVD，高维 SVD)：新张量的 Tucker 分解是主成分分析（Principal Component Analysis，简称 PCA）的高维度表示。
- DTA[101] (Dynamic Tensor Analysis，动态张量分析)：该方法能够快速地对张量进行降维处理，更新协方差矩阵。这并不需要保存任何的历史张量，但还是不得不分解巨大的协方差矩阵。和这个方法比较效率后知，本文所提出的在线处理方法更有效率。

为了比较近似算法的质量和效率，还实现了 FEMA 的离线处理方法：

- FMA：该模型和 FEMA 所使用的知识是一样的，然而，这个方法是把新增的行为数据和历史数据混合，并每次对更新后的张量做分解。

实验中在 MATLAB 上实现所有上述方法[97]，并在单机上进行实验：用 2.40GHz

的 Intel Xeon CPU 和 32GB 的 RAM 的机器，这个机器的系统为 Windows Server 2008。使用的默认参数为 $r^{(i)} = 50$ 和 $\mu^{(i)} = 0.3$，其中 $i = 1, 2, 3$。不同的参数设置的讨论后续会给出。

下面介绍一下评价指标。对于第一个任务，即行为预测，使用标准的评价指标：平均绝对误差（MAE）和均方根误差（RMSE）[73]：

$$MAE = \frac{\sum_{(u,v,w)\in D} |r_{u,v,w} - \hat{r}_{u,v,w}|}{|D|} \tag{3-55}$$

$$RMSE = \frac{\sum_{(u,v,w)\in D} (r_{u,v,w} - \hat{r}_{u,v,w})^2}{|D|} \tag{3-56}$$

其中 D 表示测试集合；$r_{u,v,w}$ 是研究人员 u 在研究机构 w 发表含有关键字 v 的论文这一行为的概率；$\hat{r}_{u,v,w}$ 是测试集中行为的频数，而 0 表示没有发生。较小的 MAE 和 RMSE 意味着模型的效果更好。同时，还使用两种频繁使用的预测指标：准确率 (Precision) 和召回率 (Recall)[116] 来评价预测结果的排序质量是否好。定义 $T(u, v, w)$ 为测试集中的行为集合，定义 $P(u, v, w)$ 为被推荐的行为集合。准确率是说预测正值的行为有多少的确是正值，而召回率是说所有的正值行为有多少被预测到。于是就可以通过改变 $P(u, v, w)$ 的预测值下限画出准确率 -召回率曲线。

$$Precision = \frac{|P(u, v, w) \cap T(u, v, w)|}{|P(u, v, w)|} \tag{3-57}$$

$$Recall = \frac{|P(u, v, w) \cap T(u, v, w)|}{|T(u, v, w)|} \tag{3-58}$$

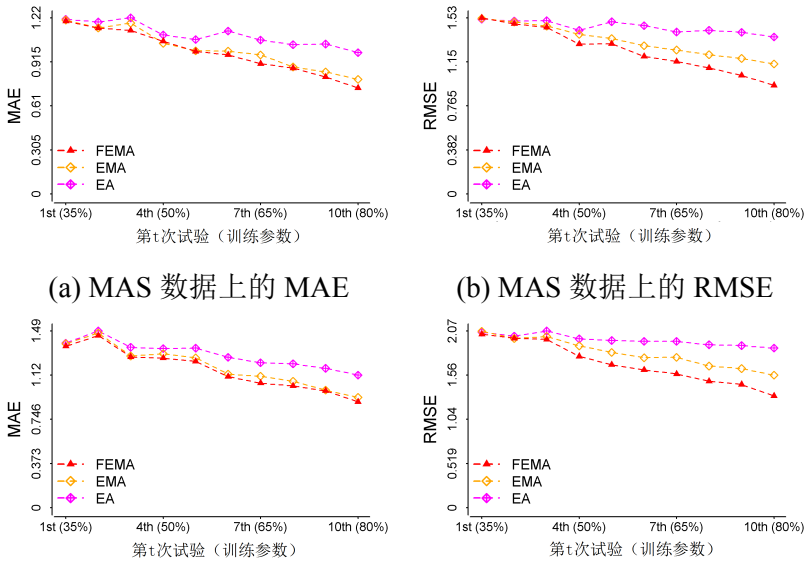
更高的准确率和召回率意味着模型更好。微博数据上可以为每一个源用户 u 给出 $N(N = 5)$ 个微博数据中预测的要被提及的用户集合 $R_{u,w}$，如果目标用户 v 出现在列表中，称为一次命中。于是定义命中率[3]（Hit Ratio）为

$$Hit\ Ratio = \frac{\sum_{u,v,w} I(v \in R_{u,w})}{|U|} \tag{3-59}$$

其中 $I(\cdot)$ 是指示函数，$R_{u,w}$ 是给定用户 u 和微博词汇 w 的前 n 个被提及的用户，v 是测试集合中 u 发出带有 “@v” 的微博的用户。更高的命中率意味着模型更好。对于第二个任务，把数据按照研究人员的研究机构维度和微博词汇维度进行压缩降维，得到的是矩阵数据。同样是用 MAE，RMSE，准确率，召回率和命中率来预测行为。也就是在所有公式中用 (u, v) 来替代 (u, v, w)，用 (u) 代替 (u, w)。

接下来用三种不同的实验来证实 FEMA 的效果和效率。首先，通过在 MAS 数据和 Weibo 数据上进行 2W 预测来测试使用多面信息的好处。第二，通过 3W 预测来测试灵活正则项的效果。最后证明 FEMA 动态分析的有效性和效率。

实验中展现了预测行为的 2W 任务结果，包括在 MAE 数据上的研究人员 -关键字行为和 Weibo 数据上的目标用户 -微博词汇的行为。将 FEMA 和 EMA 的效



(a) MAS 数据上的 MAE　(b) MAS 数据上的 RMSE

(c) Weibo 数据上的 MAE　(d) Weibo 数据上的 RMSE

图 3.15　多面分析能够让 2W 预测效果更为准确：FEMA 和 EMA 都比 EA 在进行人类行为建模上效果要好，也就是说用高维张量更为有效，能够得到更小的 MAE 和 RMSE。

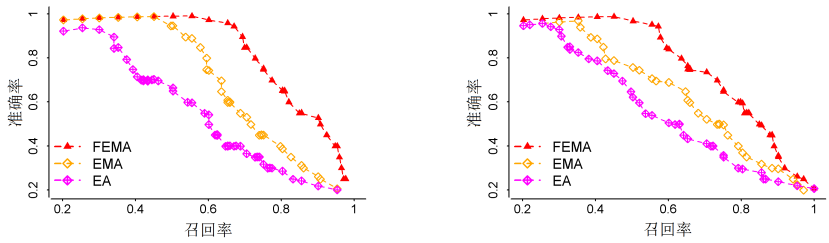
果与 EA 作比较，因为 EA 并不用研究机构和微博词汇的信息。图3.15展示了上述算法随着 α_t 从 35% 到 80%，逐步增加 5% 的预测效果。图3.15(a) 和图3.15(b) 给出的是 MAE 数据上的结果；图3.15(c) 和图3.15(d) 给出的是 Weibo 数据上的结果。即使 EMA 比起 EA 要更好，但 FEMA 还是得到了最小的 MAE 和 RMSE。此外，在表3.9中展现了当 $\alpha_t = 80\%$ 时的预测结果。FEMA 比起 EA 来说，在 MAE 数据上减小 RMSE 达到 30.8%，在 Weibo 数据上减小 RMSE 达到 30.0%。

	MAE 数据		Weibo 数据	
	MAE	RMSE	MAE	RMSE
FEMA (+ 灵活正则项)	**0.735**	**0.944**	**0.894**	**1.312**
EMA (张量)	0.794	1.130	0.932	1.556
EA (矩阵)	0.979	1.364	1.120	1.873

表 3.9　基于张量的方法 FEMA 和 EMA 比起基于矩阵的算法 MA 在 2W 预测任务上有更小的 MAE 和 RMSE。FEMA 用张量来做行为建模，学习灵活的正则项，同时得到最小的错误。如果 MAE 和 RMSE 越小，模型效果越好。

图3.16给出了人类行为预测的准确率 -召回率曲线。证实了基于张量的方法 EMA 比起基于矩阵的方法 EA 的推荐效果更好，而在 MAE 数据和 Weibo 数据上 FEMA 能达到最好的推荐效果。

EA 算法使用研究人员 -关键字矩阵来为学术研究行为建模，而 EMA 和 FEMA



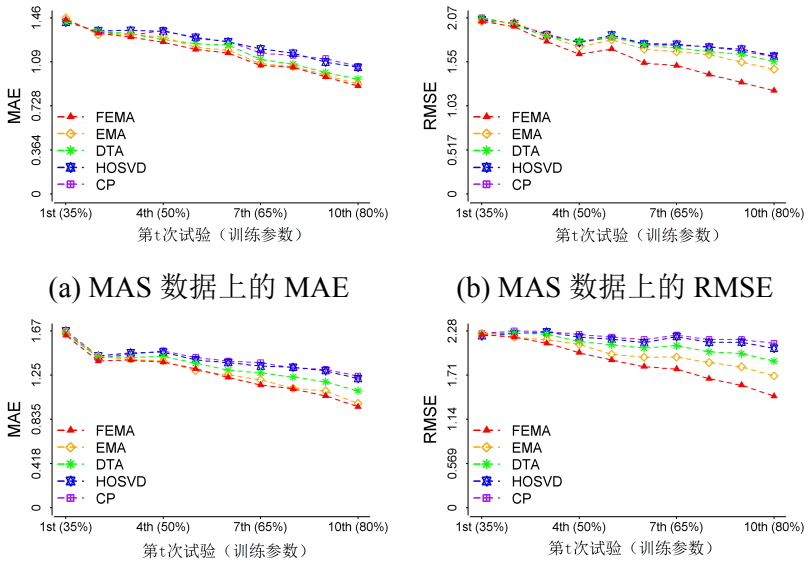
(a) MAS 数据上的准确率 -召回率　(b) Weibo 数据上的准确率 -召回率

图 3.16　FEMA 和 EMA 使用高维张量来刻画人类行为会比起基于矩阵的方法 EA 在 2W 预测任务上展现出更好的准确率和召回率（当 $t = 10$ 和 $\alpha_t = 80\%$）。如果准确率和召回率更大的时候，模型更好。

是用研究人员 -研究机构 -关键字的张量模型建模，后者的推荐效果更好。这是因为研究机构的信息比起关键字更有效果：如果一个研究人员改换了研究机构，他的科研方向也会发生改变，因为他的合作人员和项目都会改变。例如在之前给出的图3.5中，可以知道 Prof. Han 从西蒙弗雷泽大学 (Simon Fraser University) 搬去伊利诺伊大学香槟分校 (UIUC)，他主要的研究方向会从数据库系统转变为数据挖掘。多面分析方法 EMA 和 FEMA 能够从 MAE 数据中学习到研究机构信息，也就可以更好的预测研究人员将会发表什么关键字的论文。相似的，EMA 和 FEMA 使用微博词汇作为第三个维度来为微博数据集中的提及行为建模。微博用户通常会根据不同的微博内容去提及不同的用户。例如体育迷会在向他们最欣赏的运动员发去祝贺、祝福、祝愿消息的时候提及他们；如果提及关于婚姻、毕业、旅行和折扣信息，也会提及这些偶像知名的亲朋好友。

下面介绍 3W 预测任务的结果：在学术研究数据集 MAS 上预测研究人员 -研究机构 -关键字的行为，在微博数据集 Weibo 上预测源用户 -目标用户 -微博词汇的提及行为。将 FEMA 算法和 EMA 以及其他三个方法 DTA，HOSVD 和 CP 作比较，这些方法都不在分解之外使用灵活正则项做约束。和图3.15相似，在图3.17中给出了这些方法的 MAE 和 RMSE 随着 α_t 从 35% 到 80% 逐步增大的结果。图3.17(a) 和图3.17(b) 画出了在 MAS 数据集上的结果，而在图3.17(c) 和图3.17(d) 给出在 Weibo 数据集上的结果。随着训练数据的增加，模型能够学到更多的知识，所以 MAE 和 RMSE 随着训练集变大逐步降低。而 FEMA 方法通常能够得到最小的 MAE 和 RMSE，说明灵活的正则项是可以缓解稀疏度问题的，从而提升预测任务的准确性。此外，在表3.10中给出了在训练参数为 $\alpha_t = 80\%$ 时的推荐效果。FEMA 比起所有基线算法中最好的还是在 MAS 数据上降低 RMSE 达到 17.1%，在 Weibo 数据上降低 RMSE 达到 15.4%。

和图3.16相似，图3.18给出准确率 -召回率曲线来说明灵活正则项的有效性。



(a) MAS 数据上的 MAE　(b) MAS 数据上的 RMSE

(c) Weibo 数据上的 MAE　(d) Weibo 数据上的 RMSE

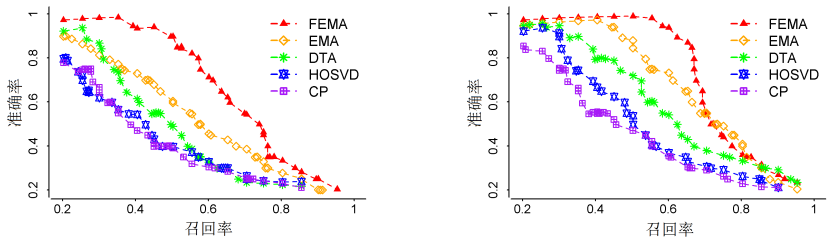
图 3.17　灵活正则项能够缓解高稀疏度的问题：在 3W 预测任务中，FEMA 比起其他不使用正则项的方法都效果要好。当 MAE 和 RMSE 越小的时候，模型越好。

	MAS 数据		Weibo 数据	
	MAE	RMSE	MAE	RMSE
FEMA	**0.893**	**1.215**	**0.954**	**1.437**
EMA	0.909	1.466	0.986	1.698
DTA[101]	0.950	1.556	1.105	1.889
HOSVD[280]	1.047	1.618	1.220	2.054
CP[103]	1.055	1.612	1.243	2.117

表 3.10　带有灵活正则项的方法 FEMA 在 3W 预测任务中能够达到最小的 MAE 和 RMSE（$t = 10$ 和 $\alpha_t = 80\%$）。如果 MAE 和 RMSE 越小，模型越好。

可以看到 FEMA 比起其他方法中最好的结果还是要好的。FEMA 采用论文合作关系来约束研究人员矩阵，合作关系网络是一个能够反映出研究机构网络的富含信息的知识网络。所以也能够很好的反映学术论文关键字的分布。如果研究人员 -研究机构 -关键字张量过于稀疏，那么学习合作关系矩阵就能够更好的来理解和预测研究人员的学术行为。相通的是，在社交媒体中，FEMA 能够从社交关系信息中学到源用户和目标用户的分组信息。通常情况下如果 u 已经关注用户 v，那么 u 就更有可能会在自己的微博中提到 v。因此，学习社交信息是能够帮助系统更好的预测用户的微博提及行为的。

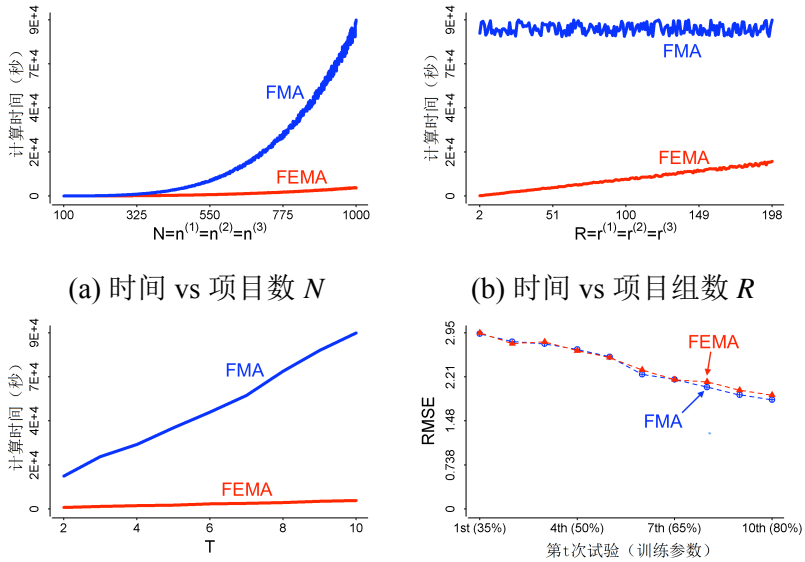
接下来评测 FEMA 算法的运行效率。FEMA 算法的运行速度受到以下三方面的影响，于是可以在测试中有如下假定：(1) 每个维度的项目数量，也就是张量的



(a) MAS 数据上的准确率 -召回率　(b) Weibo 数据上的准确率 -召回率

图 3.18　使用灵活正则项的 FEMA 在 3W 预测任务上能够达到最好的效果，准确率和召回率越大，模型的推荐效果越好。

大小 $N = n^{(1)} = n^{(2)} = n^{(3)}$，(2) 每个维度的项目组数量 $R = r^{(1)} = r^{(2)} = r^{(3)}$，(3) 张量的增量个数 T。方便起见，让每个维度的项目数量和项目组数量相等。还要观察的是 FEMA 比起 FMA 来说在节省时间的情况下会有多少精确度损失。



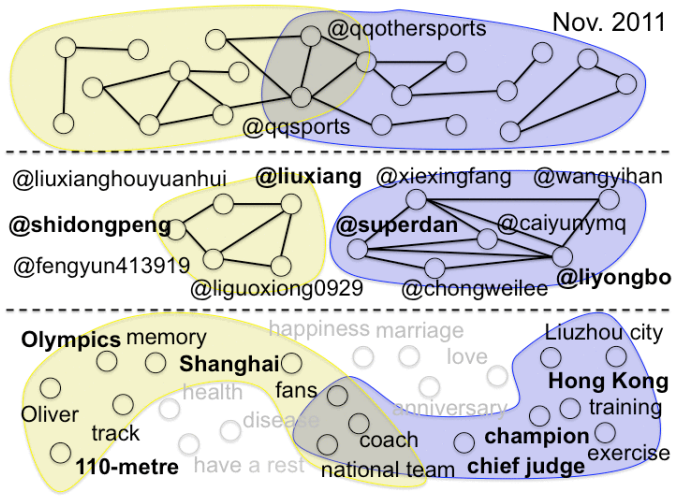
(a) 时间 vs 项目数 N　(b) 时间 vs 项目组数 R

(c) 时间 vs 增量数 T　(d) 精确度损失很小

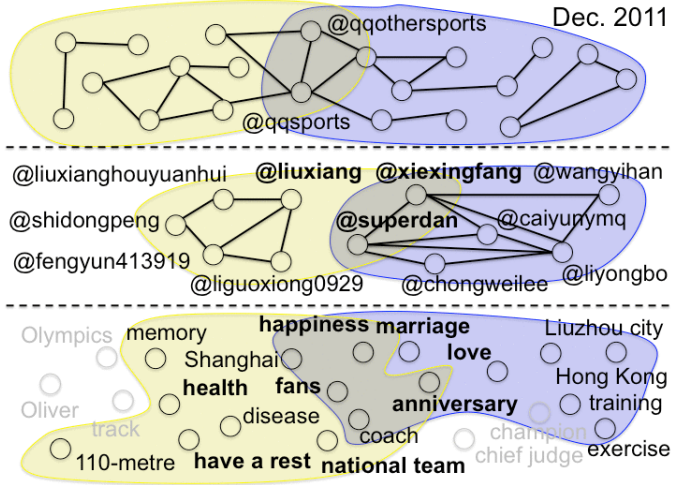
图 3.19　FEMA 比起 FMA 来说，在损失很小的精确度的同时，节省了大量的时间。

图3.19(a) 给出了随着每个维度的项目数 N 的增加，FEMA 比起 FMA 来说能节省多少时间。随机从 Weibo 数据中选取 $N \times N \times N$ 大小的张量，其中 $N = 100, \ldots, 1000$, 这样张量的密度是稳定的。然后默认 $R = 50$ 和 $T = 10$。FEMA 的运行时间比起 FMA 来说增长的非常慢。当 FMA 需要 25 个小时（超过 1 天）时，FEMA 只需要 51 分钟（不到 1 小时）。图3.19(b) 给出了每个维度的项目组数 R 从 2 增长到 100 时的运行时间变化。用 $1000 \times 1000 \times 1000$ 的采样张量，并且让 T 取 10。虽然 FEMA 的运行时间是正比于 R 的，但比起 FMA 来说还是少很多。图3.19(c) 显示运行时间是与增量数量 T 成线性关系的。所以知道动态分析算法

FEMA 能够处理稀疏的增量张量来更新映射矩阵，这对于张量分解来说节省了大量的时间。图3.19(d) 中用 $1000 \times 1000 \times 1000$ 上的采样张量，进行 3W 预测任务检查了 FEMA 算法的损失。可以看到 FMA 能够比 FEMA 得到更小的 RMSE，但是差别相当小。也就是说虽然公式中略去了高阶项，但高阶项所占的权重是非常小的，可以在实际应用中被忽略。



(a) 2011 年 11 月，刘翔的粉丝通过 “@ 刘翔” 谈论其 2004 年得到的奥运冠军，而林丹的粉丝通过 “@ 林丹” 讨论刚刚在香港结束的羽毛球赛中林丹夺得的冠军。

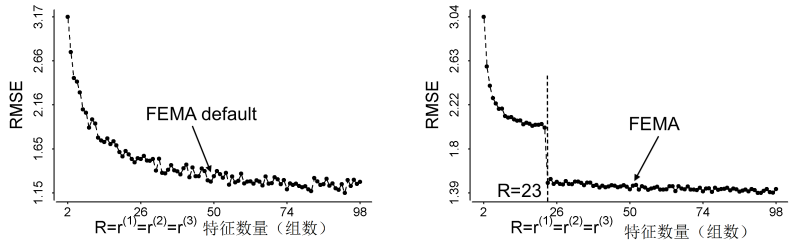


(b) 2011 年 12 月，刘翔的粉丝通过 “@ 刘翔” 来关心他最近谈及的病情，而林丹和谢杏芳的粉丝通过 “@ 林丹” 和 “@ 谢杏芳” 来庆祝他们的第一个结婚纪念日。

图 3.20　微博数据中的提及行为展现出的动态模式：FEMA 能够捕捉到在中国的跨栏运动迷和羽毛球运动迷通过 “@” 与他们偶像互动时的话题变化。

FEMA 能够从学术研究行为和微博提及行为中发掘出有意思的行为模式。在之前的图3.5中已经给出了韩家炜教授和他的研究组所做的科研工作的动态变化。

相似地，在图3.20中给出了 Weibo 数据中的发现。这个数据中存在三个维度：源用户，目标用户（被提及用户）和微博词汇。图3.20(a) 展现出左边的黄色是 2004 年雅典奥运会的 110 米栏冠军刘翔 (@liuxiang) 的粉丝群，运动员以及关于 110 米栏和跑步运动员的词汇。右边的蓝色是羽毛球迷，中国知名的羽毛球运动员和关于羽毛球的词汇。2011 年 11 月，刘翔的粉丝通过 “@ 刘翔”、“@ 史冬鹏” 与刘翔、史冬鹏 (@shidongpeng)，谈论刘翔于 2004 年荣获奥运冠军，谈论他们最美好的回忆。同时，中国羽毛球队刚刚结束了在香港的比赛，著名运动员林丹 (@superdan) 的粉丝通过 “@ 林丹” 讨论刚刚在香港结束的羽毛球赛中林丹夺得的冠军。图3.20(b) 中展现了在 2011 年 12 月这两组粉丝群的动态行为模式。刘翔跟大家谈起病情，于是刘翔的粉丝通过 “@ 刘翔” 来关心他最近的健康状况并给他祝福。而林丹发布了微博说道这是他和妻子、也是另一位著名的羽毛球运动员谢杏芳 (@xiexingfang) 的第一个结婚纪念日。所以可以看到 “爱情”，“结婚” 而不再是 “训练” 和 “运动”。要注意的是@qqsports 和@qqothersports这两个账号在源用户中突显出来，林丹和谢杏芳的权重在目标用户中凸现出来。能够看到这两个公众账号（“腾讯体育” 和 “腾讯全体育”）乃至刘翔的一部分粉丝都给林丹和谢杏芳表示祝贺。

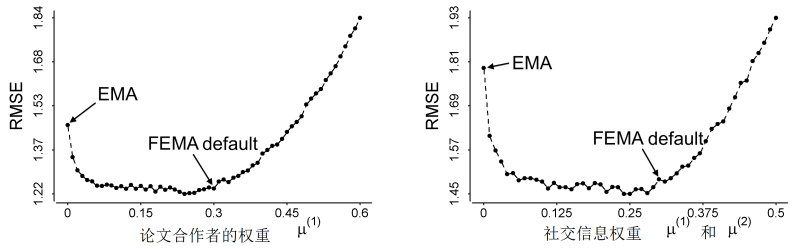


(a) 科研行为 MAS 数据　　(b) 微博提及行为 Weibo 数据

图 3.21　FEMA 给每个维度的项目组数量默认值为 50，随着 R 从 2 增加到 100，RMSE 逐步减小。在 Weibo 数据中，当 $R = 23$ 时，RMSE 减小得很迅速。

接下来讨论方法中设置的参数，一个是项目组的数量 R，另一个是正则化项的权重 μ。在两大数据集上，如图3.21所示，从 2 到 100 改变每一个维度上的项目组的数量 R，观察到 RMSE 会逐步减小；在 R 达到 30 以上的时候，RMSE 已经达到最低点。在微博数据中观察到当 $R = 23$ 时，RMSE 减小得非常快。如之前所说，FEMA 是用其他特征向量通过权重乘积再累加来更新特征向量。当 R 小于 23 的时候，如果在测试集合中存在第 23 个类别的项目时，在所有的特征向量中，这个项目的值都会是 0. 所以 R 达到 23 后，可以很好地估计这个项目的值。默认项目组数量 R 为 50。这就是在精确度（更低的 RMSE）和计算速度之间的权衡。

接着如图3.22所示，改变 FEMA 算法中的正则项稀疏 μ 从 0 到 1。当 $\mu = 0$



(a) 科研行为 MAS 数据　　(b) 微博提及行为 Weibo 数据

图 3.22　FEMA 设置正则项的参数 μ 为 0.3，而且还发现 FEMA 对于 μ 值并不敏感。

时，FEMA 就是 EMA，并不使用灵活的正则项；当 $\mu = 1$ 时，FEMA 只用正则项，但不用张量中的行为知识。在 MAS 数据和 Weibo 数据中可以看到当 μ 从 0.1 变化到 0.3 时，RMSE 先降低到谷底；接着当 μ 从 0.3 开始增加时，RMSE 开始增加。FEMA 算法对于 μ 的权重不敏感。为了方便，设置权重 μ 为 0.3。由此知道既从稀疏的高维张量中，也从灵活的正则项中学习行为数据，能够更好地理解人类行为，从而得到更好的行为建模方法。

3.4　本章小结

本章首先提出基于社交上下文因素（个人兴趣爱好和社交影响）的社会化推荐模型 ContextMF。在两个大规模真实社交媒体数据集上做充分实验，实验证明社交上下文信息能够在社交媒体数据集快速提升推荐效果：预测准确度分别有 24.2% 和 20.7% 的提升，排名评测指标 Precision@K 有 21.7% 和 12.3% 的提升。此外，提出的算法是对社交媒体来说普适的、含增量处理部分的，能够很容易地迁移到真实推荐系统中。

另一方面，本章还给出了新颖的基于时空上下文的行为预测方法 FEMA，换言之，是基于张量分解模型的多面性动态分析方法进行行为预测和模式挖掘。这种模型能够使用灵活的正则项来削减数据稀疏度问题，并给出近似算法来快速处理增量张量数据，还给出了理论保障。在两大真实数据集上的实验结果证实，FEMA 在完成行为预测时是有效且高效的。这个方法能够支持行为预测和模式发现的应用做到实时分析。

第 4 章　跨域和跨平台行为的迁移学习算法

本章从跨域性和跨平台性的角度介绍社交媒体用户行为的迁移学习算法。首先介绍单一平台上跨域行为预测的迁移学习算法，接着介绍跨平台行为预测的迁移学习算法，第三小节介绍性能评测结果，并在最后小结本章内容。

4.1　单一平台跨域行为预测的迁移学习算法

本节介绍单一社交媒体平台上用户行为的跨域性，并给出迁移学习算法。内容包括引言、相关工作、以社交纽带桥接多域的迁移性分析以及社交媒体重构思想和跨域混合随机漫步算法。

4.1.1　本节引言

社交媒体用户创造出各种类型的信息，如消息（微博和 Facebook 消息）、视频、用户标签和兴趣群组，形成了社交媒体中多种多样的信息域。通过社交域传播开的各种信息造成了严重的信息过载问题。大多数已经存在的消息推荐系统会受到很严重的目标域（比如微博域）数据稀疏度和冷启动问题影响。一种被广泛使用的基于矩阵分解推荐模型能够把某种信息域里的用户行为特征化[10,21]。然而在社交媒体上多种多样的信息都并不是与单一用户的性格和兴趣独立相关的。例如，用户会去阅读从他们社区发来的消息；用户会编辑与其好友相似的社交标签；用户会去观看有社交关系的人上传的视频。社交域里丰富的社交关系是信息能够被采纳和传播的本因。另一种解决方案是去学习直接关联的多种信息，比如音乐专辑与标签之间的关系[18]，网页和搜索内容之间的关系[242]。但是这种思路并不能够被应用到社交媒体上多个不直接相连的信息域上，比如微博、图片、推荐来的视频以及表达用户身份和兴趣的社交标签，它们两两都没有自然联系。这些信息域紧密地与社交域中大量用户相连接，不同信息域共同反映了用户兴趣和用户之间纽带强度。

在社交媒体众多的域中，社交域所蕴含的丰富社交关系形成了一个带权图，这个用户节点之间的图中的链接被叫做“域内链接”。社交域中的链接扮演着间接连接其他信息域中所有链接的重要角色，使得整个网络成为以用户图为中心的星状图。与此同时，所有信息域里的内容通过用户产生、采纳和传播，形成了“跨域链接”。图4.1展现出本工作重新勾画的星状社交网络结构。

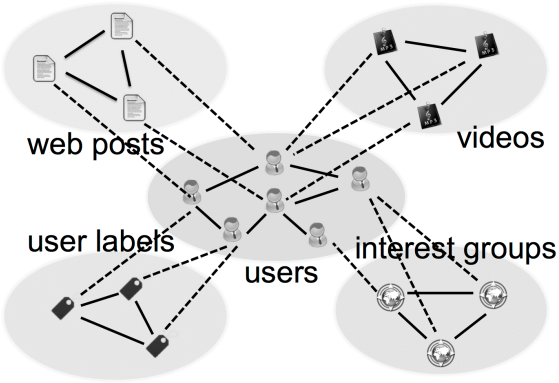


图 4.1　用混合星状图来重新为社交媒体建模：在社交域的周围连接着四个信息域。

位于社交域核心的用户纽带强度通常是指同质性[232]、社交影响力[30,44,51]、社交信任[56,57]，这是社交媒体中非常重要的因素。如果用户之间有相似特点，那么更有可能有较强纽带关系。然而，过去的工作中只能够用单一类型的跨域链接（在社交域和某一个信息域之间）来估计用户纽带强度，因为缺乏足够的信息而不够准确。跨域链接实际上可以从不同层面反映用户的特点。例如，从社交用户到某则关于 iPhone 的微博之间的跨域链接反映了用户对于 iPhone 短暂兴趣；而用户所贴有的 “果粉” 标签则能反映出其对 iPhone 的长久兴趣。有越多的辅助信息域中的知识，就越能更准确地预测用户之间的纽带强度，就能更好地实现推荐，尤其是在目标信息域极端稀疏。当一个用户和他的朋友有同样的社交标签的时候，可以增强他们之间的社交纽带，那么他们更容易转发相似的微博。

由此可以清晰地知道迁移多个关系域中信息的过程应当关注于社交域中纽带强度的更新。本工作从富足的辅助信息域中借助重要的用户关系图作为媒介，来迁移知识帮助稀疏目标域中行为的预测。但是协同地融合多个关系域的知识来深度发现可迁移知识，由此来消除单一信息域中数据的稀疏度问题和用户冷启动问题，是相当有挑战性的，因为

- 这些信息域有多种关系。除了社交域和信息域之间的跨域链接，还有在每个信息域中丰富的域内链接。例如，在社交域中，用户用社交关系相互连接；在微博域和标签域中，微博和标签都分别使用语义相似度连接起来；在兴趣群组域中，群组是用共同用户信息相互连接。如何有效地利用丰富的关系链接非常有挑战性。
- 信息域是多元异质（heterogeneous）的。多元异质性在信息排序问题中很有挑战性[244]，尤其是在社会化推荐中是更严重的。一些域内链接是有向的：微博上用户关系是有向链接。一些域内链接是无向的：微博信息域中的语义

相似度链接是无向的。一些域间链接是带符号的：用户到微博域的链接就是用户采纳或是拒绝微博信息的行为，是分正负号的。一些域间链接是不带符号的：用户添加社交标签的链接是只有正向信息的，而负向信息是难以推断的。如何从不同的多元异质信息域迁移知识会给方法的有效性提出挑战。

- 这些信息域是稀疏的，而且稀疏程度是不同的。数据稀疏是由百万级用户和信息，以及用户有限的注意力和时间的事实造成的。这对于如何充分利用可使用的信息提出挑战。
- 信息域中的信息有不同强度的迁移能力。过去的迁移学习算法会学习域间的关联性，但还没有工作分析过如何把迁移能力高的信息从辅助信息域中挑选出来迁移学习。

为了解决这样的问题，本工作中提出一种新颖的“混合随机漫步”（Hybrid Random Walk，简称 HRW）算法来从星状图种迁移学习知识为社会化推荐服务。这是一个基础性的而又实战性的研究问题，需要合理的解决方案。HRW 算法尝试估计每一对社交域中的用户和信息域中的内容之间的关联程度。用户之间的关联程度代表了用户之间的纽带强度。信息之间的关联程度代表了它们之间的语义相似度。跨域链接的关联程度代表了用户有多大可能采纳或是拒绝一个信息。混合随机漫步方法从多个关联域中融合这些信息来消除数据稀疏度问题和冷启动问题。值得强调一下本工作的贡献点：

- 提出了一个全新的方法来从社交媒体中的多个关系域，融合富含有向/无向、域内/域间、有符号/无符号链接的多元异质图迁移知识。混合随机漫步算法能够通过学习富足域内信息与目标域的一致性，自动选取高迁移能力的信息。这个方法能够广泛自然地使用在基于图的应用中，例如社交媒体，信息网络和生物信息网中。
- 大规模社交数据集上的充分实验证实混合随机漫步算法使得推荐效果有很大的提升。不活跃的或是新用户对于推荐系统来说是最为脆弱的一环，需要花费更多的精力来分析。实验展示了社交标签在提升推荐效果中的重要性，表明用多个信息源的知识（包括社交域和多个信息域）能够很好地解决数据稀疏度和用户冷启动问题。

4.1.2　相关工作

这里介绍近几年来的相关工作，并讨论本文所提出的研究工作的独特性。协同过滤技术已经为多种推荐系统都提供了核心框架[12,23,33]。基于概率矩阵分解模型[11]，同时分解社交信任关系和用户采纳信息行为的推荐算法被提出[17,21]。但是在解决信息过载问题上，协同过滤算法往往要面临数据稀疏度高的问题，因为数

据中缺乏足够的用户和信息的交互知识。

新的推荐场景预示着给出超出用户 -信息矩阵的更多新的知识能够提升效果[16,124,140,154]。研究者尝试利用评分向量作为媒介来融合多个信息域的知识[122]，或是用聚类的特征因子模型来实现跨域推荐[145]，以及用张量分解模型来推荐用户，电影/书籍和它们的标签[144]。然而，社会化推荐系统与普通的电影/书籍推荐系统是不同的：是社交关系推动了信息的传播和采纳[28,29]，所以只有考虑到用户与用户之间的社交纽带强度，社会化推荐系统才能更好地理解用户转发、分享行为的目的。还有研究者提出矩阵分解模型同时考虑用户的周边信息以及他们的社交关系来决定他们之间的相似度[27]。Facebook 也在尝试用跨域数据来提升他们的推荐系统[148]。Sedhain 等人证明了用户的周边信息对于 Facebook 的推荐系统来说是重要的[31]。用越多越复杂的周边信息，富含用户行为数据的辅助域，包括社交标签、分享视频、加入兴趣群组应该要被混合进一个比起矩阵分解模型来说，更加关系化，如随机漫步一样的模型[73]。

Adomavicius 和 Tuzhilin 调研了协同过滤技术、基于内容的推荐技术和混合推荐方法[8]。在他们的工作中预测到辅助的信息会在未来的推荐系统中发挥重要的作用。迁移学习是能够利用辅助信息域中的知识的方法[129,130,138,139,147,155,156]。协同的迁移学习方式能够把 MovieLens 上的数据迁移学习到 Netflix 数据集中降低其推荐电影中的稀疏度问题[131]。而且电影和书籍也可以互相迁移学习，提升预测评分值[125,126]。近年来提出了一种概率矩阵分子模型能够处理不同知识迁移状况下处理稀疏数据的方法[153]。通常迁移学习方法利用一致性的个人兴趣特征来连接两个不同领域的用户节点。然而，在社交媒体中，社交关系和人与人之间的纽带强度是桥接两个信息域的关键因素。与过去的迁移学习算法完全不同的是，本工作重新把社交媒体构建成一个星状图。

随机漫步算法的理念被广泛使用在推荐系统中。随机漫步算法在使用辅助信息时非常有效[73]。Tong 提出了高效快速的随机漫步算法实现[160]。ItemRank 是按照用户兴趣爱好来对产品排序的随机漫步算法[9]。TrustWalker 融合了基于信用和协同过滤的方法，定义和测量了随机漫步模型所给出的推荐概率[57]。一种随机漫步算法能够同时使用正向和负向评论信息以确保收敛[73]。然而，为了解决社会化推荐问题，不同类型的信息和社交关系通常能够形成高阶星状的多元异质网络[158,159]。本工作中给出在复杂图上的随机漫步算法来从富足的辅助信息域迁移知识到目标域的预测。最大的不同就是本文提出的模型使用了社交纽带信息作为最基础的桥梁信息来连接社交媒体中的两个不同的信息域。

4.1.3　以社交纽带桥接多域的迁移性分析

本节首先介绍含有多个信息域的大规模社交媒体数据集，然后证实从辅助信息域到目标域做迁移学习时迁移能力的存在。数据集是 2011 年 1 月份从中国的 Twitter，即腾讯微博上抓取下来的。工作中抓取了有至少一个社交标签的用户的数据。然而网站要求用户至多有 10 个社交标签，平均每个用户有 5.3 个社交标签。平均每个用户转发过 12.8 个微博。数据集中没有滤除任何的社交关系，每个用户有 14.2 个社交关系。

表4.1总结了数据集的统计情况。工作中使用长达 5 分钟的时间窗来获取负向链接：如果一个用户在 5 分钟内有两个采纳信息的行为（转发微博），可以假设用户在这个时间窗内收到的信息都被阅读过但拒绝了。除去这两个正向的用户采纳微博链接，还有负向的拒绝微博链接。微博域和社交标签域都非常稀疏但是稀疏程度不同。微博域的正向稀疏度和负向稀疏度分别是 4.2×10^{-6} 和 7.8×10^{-6}，而社交标签域的密度为 9.3×10^{-4}，几乎是 100 倍大。

信息域	信息	跨域链接		域内链接
		采纳 (+)	拒绝 (–)	无向
用户	1,427,214	-	-	20,240,902
微博消息	3,023,609	18,249,207	33,608,036	-
社交标签	5,715	7,604,679	-	-

表 4.1　数据集及其描述

图4.2展现了用户采纳微博的行为分布，用户采纳社交标签的行为分布和用户的社交关系分布。图中看到非常顺滑的幂律分布，说明数据集中没有异常情况。

如图4.3所示，真实社交媒体可以被刻画成二阶混合星状图。这不同于传统不包含域内关联信息的星状图[158] 而混合图中既包含域内链接，也包含域间链接。

表4.2总结了本章节中的符号和含义来表示图4.3中的五个子图：

- $\mathcal{G}^{(\mathcal{U})}=\{\mathcal{U},\mathcal{E}^{(\mathcal{U})}\}$，其中 $\mathcal{E}^{(\mathcal{U})}$ 表示 $\mathcal{U}$ 中的点之间的链接集合；
- $\mathcal{G}^{(\mathcal{P})}=\{\mathcal{P},\mathcal{E}^{(\mathcal{P})}\}$，其中 $\mathcal{E}^{(\mathcal{P})}$ 表示 $\mathcal{P}$ 中的点之间的链接集合；
- $\mathcal{G}^{(\mathcal{T})}=\{\mathcal{T},\mathcal{E}^{(\mathcal{T})}\}$，其中 $\mathcal{E}^{(\mathcal{T})}$ 表示 $\mathcal{T}$ 中的点之间的链接集合；
- $\mathcal{G}^{(\mathcal{UP})}=\{\mathcal{U}\bigcup\mathcal{P},\mathcal{E}^{(\mathcal{UP})}\}$，其中 $\mathcal{E}^{(\mathcal{UP})}$ 表示 $\mathcal{U}$ 和 $\mathcal{P}$ 中的点之间的链接集合；
- $\mathcal{G}^{(\mathcal{UT})}=\{\mathcal{U}\bigcup\mathcal{T},\mathcal{E}^{(\mathcal{UT})}\}$，其中 $\mathcal{E}^{(\mathcal{UT})}$ 表示 $\mathcal{U}$ 和 $\mathcal{T}$ 中的点之间的链接集合。

$\mathcal{G}^{(\mathcal{U})}$ 中的用户关系，也就是 u_i 到 u_j 的关联强度是

$$w_{ij}^{(\mathcal{U})}=\begin{cases}1 & \text{如果 } u_i \text{ 是 } u_j \text{ 的好友或是关注 } u_j\\ 0 & \text{否则}\end{cases}$$

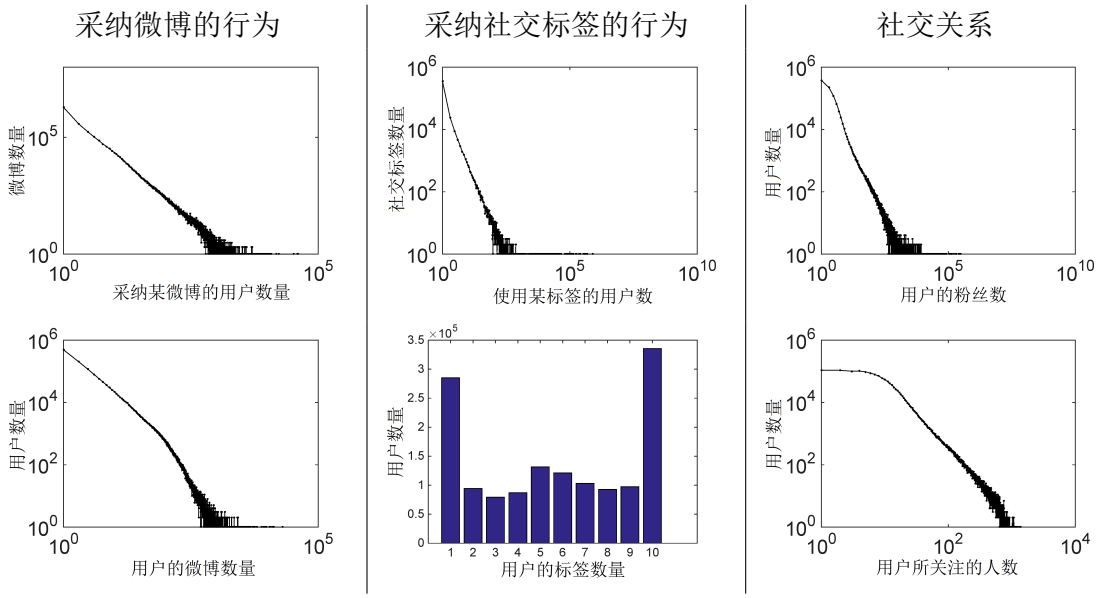


图 4.2　数据集的数据分布呈现顺滑的幂律分布：数据集中没有异常情况。另外注意，用户通常会添加 1 个或者（最多）10 个社交标签。

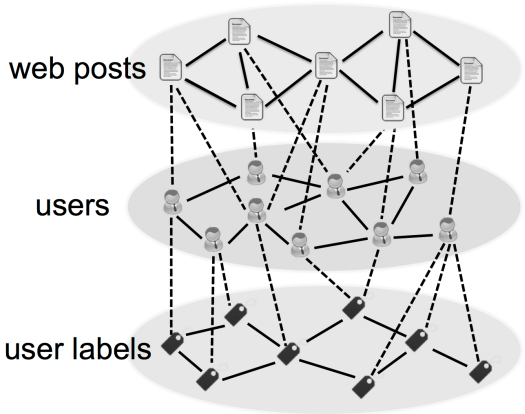


图 4.3　将社交媒体数据表征为二阶混合星状图：其中包含用户之间、微博之间、标签之间的域内链接，和用户与微博之间、用户与标签之间的域间链接。

用 TF-IDF 构建矩阵 $\mathbf{B}$ 来表征每一个微博消息 $\mathrm{b}_i = [b_{i1}, \cdots, b_{ik}, \cdots, b_{iK}]^\top$，其中 K 是词库大小；然后用如下公式计算微博 b_i 和 b_j 之间的语义相似度，来作为 $\mathcal{P}$ 中的消息关联度：

$$w_{ij}^{(\mathcal{P})} = \frac{\sum_k b_{ik} b_{jk}}{\sqrt{\sum_k {b_{ik}}^2} \sqrt{\sum_k {b_{jk}}^2}} \tag{4-1}$$

用 Jaccard 相似度来描述社交标签的关联度。假设标签 t_i 和 t_j 分别在 c_i 个消息和

符号	含义
$u_i; \mathcal{U} = \{u_1, u_2, \cdots, u_m\}$	第 i 个用户; 用户集合
$p_i; \mathcal{P} = \{p_1, p_2, \cdots, p_n\}$	第 i 条微博; 微博集合
$t_i; \mathcal{T} = \{t_1, t_2, \cdots, t_l\}$	第 i 个用户标签; 用户标签集合
d_{ij}	第 i 个域中的第 j 个信息
$\mathcal{D}_i = \{d_{i1}, d_{i2}, \cdots, d_{i\|\mathcal{D}_i\|}\}$	第 i 个域的信息集合
$\mathcal{D} = \{D_1, D_2, \cdots, D_N\}$	信息域集合

表 4.2　本章节的符号和含义

c_j 个消息中作为词汇存在。那么语义相似度可以计算为

$$w_{ij}^{(\mathcal{T})} = \frac{c_{ij}}{c_i + c_j - c_{ij}} \tag{4-2}$$

这样就得到了三个相似度矩阵 $\mathbf{W}^{(\mathcal{U})} = \{w_{ij}^{(\mathcal{U})}\}$，$\mathbf{W}^{(\mathcal{P})} = \{w_{ij}^{(\mathcal{P})}\}$ 和 $\mathbf{W}^{(\mathcal{T})} = \{w_{ij}^{(\mathcal{T})}\}$ 来对三个域内子图中的边权重进行编码。

模型中还有两个域间子图 $\mathcal{G}^{(\mathcal{UP})}$ 和 $\mathcal{G}^{(\mathcal{UT})}$，它们的边权重需要被计算。由于微博可以被转发或者忽略，但是用户标签只能被编辑，所以在用户、微博的域间既存在正向和负向链接，在用户、标签的域间只存在正向链接。这些链接可以被表示为 $e_{ij}^{(\mathcal{UP})}$ and $e_{ij}^{(\mathcal{UT})}$，权重表示如下：

$$w_{ij}^{(\mathcal{UP})^+} = \begin{cases} 1 & \text{如果 } u_i \text{ 转发微博 } \rho_j \\ 0 & \text{否则} \end{cases}$$

$$w_{ij}^{(\mathcal{UP})^-} = \begin{cases} 1 & \text{如果 } u_i \text{ 忽略微博 } \rho_j \\ 0 & \text{否则} \end{cases}$$

$$w_{ij}^{(\mathcal{UT})^+} = \begin{cases} 1 & \text{如果 } u_i \text{ 采用标签 } t_j \\ 0 & \text{否则} \end{cases}$$

于是得到了三个域间权重矩阵 $\mathbf{W}^{(\mathcal{UP})^+} = \{w_{ij}^{(\mathcal{UP})^+}\}$，$\mathbf{W}^{(\mathcal{UP})^-} = \{w_{ij}^{(\mathcal{UP})^-}\}$ 和 $\mathbf{W}^{(\mathcal{UT})^+} = \{w_{ij}^{(\mathcal{UT})^+}\}$。

接下来用数据分析来证实

- 富足的社交标签域的知识是可以被迁移到预测微博域的目标行为，换句话说，用户采纳标签的行为是与用户转发微博行为存在一致性的；
- 用户采纳标签的行为也与社交域中用户与用户关系存在一致性的；
- 不是每一个标签都可以被迁移，也不是最受欢迎的标签最容易被迁移。

首先，定义社交标签 j 的 “流行度” 为多少个用户采用这个标签：

$$popularity(j) = \sum_i w_{ij}^{(\mathcal{UT})^+} \tag{4-3}$$

第二，定义了社交标签 j 的三种一致性，包括

- 与微博内容的一致性，即有标签 j 的用户两两之间采纳微博的平均相似度：

$$cons_{post}(j) = \frac{\sum_{i\neq k; w_{ij}^{(\mathcal{UT})^+}, w_{kj}^{(\mathcal{UT})^+}>0} \frac{\mathbf{w}_{i,:}^{(\mathcal{UP})^+} \mathbf{w}_{k,:}^{(\mathcal{UP})^+}}{\|\mathbf{w}_{i,:}^{(\mathcal{UP})^+}\| \|\mathbf{w}_{k,:}^{(\mathcal{UP})^+}\|}}{\sum_{i\neq k; w_{ij}^{(\mathcal{UT})^+}, w_{kj}^{(\mathcal{UT})^+}>0} 1} \tag{4-4}$$

- 与用户粉丝群的一致性，即有标签 j 的用户的粉丝群会有多相似：

$$cons_{follower}(j) = \frac{\sum_{i\neq k; w_{ij}^{(\mathcal{UT})^+}, w_{kj}^{(\mathcal{UT})^+}>0} \frac{\mathbf{w}_{:,i}^{(\mathcal{U})} \mathbf{w}_{:,k}^{(\mathcal{U})}}{\|\mathbf{w}_{:,i}^{(\mathcal{U})}\| \|\mathbf{w}_{:,k}^{(\mathcal{U})}\|}}{\sum_{i\neq k; w_{ij}^{(\mathcal{UT})^+}, w_{kj}^{(\mathcal{UT})^+}>0} 1} \tag{4-5}$$

- 与用户所关注的人的一致性，即有标签 j 的用户所关注人群相似度：

$$cons_{followee}(j) = \frac{\sum_{i\neq k; w_{ij}^{(\mathcal{UT})^+}, w_{kj}^{(\mathcal{UT})^+}>0} \frac{\mathbf{w}_{i,:}^{(\mathcal{U})} \mathbf{w}_{k,:}^{(\mathcal{U})}}{\|\mathbf{w}_{i,:}^{(\mathcal{U})}\| \|\mathbf{w}_{k,:}^{(\mathcal{U})}\|}}{\sum_{i\neq k; w_{ij}^{(\mathcal{UT})^+}, w_{kj}^{(\mathcal{UT})^+}>0} 1} \tag{4-6}$$

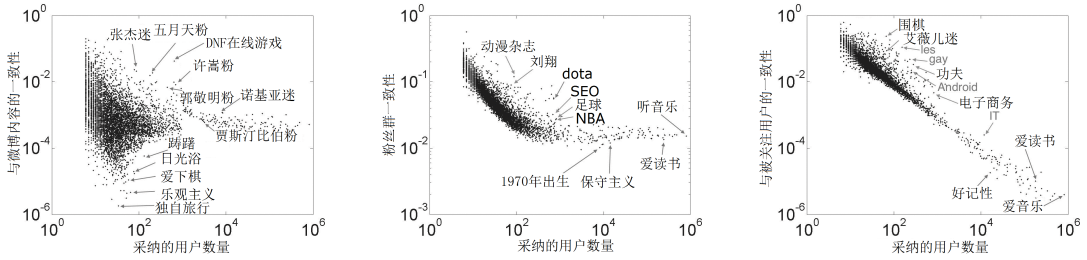


图 4.4　不是最流行的社交标签可以迁移最多的信息：不同的社交标签会在微博行为、粉丝群、关注的人群上有不同的一致性。令人惊奇的是，一致性往往会与流行度成反比。选取可迁移的用户标签是有重要意义的。

图4.4比较了用户标签的流行度和三种一致性。令人惊奇的是，一致性得分往往会和流行度成反比。细节上讨论起来，用户给自己添加如 “Nokia 粉” 和 “Justin Bieber 粉” 的用户标签的行为与他们转发微博的行为往往一致。但是一些精神层面的标签，比如 “独自旅行” 和 “乐观主义” 都很难保证一致性。有 “足球”、“NBA” 和 “SEO” 标签的用户往往有同样的粉丝，因为他们很活跃地去核朋友谈论这些内容。而有 “IT” 和 “电子商务” 标签的用户因为经常关注相关领域知名的人，所以他们往往有很相似的关注的人群。要注意的是最为流行的用户标签，比如 “听音乐” 和 “看电影” 无论在目标微博行为域上还是社交域上都没有强一致性。这些标

签并不能够丰富用户行为的知识，所以迁移能力非常弱。可以发现根据一致性强度来选取和采纳信息行为、形成社交关系行为一致的标签，对于知识迁移效果是非常有意义的。

4.1.4　跨域混合随机漫步算法

本节介绍用于社会化推荐的混合随机漫步算法。由于目标域中的数据稀疏度问题，传统的随机漫步算法（例如协同过滤的实现）并不能够准确地得到用户纽带强度来预测用户行为。幸运的是存在同一原因（同质性，信任和影响力）产生用户纽带的辅助信息域。核心想法是利用辅助信息域的富足知识来更好的描述用户纽带强度，从而更准确地预测用户在目标域中的行为。由此基于星状图设计了混合随机漫步算法。

该随机漫步算法用来预测在 $\mathcal{G}^{(\mathcal{UP})}$ 和 $\mathcal{G}^{(\mathcal{UT})}$ 中缺失的域间链接，其中包括域内的随机漫步和域间的随机漫步。用 $\mathcal{G}^{(\mathcal{U})}$，$\mathcal{G}^{(\mathcal{P})}$ 和 $\mathcal{G}^{(\mathcal{T})}$ 生成稳态分布[160]，反映了用户之间、微博之间和标签之间的本质上的相关性。对于标准的随机漫步模型来说，一个漫步者从第 i 个节点开始，以转移概率 $\mathbf{p}_i = \{p_{i1}, \cdots, p_{in}\}$（$p_{ii} = 1-\alpha$）迭代跳转到其他节点。到达稳态分布后，漫步者停留在第 j 个节点的概率可以从节点 j 到节点 i 之间的关联分数得来。特别的，转移概率矩阵可以计算为

$$\mathbf{P}^{(\mathcal{U})} = (\mathbf{D}^{(\mathcal{U})})^{-1}\mathbf{W}^{(\mathcal{U})} \tag{4-7}$$

$$\mathbf{P}^{(\mathcal{P})} = (\mathbf{D}^{(\mathcal{P})})^{-1}\mathbf{W}^{(\mathcal{P})} \tag{4-8}$$

$$\mathbf{P}^{(\mathcal{T})} = (\mathbf{D}^{(\mathcal{T})})^{-1}\mathbf{W}^{(\mathcal{T})} \tag{4-9}$$

其中用 $\mathbf{D}^{(\mathcal{UP})^+}$，$\mathbf{D}^{(\mathcal{UP})^-}$ 和 $\mathbf{D}^{(\mathcal{UT})^+}$ 来表征跨域链接的度数矩阵。最终的稳态分布概率矩阵通过迭代更新得到：

$$\mathbf{R}^{(\mathcal{U})}(t+1) = \alpha\mathbf{P}^{(\mathcal{U})}\mathbf{R}^{(\mathcal{U})}(t) + (1-\alpha)\mathbf{I} \tag{4-10}$$

$$\mathbf{R}^{(\mathcal{P})}(t+1) = \beta\mathbf{P}^{(\mathcal{P})}\mathbf{R}^{(\mathcal{P})}(t) + (1-\beta)\mathbf{I} \tag{4-11}$$

$$\mathbf{R}^{(\mathcal{T})}(t+1) = \gamma\mathbf{P}^{(\mathcal{T})}\mathbf{R}^{(\mathcal{T})}(t) + (1-\gamma)\mathbf{I} \tag{4-12}$$

其中 $\mathbf{R}^{(\mathcal{U})}(t)$，$\mathbf{R}^{(\mathcal{P})}(t)$，$\mathbf{R}^{(\mathcal{T})}(t)$，$\mathbf{R}^{(\mathcal{U})}(t+1)$，$\mathbf{R}^{(\mathcal{P})}(t+1)$ 和 $\mathbf{R}^{(\mathcal{T})}(t+1)$ 是在时刻 t 和 $t+1$ 的状态概率矩阵。而 $0 \le \alpha, \beta, \gamma \le 1$ 是漫步者是以多大概率离开当前的状态。很容易看到的是上述的迭代更新当 $t \to \infty$ 最终会收敛在一个稳态矩阵。

$$\mathbf{R}^{(\mathcal{U})} = (1-\alpha)(\mathbf{I} - \alpha\mathbf{P}^{(\mathcal{U})})^{-1} \tag{4-13}$$

$$\mathbf{R}^{(\mathcal{P})} = (1-\beta)(\mathbf{I} - \beta\mathbf{P}^{(\mathcal{P})})^{-1} \tag{4-14}$$

$$\mathbf{R}^{(\mathcal{T})} = (1-\gamma)(\mathbf{I}-\gamma\mathbf{P}^{(\mathcal{T})})^{-1} \tag{4-15}$$

对于跨域链接，计算如下的概率转移矩阵：

$$\mathbf{P}^{(\mathcal{UP})^+} = (\mathbf{D}^{(\mathcal{UP})^+})^{-1}\mathbf{W}^{(\mathcal{UP})^+} \tag{4-16}$$

$$\mathbf{P}^{(\mathcal{UP})^-} = (\mathbf{D}^{(\mathcal{UP})^-})^{-1}\mathbf{W}^{(\mathcal{UP})^-} \tag{4-17}$$

$$\mathbf{P}^{(\mathcal{UT})^+} = (\mathbf{D}^{(\mathcal{UT})^+})^{-1}\mathbf{W}^{(\mathcal{UT})^+} \tag{4-18}$$

其中矩阵元素 $p_{ij}^{(\mathcal{UP})^+}$ 和 $p_{ij}^{(\mathcal{UP})^-}$ 表示用户 u_i 采纳或是忽略微博 p_j 的转移概率。元素 $p_{ij}^{(\mathcal{UT})^+}$ 代表用户 u_i 是否采纳社交标签 t_j 的转移概率。可以同时学习每一对用户之间的相关系数 $\mathbf{R}^{(\mathcal{U})} = \{r_{ij}^{(\mathcal{U})}\}$，最终它反映了用户之间的真实纽带强度。元素 $r_{ij}^{(\mathcal{U})}$ 代表了漫步者从用户 u_i 到 u_j 的跳转概率。考虑上述的转移路径并估计 $p_{ij}^{(\mathcal{UP})^+}$，$p_{ij}^{(\mathcal{UP})^-}$，$p_{ij}^{(\mathcal{UT})^+}$ 和 $r_{ij}^{(\mathcal{U})}$ 这些在子图 $\mathcal{G}^{(\mathcal{UP})}$，$\mathcal{G}^{(\mathcal{UT})}$ 和 $\mathcal{G}^{(\mathcal{U})}$ 上漫步者一步跳转的转移概率：

$$p_{ij}^{(\mathcal{UP})^+} = \delta\sum_{u_k\in\mathcal{U}} r_{ik}^{(\mathcal{U})}p_{kj}^{(\mathcal{UP})^+} + (1-\delta)\sum_{p_k\in\mathcal{P}} p_{ik}^{(\mathcal{UP})^+}r_{kj}^{(\mathcal{P})} \tag{4-19}$$

$$p_{ij}^{(\mathcal{UP})^-} = \delta\sum_{u_k\in\mathcal{U}} r_{ik}^{(\mathcal{U})}p_{kj}^{(\mathcal{UP})^-} + (1-\delta)\sum_{p_k\in\mathcal{P}} p_{ik}^{(\mathcal{UP})^-}r_{kj}^{(\mathcal{P})} \tag{4-20}$$

$$p_{ij}^{(\mathcal{UT})^+} = \eta\sum_{u_k\in\mathcal{U}} r_{ik}^{(\mathcal{U})}p_{kj}^{(\mathcal{UT})^+} + (1-\eta)\sum_{t_k\in\mathcal{T}} p_{ik}^{(\mathcal{UT})^+}r_{kj}^{(\mathcal{T})} \tag{4-21}$$

$$\begin{aligned} r_{ij}^{(\mathcal{U})} = {} & \tau^{(\mathcal{P})}(\mu\sum_{p_k\in\mathcal{P}} p_{ik}^{(\mathcal{UP})^+}p_{jk}^{(\mathcal{UP})^+} + (1-\mu)\sum_{p_k\in\mathcal{P}} p_{ik}^{(\mathcal{UP})^-}p_{jk}^{(\mathcal{UP})^-}) \\ & +\tau^{(\mathcal{T})}\sum_{t_k\in\mathcal{T}} p_{ik}^{(\mathcal{UT})^+}p_{jk}^{(\mathcal{UT})^+} + \tau^{(\mathcal{U})}\sum_{u_k\in\mathcal{U}} r_{ik}^{(\mathcal{U})}r_{kj}^{(\mathcal{U})} \end{aligned} \tag{4-22}$$

其中 $0 \le \delta,\eta,\mu,\tau^{(\mathcal{P})},\tau^{(\mathcal{T})},\tau^{(\mathcal{U})} \le 1$ 是转移路径上的权衡参数。跨域转移概率矩阵需要考虑不同转移路径的权重系数。如图4.5所示，考虑两种路径来更新跨域转移概率矩阵。假设转移概率矩阵的更新会影响到域内转移概率矩阵。图4.6考虑了三种不同的转移路径。

图 4.5　域间转移概率矩阵的路径生成。

本工作中所提出的模型假设域间转移概率会影响到域内转移概率，因为用户的纽带强度受到 (1) 共同的微博，(2) 共同的社交标签和 (3) 共同的社交关系的影

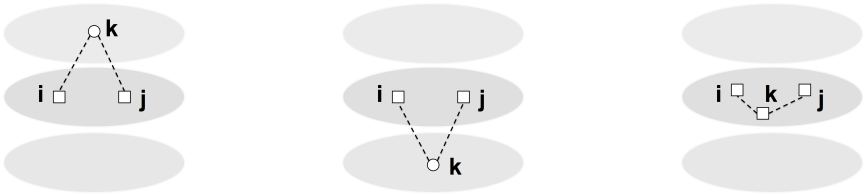


图 4.6　域内转移概率矩阵的路径生成；用户之间的纽带强度更为复杂。

响。域间链接（也就是采纳信息行为）是不会影响到信息域内部的信息之间的转移概率的。原因是同种信息的相互转移概率应该从它们的语义相似度中得到，而不以人的行为为转移。因此 HRW 算法只更新从用户到每一种信息的域间链接，以及人与人之间的域内链接，而不更新信息之间的域内链接。由此可以给出从时刻 t 到 $t+1$ 的转移概率更新的矩阵形式：

$$\mathbf{P}^{(\mathcal{UP})^+}(t+1) = \delta\mathbf{R}^{(\mathcal{U})}(t)\mathbf{P}^{(\mathcal{UP})^+}(t) + (1-\delta)\mathbf{P}^{(\mathcal{UP})^+}(t)\mathbf{R}^{(\mathcal{P})} \tag{4-23}$$

$$\mathbf{P}^{(\mathcal{UP})^-}(t+1) = \delta\mathbf{R}^{(\mathcal{U})}(t)\mathbf{P}^{(\mathcal{UP})^-}(t) + (1-\delta)\mathbf{P}^{(\mathcal{UP})^-}(t)\mathbf{R}^{(\mathcal{P})} \tag{4-24}$$

$$\mathbf{P}^{(\mathcal{UT})^+}(t+1) = \eta\mathbf{R}^{(\mathcal{U})}(t)\mathbf{P}^{(\mathcal{UT})^+}(t) + (1-\eta)\mathbf{P}^{(\mathcal{UT})^+}(t)\mathbf{R}^{(\mathcal{T})} \tag{4-25}$$

$$\begin{aligned}\mathbf{R}^{(\mathcal{U})}(t+1) = {} & \tau^{(\mathcal{P})}(\mu\mathbf{P}^{(\mathcal{UP})^+}(t)\mathbf{P}^{(\mathcal{UP})^+}(t)^T \\ & +(1-\mu)\mathbf{P}^{(\mathcal{UP})^-}(t)\mathbf{P}^{(\mathcal{UP})^-}(t)^T) \\ & +\tau^{(\mathcal{T})}\mathbf{P}^{(\mathcal{UT})^+}(t)\mathbf{P}^{(\mathcal{UT})^+}(t)^T + \tau^{(\mathcal{U})}\mathbf{R}^{(\mathcal{U})}(t)\mathbf{R}^{(\mathcal{U})}(t)^T\end{aligned} \tag{4-26}$$

在下一次随机漫步时，漫步者可以在图 $\mathcal{G}^{(\mathcal{U})}$，$\mathcal{G}^{(\mathcal{UP})}$ 和 $\mathcal{G}^{(\mathcal{UT})}$ 上计算转移概率矩阵 $\mathbf{R}^{(\mathcal{U})}$，$\mathbf{P}^{(\mathcal{UP})^+}$，$\mathbf{P}^{(\mathcal{UP})^-}$ 和 $\mathbf{P}^{(\mathcal{UT})^+}$。

算法4总结了基于二阶星状图的随机漫步算法的过程来预测采纳微博和采纳标签的行为。算法的空间复杂度为 $O(m^2+n^2+l^2+2m(n+l))$，时间复杂度是 $O((m^2+4m(n+l)+2(n^2+l^2))mT)$，其中 m，n 和 l 分别是用户数量、微博数量和标签数量，T 是算法迭代次数。通常矩阵都很稀疏，也就是 $m,n \gg l$，所以空间复杂度是 $O((m+n)^2)$，时间复杂度是 $O((m+n)ET)$，其中 E 是用户和微博之间链接的数量。

如之前讨论的，信息域中的不同信息具有不同强度迁移能力。本工作中采用广泛使用的（1770 年由 Jean Charles de Borda 提出的）投票法来选取可迁移的信息、提升迁移学习效果，称这种迁移能力选择性的 HRW 算法为 HRW-Borda。对于每一种特征，例如流行度、发帖一致性、粉丝一致性，关注者一致性，都给投票系统提供一种选取社交标签的结果。每一种排序结果都给每一个标签一定的分数，系统给出的总分数能够评价社交标签的迁移能力。工作中选取了最好的 $l_{top}=1,000$ 个迁移能力高的社交标签。

Algorithm 4 基于二阶星状图的随机漫步算法迭代预测采纳行为

Require: $0 \le \alpha,\beta,\gamma,\delta,\eta,\mu,\tau^{(\mathcal{P})},\tau^{(\mathcal{T})},\tau^{(\mathcal{U})} \le 1$

1: 构建子图 $\mathcal{G}^{(\mathcal{U})},\mathcal{G}^{(\mathcal{P})},\mathcal{G}^{(\mathcal{T})},\mathcal{G}^{(\mathcal{UP})},\mathcal{G}^{(\mathcal{UT})}$

2: 计算转移概率 $\mathbf{P}^{(\mathcal{U})}$, $\mathbf{P}^{(\mathcal{P})}$ 和 $\mathbf{P}^{(\mathcal{T})}$

3: 计算稳态分布 $\mathbf{R}^{(\mathcal{U})}$, $\mathbf{R}^{(\mathcal{P})}$ 和 $\mathbf{R}^{(\mathcal{T})}$

4: 初始化转移概率矩阵 $\mathbf{P}^{(\mathcal{UP})^+}(0)$, $\mathbf{P}^{(\mathcal{UP})^-}(0)$ 和 $\mathbf{P}^{(\mathcal{UT})^+}(0)$

5: **for** $t = 1 : T$ **do**

6: 计算用户纽带强度矩阵 $\mathbf{R}^{(\mathcal{U})}(t)$ 和转移概率矩阵 $\mathbf{P}^{(\mathcal{UP})^+}(t)$, $\mathbf{P}^{(\mathcal{UP})^-}(t)$ 和 $\mathbf{P}^{(\mathcal{UT})^+}(t)$

7: **end for**

8: **输出:** 最终的转移概率矩阵 $\mathbf{R}^{(\mathcal{U})}$, $\mathbf{P}^{(\mathcal{UP})^+}$, $\mathbf{P}^{(\mathcal{UP})^-}$ 和 $\mathbf{P}^{(\mathcal{UT})^+}$

假设和用户相关的信息域有两个：微博域和标签域。然而，在线社交媒体上有很多类型的 UGC（User Generated Content，用户产生内容），包括微博、社交标签、音乐和电影。二阶星状图并不足够描述所有的社交媒体内容。图4.1中已经给出了典型的混合星状图，其中包括四种不同的 UGC。那么就需要一个通用方法在多种多样的信息域上预测用户行为。而随机漫步方法很容易就可以推广到高阶条件下的。用下述标记来表示高阶混合星状图种的子图：

- $\mathcal{G}^{(\mathcal{U})} = \{\mathcal{U},\mathcal{E}^{(\mathcal{U})}\}$，其中 $\mathcal{E}^{(\mathcal{U})}$ 表示 $\mathcal{U}$ 中的用户节点之间的链接；
- $\mathcal{G}^{(\mathcal{D}_i)} = \{\mathcal{D}_i,\mathcal{E}^{(\mathcal{D}_i)}\}$，其中 $\mathcal{E}^{(\mathcal{D}_i)}$ 表示 $\mathcal{D}_i$（$i = 1,\cdots,N$）信息域中的点点之间的链接；
- $\mathcal{G}^{(\mathcal{UD}_i)} = \{\mathcal{U}\bigcup\mathcal{D}_i,\mathcal{E}^{(\mathcal{UD}_i)}\}$，其中 $\mathcal{E}^{(\mathcal{UD}_i)}$ 表示 $\mathcal{U}$ 和 $\mathcal{D}_i$（$i = 1,\cdots,N$）之间节点的域间链接。

用图 $\mathcal{G}^{(\mathcal{U})}$ 和 $\{\mathcal{G}^{(\mathcal{D}_i)}\}_{i=1}^N$ 中的信息来构建对应的边权矩阵 $\mathbf{W}^{(\mathcal{U})}$ 和 $\{\mathbf{W}^{(\mathcal{D}_i)}\}_{i=1}^N$。那么域内的转移概率矩阵可以表达为（$i = 1,\cdots,N$）：

$$\mathbf{P}^{(\mathcal{U})} = (\mathbf{D}^{(\mathcal{U})})^{-1}\mathbf{W}^{(\mathcal{U})} \tag{4-27}$$

$$\mathbf{P}^{(\mathcal{D}_i)} = (\mathbf{D}^{(\mathcal{D}_i)})^{-1}\mathbf{W}^{(\mathcal{D}_i)} \tag{4-28}$$

其中 $\mathbf{D}^{(\mathcal{U})}$ 和 $\{\mathbf{D}^{(\mathcal{D}_i)}\}_{i=1}^N$ 是从 $\mathbf{W}^{(\mathcal{U})}$ 和 $\{\mathbf{W}^{(\mathcal{D}_i)}\}_{i=1}^N$ 中得到的度数矩阵。最终的稳态概率矩阵可以迭代计算为

$$\mathbf{R}^{(\mathcal{U})}(t+1) = \alpha\mathbf{P}^{(\mathcal{U})}\mathbf{R}^{(\mathcal{U})}(t) + (1-\alpha)\mathbf{I} \tag{4-29}$$

$$\mathbf{R}^{(\mathcal{D}_i)}(t+1) = \beta_i\mathbf{P}^{(\mathcal{D}_i)}\mathbf{R}^{(\mathcal{D}_i)}(t) + (1-\beta_i)\mathbf{I} \tag{4-30}$$

其中 $i = 1,2,\cdots,N$, $0 \le \alpha,\beta_1,\cdots,\beta_N \le 1$。对于跨域图 $\{\mathcal{G}^{(\mathcal{UD}_i)}\}_{i=1}^N$ 来说，计算根据

其他信息域 $\{\mathcal{D}_i\}_{i=1}^N$ 中的用户交互信息，计算边权矩阵 $\{\mathbf{W}^{(\mathcal{UD}_i)}\}_{i=1}^N$。那么，跨域转移概率矩阵可以计算为（$i=1,\cdots,N$）：

$$\mathbf{P}^{(\mathcal{UD}_i)^+} = (\mathbf{D}^{(\mathcal{UD}_i)^+})^{-1}\mathbf{W}^{(\mathcal{UD}_i)^+} \tag{4-31}$$

$$\mathbf{P}^{(\mathcal{UD}_i)^-} = (\mathbf{D}^{(\mathcal{UD}_i)^-})^{-1}\mathbf{W}^{(\mathcal{UD}_i)^-} \tag{4-32}$$

当更新域间链接形成的概率转移矩阵时，依旧如图4.5 和图4.6所示来考虑转移路径，所以更新法则如下：

$$\mathbf{P}^{(\mathcal{UD}_i)^+}(t+1) = \delta_i\mathbf{R}^{(\mathcal{U})}(t)\mathbf{P}^{(\mathcal{UD}_i)^+}(t) + (1-\delta_i)\mathbf{P}^{(\mathcal{UD}_i)^+}(t)\mathbf{R}^{(\mathcal{D}_i)} \tag{4-33}$$

$$\mathbf{P}^{(\mathcal{UD}_i)^-}(t+1) = \delta_i\mathbf{R}^{(\mathcal{U})}(t)\mathbf{P}^{(\mathcal{UD}_i)^-}(t) + (1-\delta_i)\mathbf{P}^{(\mathcal{UD}_i)^-}(t)\mathbf{R}^{(\mathcal{D}_i)} \tag{4-34}$$

$$\begin{aligned}\mathbf{R}^{(\mathcal{U})}(t+1) = &\sum_{\mathcal{D}_i\in\mathcal{D}}\tau_i\mu_i\mathbf{P}^{(\mathcal{UD}_i)^+}(t)\mathbf{P}^{(\mathcal{UD}_i)^+}(t)^T \\ &+\sum_{\mathcal{D}_i\in\mathcal{D}}\tau_i(1-\mu_i)\mathbf{P}^{(\mathcal{UD}_i)^-}(t)\mathbf{P}^{(\mathcal{UD}_i)^-}(t)^T \\ &+\tau^{(\mathcal{U})}\mathbf{R}^{(\mathcal{U})}(t)\mathbf{R}^{(\mathcal{U})}(t)^T\end{aligned} \tag{4-35}$$

其中 $0\le\delta_i,\mu_i,\tau_i\le 1$ $(i=1,2,\cdots,N)$ 是权衡参数。对于不含有负向信息的信息域 $\mathcal{D}_i$ 来说，设置 $\mu_i=1$ 来更新 $\mathbf{R}^{(\mathcal{U})}$。

算法5总结了高阶星状图的随机漫步算法。其空间复杂度为 $O(m^2+2m\sum|\mathcal{D}_i|+\sum|\mathcal{D}_i|^2)$，时间复杂度为 $O((m^2+4m\sum|\mathcal{D}_i|+2\sum|\mathcal{D}_i|^2)mT)$, T 为迭代次数。

Algorithm 5 基于高阶星状图的随机漫步算法迭代预测采纳行为

Require: $0\le\alpha,\{\beta_i\}_{i=1}^N,\{\delta_i\}_{i=1}^N,\{\mu_i\}_{i=1}^N,\{\tau_i\}_{i=1}^N\le 1$

1: 构建 $\mathcal{G}^{(\mathcal{U})}$, $\{\mathcal{G}^{(\mathcal{D}_i)}\}_{i=1}^N$, $\{\mathcal{G}^{(U\mathcal{D}_i)}\}_{i=1}^N$
2: 计算 $\mathbf{P}^{(\mathcal{U})}$ 和 $\{\mathbf{P}^{(\mathcal{D}_i)}\}_{i=1}^N$
3: 计算 $\mathbf{R}^{(\mathcal{U})}$ 和 $\{\mathbf{R}^{(\mathcal{D}_i)}\}_{i=1}^N$
4: 初始化 $\{\mathbf{P}^{(\mathcal{UD}_i)^+}(0)\}_{i=1}^N$ 和 $\{\mathbf{P}^{(\mathcal{UD}_i)^-}(0)\}_{i=1}^N$
5: **for** $t=1:T$ **do**
6:　　计算 $\mathbf{R}^{(\mathcal{U})}(t)$, $\{\mathbf{P}^{(\mathcal{UD}_i)^+}(t)\}_{i=1}^N$ and $\{\mathbf{P}^{(\mathcal{UD}_i)^-}(t)\}_{i=1}^N$
7: **end for**
8: **输出:** 最终转移矩阵为 $\mathbf{R}^{(\mathcal{U})}$, $\{\mathbf{P}^{(\mathcal{UD}_i)^+}\}_{i=1}^N$ and $\{\mathbf{P}^{(\mathcal{UD}_i)^-}\}_{i=1}^N$

4.2　跨平台行为预测的迁移学习算法

本节介绍社交媒体用户行为的跨平台性，并给出迁移学习算法。内容包括引言、相关工作，以重合用户桥接多平台的迁移性分析以及跨平台半监督迁移学习

算法和增量处理算法。

4.2.1　本节引言

网络信息平台已经是用户获取知识的重要信息源。用户因为有广泛的、不同的信息需求衍生出不同种类的信息平台。通常情况下，人们会使用 YouTube 来观看视频，用 Flickr 浏览图片，用 Facebook 来分享社交消息。在不同平台上用户所采纳的内容往往有着外在或内在联系，以满足他们的不同层面需求。为了实现智能化满足用户需求的终极目标，根本方法是充分认知用户的需求。然而当前的信息平台或是独立存在，或是严重缺乏关注其关联关系。如何合理融合和桥接跨平台信息对于完成以人为本的信息需求任务非常重要，更广泛的说，是最大化在不同平台的大数据中潜在价值。本工作专注于跨平台的行为预测问题：如何基于其他平台的用户行为更好的预测目标平台的用户行为？对于每一对辅助平台和目标平台，可以定义问题如下：

定义 4.1 (跨平台行为预测)：**给定**：目标平台 P 中的行为数据（如电子商务、健康医疗等），辅助平台 Q 上的行为数据（如社交媒体、可穿戴设备等），平台 P 和 Q 之间的用户交集；**预测**：目标平台 P 上的缺失用户行为。

基于迁移学习方法的传统研究工作探索了跨域行为预测的问题。Codebook[125] 假设辅助平台和目标平台之间并没有重合用户，比如 Netflix 和 Movielens，但是它们共享用户 -信息的评分模式。Lin 等[153] 提出了因子分解模型 TPCF 来探索辅助域中不同类型的数据：(1) 用户对齐的数据，(2) 信息对齐的数据，以及 (3) 不对齐但与目标域同质的辅助数据。传统的工作假设用户之间或是完全对齐[83,153] 或是跨域、跨平台场景下完全没有关联[125,153]。然而真实情况是不同平台的用户存在重合。重合用户的数量往往不多，重合用户的联系少是因为统一 ID 体系的缺失。例如，可用的跨新浪微博（中国超过 3 亿用户的微博平台）和豆瓣（中国有 5000 万用户的电影、书籍、音乐评分平台）的用户所占百分比不到 1%。跨平台行为预测问题的难点就是如何结合这少量的重合用户有效桥接在不同平台上的用户行为。这个问题有如下的挑战：

- 高稀疏度：用户在一个平台上通常只能采纳一小部分信息。平均来说，一个豆瓣用户会从 50,000 个信息里给 60 个书籍，200 个电影和 100 个歌曲打分。在新浪微博上，每个用户只有 10,000 个社交标签中的平均 4.5 个，发布 100,000 个微博实体中的 5,000。
- 多元异构性：不同平台的行为数据通常是多元异构的，也就是有不同的信息，不同的链接和不同的评分规模。豆瓣用户给书籍、电影和音乐打分是

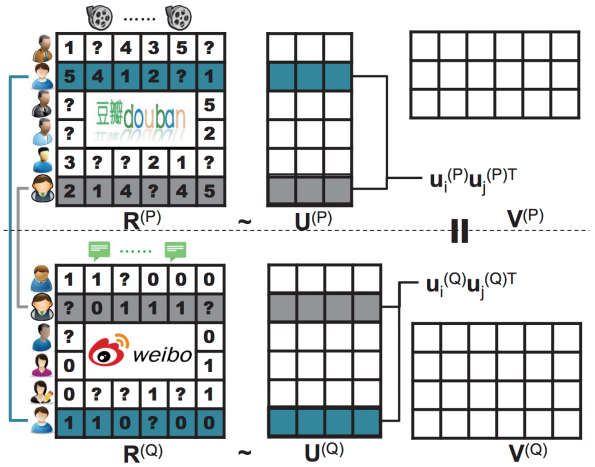


图 4.7　跨平台行为预测的半监督迁移学习方法 XPTrans：(1) 跨平台优化行为表征来解决单一平台稀疏度高的问题；(2) 用跨平台用户的相似性来约束两个平台的用户表征。

从 1 到 5 的，而微博上采纳社交标签是二元值，即 0 或 1，采纳微博实体的次数是非负整数值。社交媒体、电子商务、穿戴设备的信息往往是不同的：“喜欢” 的数目，评论的表情，跑步距离和血压值。

- 跨平台的不同表征：评分行为、转发行为和购物行为往往有不同的行为模式。给电影打分往往是看电影的类型、导演、演员等是否符合爱好，转发微博是看相关话题和社交影响。用户在不同平台的行为模式并不能用同样的特征空间来表示，这使得这一工作和传统工作非常不同。
- 部分重合的用户：两个平台之间自然的桥接关系就是它们重合的用户集合，因为这些人的兴趣爱好、品味和个性是在跨平台情况下也是一致的。如何充分利用这些跨平台重合用户来做好行为建模还是开放和有挑战性的问题。

本文为了解决上述问题，提出了新颖的基于矩阵因子化模型的半监督迁移学习方法 XPTrans。图4.7介绍了模型设计：首先，XPTrans 将包含电影、书籍、微博等多元异构的平台数据表示为带权/二元矩阵；第二，XPTrans 同时优化用户在不同平台的表征来解决高稀疏度问题；第三，同一个用户在不同平台上的表征是相似的，但并不完全相同。一个平台的用户表征空间和其他平台应该是不同的，XPTrans 采用重合用户相似度，而不是用户表征数值来约束重合用户的特征空间。工作中的假设是重合用户的相似度在跨平台环境下是一致的。

图4.8给出真实数据上的实验效果：XPTrans 通过迁移新浪微博的微博数据来学习豆瓣用户的电影评分行为。采用更多活跃的重合用户或是更多的重合行为数据，预测错误会持续降低，准确率会上升。这里证实了采用最活跃的 26% 重合用

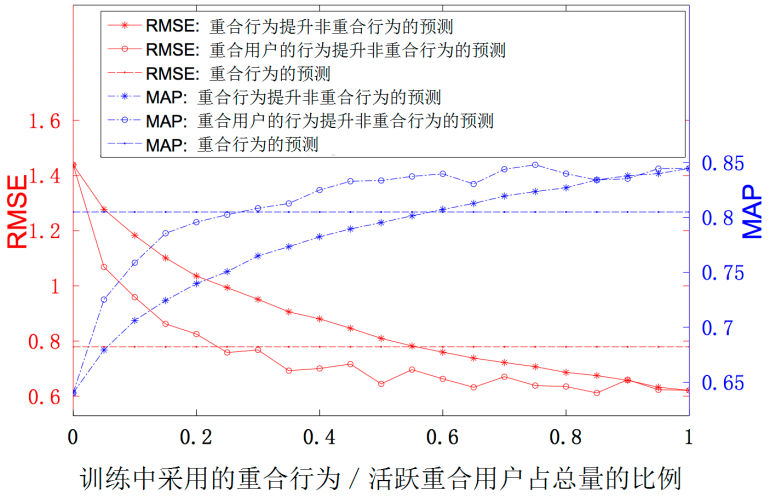


图 4.8　只需要用新浪微博和豆瓣的 26% 最活跃的重合用户，XPTrans 能够预测不重合的目标平台用户行为，达到和重合用户而不迁移时相同的效果。

户的数据，XPTrans 预测不重合用户的行为达到不迁移辅助平台中不重合用户时预测重合用户的效果。这说明小部分的重合用户能够成功地桥接不同平台的行为信息。XPTrans 的实际效果比起不采用重合用户信息要好很多，这证明了重合用户的重要性。重合用户比例很小，但对于跨平台行为预测有重大意义。

本工作的贡献和创新点如下：

- 为工业和科研提出了跨平台行为预测的有挑战性的和前景的问题。
- 分析了在两大真实的社交媒体（包括豆瓣和新浪微博）的行为，给出了采用重合用户作为跨平台联系来解决目标平台的高系数度问题。
- 设计了半监督迁移学习方法 XPTrans 来预测用户跨平台行为。大量实验证明 XPTrans 比起已有迁移学习方法的效果要好很多。实验证实了 (1) 随着重合用户的增多，非重合用户的行为预测效果一直提升；(2) 即使一小部分的重合用户都能给跨平台行为预测带来大幅度的效果提升。

4.2.2　相关工作

本小节调研相关的三个研究领域，包括跨域协同过滤、迁移学习和半监督矩阵因子分解，并指出本工作的独特之处。

许多应用系统都广泛使用跨域协同过滤方面的研究成果[4,124,133,134,140,145,148]。Adomavicius 等[8] 调研了现有推荐系统中的问题，描述了局限性并讨论了可能的拓展方向。下一代推荐系统会采纳多种数据源信息。Shi 等[137] 提出一种生成的跨标签域协同过滤方法从社交标签中获取更多的链接知识。他们调研了采纳用户 -信息交互信息的两类协同过滤算法。然而，跨域场景下，用户并不总是使用豆瓣和

符号	含义
m_P; n_P; r_P	平台 P 的用户数量; 信息数量; 特征空间的大小
K	平台数量
$\mathbf{R}^{(P)}$; $\mathbf{W}^{(P)}$	平台 P 的用户 -信息矩阵; 观测矩阵
$\mathbf{U}^{(P)}$; $\mathbf{V}^{(P)}$	平台 P 的用户聚类矩阵; 信息聚类矩阵
$\mathbf{W}^{(P,Q)}$	平台 P 和 Q 的重合用户匹配矩阵

表 4.3　本章节的符号和定义

新浪微博。另外，收到数据限制和用户匹配技术的限制，两个百万用户平台的重合用户数量实则非常小。不仅两个平台的信息内容之间并没有自然链接，而且用户集合也并没有办法在同样的表征空间对齐。跨平台行为建模依旧是开放性问题。

迁移学习算法因为效果突出而广泛运用在实际系统中[123,129,131,136,139,147,151,155]。Pan 等[130] 调研了迁移学习的具体分类，其中很多工作都在尝试加深迁移学习的效果。Li 等[126] 提出了基于多个评分矩阵的生成模型。Li 等提出了 Codebook[125] 通过特征空间来迁移知识，该方法假设辅助域（Netflix）的行为和目标域（MovieLens）的行为共享同样的评分模式。Yang 等[127,138] 提出了多元异构迁移学习来桥接 “用户 -标签” 和 “用户 -图像” 的网络。Chen 等[144] 用张量分解模型来融合用户、标签和书籍/电影信息到单一模型中。Tan 等[156] 提出了多个视图和多个信息源的迁移学习。Lin 等[153] 假设同质数据（评分范围是一致的）中的用户可以在生成模型中共用参数。然而，在不同平台上的不同类型用户行为形成了不同的行为模式，也就需要不同的表征方法。上述迁移学习方法或是特征空间中把用户对齐，或是假设两部分用户集合独立。本工作指出跨平台行为预测中重合用户的重要性。

采用非负矩阵分解实现半监督学习的方法已经被提出过[10,92,152]。对于多标签学习，Liu 等[164] 提出了带约束的非负矩阵分解来最小化输入模式之间的不同。对于聚类问题，Li 等[165] 采用非负矩阵分解来整合来自分布式数据资源中的不同形式的背景知识。Wang 等[166] 利用矩阵分解的约束项实现不同类型数据集的同时聚类。Lee 等[167] 在文档聚类和 EEG 分类任务中采用了半监督的非负矩阵因子分解模型。受到上述方法和应用系统的启发，本工作考虑了如何用跨平台的重合用户作为约束项来实现用于行为预测的半监督非负矩阵分解方法。

4.2.3　以重合用户桥接多平台的迁移性分析

本小节给出跨域行为预测问题的基本定义，并在真实数据集上证实了半监督信息（跨平台重合用户）的重要性。

某平台域	重合用户数量/总用户数量	信息数量	行为数量
豆瓣读书	21,364/30,536	212,835	1,877,069
豆瓣电影	28,204/40,246	64,090	8,087,364
豆瓣音乐	23,757/33,938	286,464	4,141,708
微博标签	29,870/2,721,365	10,176	12,328,272
微博实体	12,027/25,586	113,591	141,908,323

表 4.4　数据统计：豆瓣平台和新浪微博平台的重合用户数量为 32,868。

表4.3列出了本章节的符号和定义。这里用 m_P 和 m_Q 分别标记平台 P 和 Q 的用户数量，并用 n_P 和 n_Q 分别标记 P 和 Q 的信息数量，用 r_P 和 r_Q 标记用户/信息的特征数量，也就是特征空间的大小。接着用 $\mathbf{R}^{(P)} \in \mathbb{R}^{m_P \times n_P}$ 和 $\mathbf{R}^{(Q)} \in \mathbb{R}^{m_Q \times n_Q}$ 定义 P 和 Q 的用户 -信息评分矩阵，其中元素 $R_{i,j}^{(P)} \geq 0$ 表示用户 i 对信息 j 在平台 P 的评分。接着，定义 $\mathbf{W}^{(P)}$ 和 $\mathbf{W}^{(Q)}$ 为行为矩阵的观测数据，其中 $W_{ij}^{(P)}$ 是二元值：

$$W_{ij}^{(P)} = \begin{cases} 1, & \text{如果 } R_{ij}^{(P)} \text{ 可以观测；} \\ 0, & \text{如果 } R_{ij}^{(P)} \text{ 不可观测。} \end{cases}$$

接着，定义目标平台和辅助平台的用户聚类矩阵分别为 $\mathbf{U}^{(P)} \in \mathbb{R}^{m_P \times r_P}$ 和 $\mathbf{U}^{(Q)} \in \mathbb{R}^{m_Q \times r_Q}$；定义目标平台和辅助平台的信息聚类矩阵分别为 $\mathbf{V}^{(P)} \in \mathbb{R}^{r_P \times n_P}$ 和 $\mathbf{V}^{(Q)} \in \mathbb{R}^{r_Q \times n_Q}$；并定义平台 P 和平台 Q 的重合用户为 $\mathbf{W}^{(P,Q)} \in \mathbb{R}^{m_P \times m_Q}$，根据给定的两个平台某一对用户是否匹配决定元素为 0 或 1，也就是说

$$W_{i,j}^{(P,Q)} = \begin{cases} 1 & \text{如果平台 } P \text{ 的用户 } u_i \text{ 和平台 } Q \text{ 的用户 } u_j \text{ 相匹配，} \\ 0, & \text{否则。} \end{cases}$$

由此给出跨平台行为预测的定义如下。

定义 4.2 (跨平台行为预测)：　**给定:** 目标平台 P 和辅助平台 Q；用户 -信息矩阵 $\mathbf{R}^{(P)}$ 和 $\mathbf{R}^{(Q)}$；二元值观测矩阵 $\mathbf{W}^{(P)}$ 和 $\mathbf{W}^{(Q)}$；重合用户的匹配矩阵 $\mathbf{W}^{(P,Q)}$，**找到:** 用户聚类矩阵 $\mathbf{U}^{(P)}$ 和 $\mathbf{U}^{(Q)}$；信息聚类矩阵 $\mathbf{V}^{(P)}$ 和 $\mathbf{V}^{(Q)}$；**预测:** $\mathbf{R}^{(P)}$ 中的缺失值。

下面通过统计分析两个真实社交网络平台数据上的跨平台行为来证实跨平台预测的挑战和离合用户作为纽带的解决方案。数据集是从新浪微博和豆瓣两大社交平台爬取，在微博上有社交标签和微博实体两种信息类型，在豆瓣上又书籍、电影和音乐三种信息类型。采纳行为包括评分、转发等可以看作用户和信息之间的二值或带权值互动的链接。表4.4中列出了每一个信息域中的用户数量、信息数量和采纳信息行为数量。可以观察到的是这些平台域都非常的稀疏，而且是不同

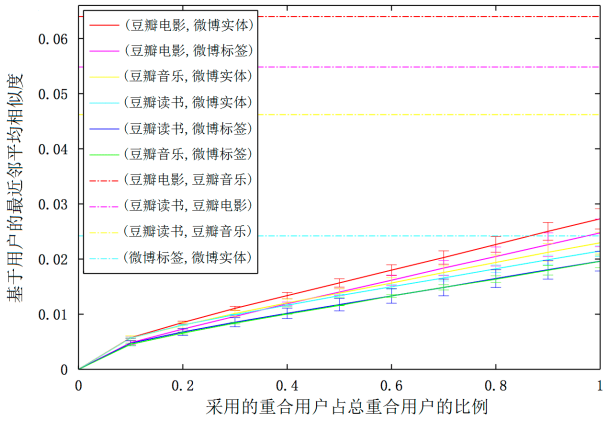


图 4.9　重合用户对联系跨平台的信息的作用：(1) 跨平台迁移比起平台内迁移要困难很多，因为基于用户的信息相似度更小；(2) 随着平台之间重合用户的增多，找到的跨平台最匹配信息相似度会更高。如果重合用户很少的时候，多一点点重合用户都会很有意义。

程度的稀疏：读书、音乐和标签的密度小到只有 0.03%-0.04%，而微博实体和电影评分的密度分别是 5% 和 0.3%。

跨平台行为模式的差异有多大？ 同一平台下，用户往往在采纳不同类型信息的时候会保持相似的行为模式。例如，豆瓣用户如果喜欢读浪漫文学，那么也会给浪漫题材电影打高分。但在跨平台的场景下，微博用户如果经常发表含有 “政府” 和 “政治” 的微博，是否会给 “纸牌屋 “ 等美剧打出高分呢？对于一对信息类型 A 和 B，比如微博实体和电影，每给定一个 A 类信息，根据重合用户寻找 Jaccard 相似度最高的 B 类信息。图4.9中画出了每两对信息类型之间最大相似度的平均值。对于跨平台的情形，可以隐藏一定比例的重合用户并给出相似度值。可以观察到如下现象：

- 跨平台迁移比起平台内迁移要困难很多。同一平台下每一对信息类型之间基于用户的最大相似度（虚线）比起跨平台的情形一直要高很多（实线）。
- 当两个平台的重合用户增多的时候，基于用户的相似度也平稳上升。
- 如果并不存在多少重合用户，每增多一个都会带来很大的积极效应。可以看到实线从 0 到 10% 的坡度比起其他地方的坡度都要大很多。
- 两个信息类型之间的行为模式相似度是非常不同的。用户的电影品味和微博实体之间的相似度比起音乐品味和标签之间的相似度要更强一些。

重合用户是否对联系不同平台的信息有帮助？ 不同平台的各类信息，比如电影和社交标签之间、书籍和微博实体之间没有多少基于内容的联系。然而，用户的行为把跨平台的 “电影 -用户 -标签” 和 “书籍 -用户 -实体” 的联系建立起来。一个既使用豆瓣又使用新浪微博的用户在给电影打分和采纳社交标签时候会有自己的品味和行为模式。所以这些平台间重合的用户是连接两个平台的关键因素。

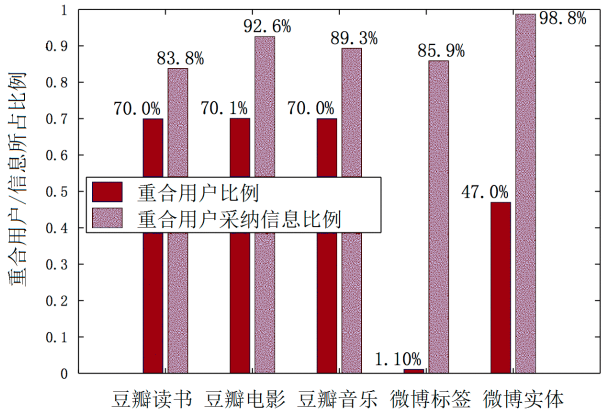


图 4.10　重合用户带来的高覆盖率：即使两个社交平台之间只有不到 1.1% 的用户重合，但这些重合用户可以覆盖每个平台上超过 80% 的信息。

那么有多少信息可以被部分重合的用户覆盖到？图4.10中给出了每一个平台域中用户和信息被重合行为覆盖的程度。重合用户数量为 32,868，而豆瓣的总用户是 4.7 万，新浪微博是 587 万。重合用户在每一个域中所覆盖的信息超过 80%。只有 1.1% 的微博用户拥有豆瓣账号，但他们对 85.9% 的社交标签都采纳过。

4.2.4　跨平台半监督迁移学习算法

要解决上述问题，可以考虑用非负矩阵的联合分解模型，其中包括 (1) 基于目标平台（平台 P）行为的特征空间优化

$$\sum_{i=1}^{m_P}\sum_{j=1}^{n_P} W_{i,j}^{(P)}\left(R_{i,j}^{(P)} - \sum_{r=1}^{r_P} U_{i,r}^{(P)} V_{r,j}^{(P)}\right)^2; \tag{4-36}$$

(2) 基于辅助平台（平台 Q）行为的特征空间优化

$$\sum_{i=1}^{m_Q}\sum_{j=1}^{n_Q} W_{i,j}^{(Q)}\left(R_{i,j}^{(Q)} - \sum_{r=1}^{r_Q} U_{i,r}^{(Q)} V_{r,j}^{(Q)}\right)^2; \tag{4-37}$$

以及 (3) 用跨平台重合用户的指示矩阵 $\mathbf{W}^{(P,Q)}$ 来约束跨平台的用户间相似度

$$\begin{aligned}
&\sum_{i_1=1}^{m_P}\sum_{j_1=1}^{m_Q}\sum_{i_2=1}^{m_P}\sum_{j_2=1}^{m_Q} W_{i_1,j_1}^{(P,Q)} W_{i_2,j_2}^{(P,Q)}\left(A_{i_1,i_2}^{(P)} - A_{j_1,j_2}^{(Q)}\right)^2 \\
= &\sum_{i_1=1}^{m_P}\sum_{i_2=1}^{m_P} A_{i_1,i_2}^{(P)2}\sum_{j_1=1}^{m_Q}\sum_{j_2=1}^{m_Q} W_{i_1,j_1}^{(P,Q)} W_{i_2,j_2}^{(P,Q)} + \sum_{j_1=1}^{m_Q}\sum_{j_2=1}^{m_Q} A_{j_1,j_2}^{(Q)2}\sum_{i_1=1}^{m_P}\sum_{i_2=1}^{m_P} W_{i_1,j_1}^{(P,Q)} W_{i_2,j_2}^{(P,Q)} \\
&-2\sum_{i_1=1}^{m_P}\sum_{j_1=1}^{m_Q}\sum_{i_2=1}^{m_P}\sum_{j_2=1}^{m_Q} W_{i_1,j_1}^{(P,Q)} W_{i_2,j_2}^{(P,Q)} A_{i_1,i_2}^{(P)} A_{j_1,j_2}^{(Q)}
\end{aligned} \tag{4-38}$$

其中 $A^{(P)}_{i_1,i_2}$ 是平台 P 上用户 u_{i_1} 和 u_{i_2} 之间的相似度，$A^{(Q)}_{j_1,j_2}$ 是平台 Q 上用户 u_{j_1} 和 u_{j_2} 之间的相似度：

$$A^{(P)}_{i_1,i_2} = \sum_{r=1}^{r_P} U^{(P)}_{i_1,r} U^{(P)}_{i_2,r};\ A^{(Q)}_{j_1,j_2} = \sum_{r=1}^{r_Q} U^{(Q)}_{j_1,r} U^{(Q)}_{j_2,r} \tag{4-39}$$

由此可以得到优化问题如下，也就是最小化

$$\begin{aligned}\mathcal{J} \ = \ & \sum_{i,j} W^{(P)}_{i,j}\left(R^{(P)}_{i,j} - \sum_r U^{(P)}_{i,r} V^{(P)}_{r,j}\right)^2 + \lambda \sum_{i,j} W^{(Q)}_{i,j}\left(R^{(Q)}_{i,j} - \sum_r U^{(Q)}_{i,r} V^{(Q)}_{r,j}\right)^2 \\ & + \mu \sum_{i_1,j_1,i_2,j_2} W^{(P,Q)}_{i_1,j_1} W^{(P,Q)}_{i_2,j_2}\left(A^{(P)}_{i_1,i_2} - A^{(Q)}_{j_1,j_2}\right)^2 \end{aligned} \tag{4-40}$$

公式 (4-40) 可以写成非负矩阵分解问题的形式。目标函数是最小化

$$\begin{aligned}\mathcal{J} = & \|\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{V}^{(P)})\|^2_F + \lambda \|\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - \mathbf{U}^{(Q)}\mathbf{V}^{(Q)})\|^2_F \\ & + \mu(\|\mathbf{W}^{(P,Q)}\mathbf{1}^{(Q)}\mathbf{W}^{(P,Q)\top} \odot \mathbf{U}^{(P)}\mathbf{U}^{(P)\top} \odot \mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\| \\ & + \|\mathbf{W}^{(P,Q)\top}\mathbf{1}^{(P)}\mathbf{W}^{(P,Q)} \odot \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top} \odot \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\| \\ & - 2\|\mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\mathbf{W}^{(P,Q)}\mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\mathbf{W}^{(P,Q)\top}\|) \\ s.t.\ & \mathbf{U}^{(P)} > 0, \mathbf{V}^{(P)} > 0, \mathbf{U}^{(Q)} > 0, \mathbf{V}^{(Q)} > 0 \end{aligned} \tag{4-41}$$

其中 $\mathbf{1}^{(P)} \in \mathbb{R}^{m_P \times m_P}$ 和 $\mathbf{1}^{(Q)} \in \mathbb{R}^{m_Q \times m_Q}$ 都是用 1 填充的矩阵。λ 是描述从辅助平台 Q 到目标平台 P 做知识迁移的非监督项的权重，μ 是决定监督项也就是重合用户相似度约束表征的权重，$\odot$ 是阿达玛乘积，$\|\cdot\|$ 是 1-范数，$\|\cdot\|_F$ 是 Frobenius 范数。

如标准非负矩阵分解方法[167]，最小化公式 (4-41) 时 $\mathbf{U}^{(P)}$，$\mathbf{U}^{(Q)}$，$\mathbf{V}^{(P)}$ 和 $\mathbf{V}^{(Q)}$ 的梯度很容易得到：

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{U}^{(P)}} \ = \ & -2[\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{V}^{(P)})]\mathbf{V}^{(P)\top} + 4\mu[\mathbf{W}^{(P,Q)}\mathbf{1}^{(Q)}\mathbf{W}^{(P,Q)\top} \odot \mathbf{U}^{(P)}\mathbf{U}^{(P)\top}]\mathbf{U}^{(P)} \\ & -4\mu[\mathbf{W}^{(P,Q)}\mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\mathbf{W}^{(P,Q)\top}]\mathbf{U}^{(P)} \end{aligned} \tag{4-42}$$

$$\begin{aligned}\frac{\partial J}{\partial \mathbf{U}^{(Q)}} \ = \ & -2\lambda[\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - \mathbf{U}^{(Q)}\mathbf{V}^{(Q)})]\mathbf{V}^{(Q)\top} + 4\mu[\mathbf{W}^{(P,Q)\top}\mathbf{1}^{(P)}\mathbf{W}^{(P,Q)} \odot \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}]\mathbf{U}^{(Q)} \\ & -4\mu[\mathbf{W}^{(P,Q)\top}\mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\mathbf{W}^{(P,Q)}]\mathbf{U}^{(Q)} \end{aligned} \tag{4-43}$$

$$\frac{\partial \mathcal{J}}{\partial \mathbf{V}^{(P)}} = -2\mathbf{U}^{(P)\top}[\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{V}^{(P)})] \tag{4-44}$$

$$\frac{\partial \mathcal{J}}{\partial \mathbf{V}^{(Q)}} = -2\mathbf{U}^{(Q)\top}[\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - \mathbf{U}^{(Q)}\mathbf{V}^{(Q)})] \tag{4-45}$$

那么矩阵的更新法则如下：

$$\mathbf{U}^{(P)} \ \leftarrow \ \mathbf{U}^{(P)} \odot \frac{[\mathbf{W}^{(P)} \odot \mathbf{R}^{(P)}]\mathbf{V}^{(P)\top} + 2\mu[\mathbf{W}^{(P,Q)}\mathbf{A}^{(Q)}\mathbf{W}^{(P,Q)\top}]\mathbf{U}^{(P)}}{[\mathbf{W}^{(P)} \odot \mathbf{U}^{(P)}\mathbf{V}^{(P)}]\mathbf{V}^{(P)\top} + 2\mu[\mathbf{W}^{(P,Q)}\mathbf{1}^{(Q)}\mathbf{W}^{(P,Q)\top} \odot \mathbf{A}^{(P)}]\mathbf{U}^{(P)}} \tag{4-46}$$

$$\mathbf{U}^{(Q)} \leftarrow \mathbf{U}^{(Q)} \odot \frac{\lambda[\mathbf{W}^{(Q)} \odot \mathbf{R}^{(Q)}]\mathbf{V}^{(Q)\top} + 2\mu[\mathbf{W}^{(P,Q)\top}\mathbf{A}^{(P)}\mathbf{W}^{(P,Q)}]\mathbf{U}^{(Q)}}{\lambda[\mathbf{W}^{(Q)} \odot \mathbf{U}^{(Q)}\mathbf{V}^{(Q)}]\mathbf{V}^{(Q)\top} + 2\mu[\mathbf{W}^{(P,Q)\top}\mathbf{1}^{(P)}\mathbf{W}^{(P,Q)} \odot \mathbf{A}^{(Q)}]\mathbf{U}^{(Q)}} \tag{4-47}$$

$$\mathbf{V}^{(P)} \leftarrow \mathbf{V}^{(P)} \odot \frac{\mathbf{U}^{(P)\top}[\mathbf{W}^{(P)} \odot \mathbf{R}^{(P)}]}{\mathbf{U}^{(P)\top}[\mathbf{W}^{(P)} \odot \mathbf{U}^{(P)}\mathbf{V}^{(P)}]} \tag{4-48}$$

$$\mathbf{V}^{(Q)} \leftarrow \mathbf{V}^{(Q)} \odot \frac{\mathbf{U}^{(Q)\top}[\mathbf{W}^{(Q)} \odot \mathbf{R}^{(Q)}]}{\mathbf{U}^{(Q)\top}[\mathbf{W}^{(Q)} \odot \mathbf{U}^{(Q)}\mathbf{V}^{(Q)}]} \tag{4-49}$$

其中 $\mathbf{A}^{(P)} = \mathbf{U}^{(P)}\mathbf{U}^{(P)\top}$ 和 $\mathbf{A}^{(Q)} = \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}$。

通用的跨平台行为预测的表示： 可以把跨平台行为预测扩展到通用情况。定义平台数量为 K，$\mathbf{R}^{(k)} \in \mathbb{R}^{m_k \times n_k}$ 是在第 k 个平台上用户 -信息的行为矩阵，其中 m_k 是该平台的用户数量，n_k 是信息数量。类似地，定义用户特征矩阵和信息特征矩阵为 $\mathbf{U}^{(k)} \in \mathbb{R}^{m_k \times r_k}$，$\mathbf{V}^{(k)} \in \mathbb{R}^{r_k \times n_k}$，定义二值的观测矩阵为 $\mathbf{W}^{(k)} \in \mathbb{R}^{m_k \times n_k}$。接着定义第 k 个平台和第 k' 个平台之间部分重合用户的指示矩阵为 $\mathbf{W}^{(k,k')} \in \mathbb{R}^{m_k \times m_{k'}}$。预测方法要联合优化用户行为矩阵，并利用跨平台的重合用户作为监督项。所以通用的目标函数是最小化如下公式

$$\mathcal{J} = \sum_k \lambda_k \sum_{i,j} W_{i,j}^{(k)} \left(R_{i,j}^{(k)} - \sum_r U_{i,r}^{(k)} V_{r,j}^{(k)} \right)^2 + \sum_{(k,k')} \mu_{k,k'} \sum_{i_1,j_1,i_2,j_2} W_{i_1,j_1}^{(k,k')} W_{i_2,j_2}^{(k,k')} \left(A_{i_1,i_2}^{(k)} - A_{j_1,j_2}^{(k')} \right)^2 \tag{4-50}$$

其中

$$A_{i_1,i_2}^{(k)} = \sum_{r=1}^{r_k} U_{i_1,r}^{(k)} U_{i_2,r}^{(k)};\ A_{j_1,j_2}^{(k')} = \sum_{r=1}^{r_{k'}} U_{j_1,r}^{(k')} U_{j_2,r}^{(k')}, \tag{4-51}$$

λ_k 是第 k 平台的用户行为权重，$\mu_{k,k'}$ 是第 k 平台和第 k' 平台的重合用户行为相似度作为约束项的权重。

根据 $\mathbf{U}^{(k)}$ 和 $\mathbf{V}^{(k)}$ 得到预测错误的目标函数的梯度：

$$\begin{aligned} \frac{\partial J}{\partial \mathbf{U}^{(k)}} &= -2\lambda_k[\mathbf{W}^{(k)} \odot (\mathbf{R}^{(k)} - \mathbf{U}^{(k)}\mathbf{V}^{(k)})]\mathbf{V}^{(k)\top} + 4\sum_{k'} \mu_{k,k'}[\mathbf{W}^{(k,k')}\mathbf{1}^{(k')}\mathbf{W}^{(k,k')\top} \odot \mathbf{U}^{(k)}\mathbf{U}^{(k)\top}]\mathbf{U}^{(k)} \\ &\quad -4\sum_{k'} \mu_{k,k'}[\mathbf{W}^{(k,k')}\mathbf{U}^{(k')}\mathbf{U}^{(k')\top}\mathbf{W}^{(k,k')\top}]\mathbf{U}^{(k)} \end{aligned} \tag{4-52}$$

$$\frac{\partial J}{\partial \mathbf{V}^{(k)}} = -2\lambda_k \mathbf{U}^{(k)\top}[\mathbf{W}^{(k)} \odot (\mathbf{R}^{(k)} - \mathbf{U}^{(k)}\mathbf{V}^{(k)})] \tag{4-53}$$

其中 $\mathbf{1}^{(k')} \in \mathbb{R}^{m_{k'} \times r_{k'}}$ 中所有元素都是 1。由此可以得到更新法则：

$$\mathbf{U}^{(k)} \leftarrow \mathbf{U}^{(k)} \odot \frac{\lambda_k[\mathbf{W}^{(k)} \odot \mathbf{R}^{(k)}]\mathbf{V}^{(k)\top} + 2\sum \mu_{k,k'}[\mathbf{W}^{(k,k')}\mathbf{A}^{(k')}\mathbf{W}^{(k,k')\top}]\mathbf{U}^{(k)}}{\lambda_k[\mathbf{W}^{(k)} \odot \mathbf{U}^{(k)}\mathbf{V}^{(k)}]\mathbf{V}^{(k)\top} + 2\sum \mu_{k,k'}[\mathbf{W}^{(k,k')}\mathbf{1}^{(k')}\mathbf{W}^{(k,k')\top} \odot \mathbf{A}^{(k)}]\mathbf{U}^{(k)}} \tag{4-54}$$

$$\mathbf{V}^{(k)} \leftarrow \mathbf{V}^{(k)} \odot \frac{\mathbf{U}^{(k)\top}[\mathbf{W}^{(k)} \odot \mathbf{R}^{(k)}]}{\mathbf{U}^{(k)\top}[\mathbf{W}^{(k)} \odot \mathbf{U}^{(k)}\mathbf{V}^{(k)}]} \tag{4-55}$$

其中 $\mathbf{A}^{(k)} = \mathbf{U}^{(k)}\mathbf{U}^{(k)\top}$ 和 $\mathbf{A}^{(k')} = \mathbf{U}^{(k')}\mathbf{U}^{(k')\top}$。

输入： 用户 -信息（采纳、评分）行为矩阵 $\mathbf{R}^{(k)}$，二值的观测矩阵 $\mathbf{W}^{(k)}$，重合用户的指示矩阵 $\mathbf{W}^{(k,k')}$；
输出： 用户特征矩阵 $\mathbf{U}^{(k)}$ 和信息特征矩阵 $\mathbf{V}^{(k)}$ （$k = 1, ..., K$）
1. 初始化 $\mathbf{U}^{(k)}$ 和 $\mathbf{V}^{(k)}$
2. 重复下面步骤的更新直到收敛：
 (a) 保持 $\mathbf{V}^{(k)}$ 不变，用公式 (4-54) 更新 $\mathbf{U}^{(k)}$；
 (b) 保持 $\mathbf{U}^{(k)}$ 不变，用公式 (4-55) 更新 $\mathbf{V}^{(k)}$。

表 4.5　XPTrans：用于跨平台行为预测的半监督迁移学习方法

XPTrans 的算法和复杂度： 解出公式 (4-50) 的算法在表4.5中给出。XPTrans 的计算复杂度为 $O(\sum_k m_k n_k r_k + \sum_{k,k'}(m_k m_{k'}(m_k + m_{k'} + r_{k'}) + m_k^2(r_k + r_{k'})))$。因为其中的 $n_k, m_k, m_{k'} \gg r_k, r'_k$（常数），可知算法的复杂性可表示为立方级时间 $O(m(m^2+n))$。

对重合用户的增量学习方法 XPTrans-inc： 每当系统得到新的一批平台间用户匹配数据，和重新优化目标函数这个代价颇高的做法相比，是否可以直接用增量数据更新特征矩阵？所以需要提出近似预测算法来更新特征矩阵，做到速度快、损失小。增量学习问题可以定义为：

定义 4.3 (增量重合用户的跨平台行为预测)： **给定：** 原始输入 $\mathbf{R}^{(P)}$、$\mathbf{R}^{(Q)}$、$\mathbf{W}^{(P)}$、$\mathbf{W}^{(Q)}$ 和 $\mathbf{W}^{(P,Q)}$；作为输入的原始输出 $\mathbf{U}^{(P)}$、$\mathbf{U}^{(Q)}$、$\mathbf{V}^{(P)}$ 和 $\mathbf{V}^{(Q)}$；重合用户指示矩阵的更新 $\Delta\mathbf{W}^{(P,Q)}$，**找到：** 用户特征矩阵的更新 $\Delta\mathbf{U}^{(P)}$ 和 $\Delta\mathbf{U}^{(Q)}$；信息特征矩阵的更新 $\Delta\mathbf{V}^{(P)}$ 和 $\Delta\mathbf{V}^{(Q)}$。

给定重合用户指示矩阵的增量后，新的目标函数可以用 $\mathcal{J}_{inc}$ 来表示：

$$
\begin{aligned}
\mathcal{J}_{inc} \;=\; & \|\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - (\mathbf{U}^{(P)} + \Delta\mathbf{U}^{(P)})(\mathbf{V}^{(P)} + \Delta\mathbf{V}^{(P)}))\|_F^2 \\
& + \lambda\|\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - (\mathbf{U}^{(Q)} + \Delta\mathbf{U}^{(Q)})(\mathbf{V}^{(Q)} + \Delta\mathbf{V}^{(Q)}))\|_F^2 \\
& + \mu\|[(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})\mathbf{1}^{(Q)}(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})^\top] \\
& \quad \odot (\mathbf{U}^{(P)} + \Delta\mathbf{U}^{(P)})(\mathbf{U}^{(P)} + \Delta\mathbf{U}^{(P)})^\top \odot (\mathbf{U}^{(P)} + \Delta\mathbf{U}^{(P)})(\mathbf{U}^{(P)} + \Delta\mathbf{U}^{(P)})^\top\| \\
& + \mu\|[(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})^\top\mathbf{1}^{(P)}(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})] \\
& \quad \odot (\mathbf{U}^{(Q)} + \Delta\mathbf{U}^{(Q)})(\mathbf{U}^{(Q)} + \Delta\mathbf{U}^{(Q)})^\top \odot (\mathbf{U}^{(Q)} + \Delta\mathbf{U}^{(Q)})(\mathbf{U}^{(Q)} + \Delta\mathbf{U}^{(Q)})^\top\| \\
& - 2\mu\|(\mathbf{U}^{(P)} + \Delta\mathbf{U}^{(P)})(\mathbf{U}^{(P)} + \Delta\mathbf{U}^{(P)})^\top(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})(\mathbf{U}^{(Q)} + \Delta\mathbf{U}^{(Q)}) \\
& (\mathbf{U}^{(Q)} + \Delta\mathbf{U}^{(Q)})^\top(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})^\top\| \qquad (4\text{-}56)
\end{aligned}
$$

展开该公式后，只留下一阶近似，也就是假设所有高阶项，比如 $\Delta\mathbf{U}^{(P)}\Delta\mathbf{V}^{(P)}$ 和 $\Delta\mathbf{U}^{(Q)}\Delta\mathbf{V}^{(Q)}$ 都是可忽略的。定义新的重合用户矩阵为 $\hat{\mathbf{W}}^{(P,Q)} = \mathbf{W}^{(P,Q)} =$

$\Delta\mathbf{W}^{(P,Q)}$。继续利用 $\mathcal{J}$ 已经达到最小值的事实，最小化的目标变为：

$$\begin{aligned}
\Delta\mathcal{J} &= \mathcal{J}_{inc} - \mathcal{J} \\
&= -2\|[\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{V}^{(P)})]^\top [\mathbf{W}^{(P)} \odot (\Delta\mathbf{U}^{(P)}\mathbf{V}^{(P)} + \mathbf{U}^{(P)}\Delta\mathbf{V}^{(P)})]\| \\
&\quad -2\lambda\|[\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - \mathbf{U}^{(Q)}\mathbf{V}^{(Q)})]^\top [\mathbf{W}^{(Q)} \odot (\Delta\mathbf{U}^{(Q)}\mathbf{V}^{(Q)} + \mathbf{U}^{(Q)}\Delta\mathbf{V}^{(Q)})]\| \\
&\quad +2\mu\|\Delta\mathbf{W}^{(P,Q)}\mathbf{1}^{(Q)}\hat{\mathbf{W}}^{(P,Q)\top} \odot \mathbf{U}^{(P)}\mathbf{U}^{(P)\top} \odot \mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\| \\
&\quad +4\mu\|\hat{\mathbf{W}}^{(P,Q)}\mathbf{1}^{(Q)}\hat{\mathbf{W}}^{(P,Q)\top} \odot \mathbf{U}^{(P)}\mathbf{U}^{(P)\top} \odot \Delta\mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\| \\
&\quad +2\mu\|\Delta\mathbf{W}^{(P,Q)\top}\mathbf{1}^{(P)}\hat{\mathbf{W}}^{(P,Q)} \odot \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top} \odot \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\| \\
&\quad +4\mu\|\hat{\mathbf{W}}^{(P,Q)\top}\mathbf{1}^{(P)}\hat{\mathbf{W}}^{(P,Q)} \odot \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top} \odot \Delta\mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\| \\
&\quad -4\mu\|\Delta\mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\hat{\mathbf{W}}^{(P,Q)}\mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\hat{\mathbf{W}}^{(P,Q)\top}\| \\
&\quad -4\mu\|\mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\hat{\mathbf{W}}^{(P,Q)}\Delta\mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\hat{\mathbf{W}}^{(P,Q)\top}\| \\
&\quad -4\mu\|\mathbf{U}^{(P)}\mathbf{U}^{(P)\top}\Delta\mathbf{W}^{(P,Q)}\mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}\hat{\mathbf{W}}^{(P,Q)\top}\| \qquad (4\text{-}57)
\end{aligned}$$

由此可以得到 $\Delta\mathbf{U}^{(P)}$、$\Delta\mathbf{U}^{(Q)}$、$\Delta\mathbf{V}^{(P)}$ 和 $\Delta\mathbf{V}^{(Q)}$ 的梯度值：

$$\begin{aligned}
\frac{\partial\Delta\mathcal{J}}{\partial\Delta\mathbf{U}^{(P)}} &= -2[\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{V}^{(P)})]\mathbf{V}^{(P)\top} \\
&\quad +4\mu[(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})\mathbf{1}^{(Q)}(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})^\top \odot \mathbf{U}^{(P)}\mathbf{U}^{(P)\top}]\mathbf{U}^{(P)} \\
&\quad -4\mu[(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})\mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})^\top]\mathbf{U}^{(P)} \qquad (4\text{-}58) \\
\frac{\partial\Delta J}{\partial\Delta\mathbf{U}^{(Q)}} &= -2\lambda[\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - \mathbf{U}^{(Q)}\mathbf{V}^{(Q)})]\mathbf{V}^{(Q)\top} \\
&\quad +4\mu[(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})^\top\mathbf{1}^{(P)}(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)}) \odot \mathbf{U}^{(Q)}\mathbf{U}^{(Q)\top}]\mathbf{U}^{(Q)} \\
&\quad -4\mu[(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})^\top\mathbf{U}^{(P)}\mathbf{U}^{(P)\top}(\mathbf{W}^{(P,Q)} + \Delta\mathbf{W}^{(P,Q)})]\mathbf{U}^{(Q)} \qquad (4\text{-}59) \\
\frac{\partial\Delta\mathcal{J}}{\partial\Delta\mathbf{V}^{(P)}} &= -2\mathbf{U}^{(P)\top}[\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{V}^{(P)})] \qquad (4\text{-}60) \\
\frac{\partial\Delta\mathcal{J}}{\partial\Delta\mathbf{V}^{(Q)}} &= -2\mathbf{U}^{(Q)\top}[\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - \mathbf{U}^{(Q)}\mathbf{V}^{(Q)})] \qquad (4\text{-}61)
\end{aligned}$$

将上述等式扩展为 K 个平台的通用情况，可以得到更新法则为：

$$\begin{aligned}
\frac{\partial\Delta\mathcal{J}}{\partial\Delta\mathbf{U}^{(k)}} &= -2\lambda_k[\mathbf{W}^{(k)} \odot (\mathbf{R}^{(k)} - \mathbf{U}^{(k)}\mathbf{V}^{(k)})]\mathbf{V}^{(k)\top} \\
&\quad +4\sum_{k'}\mu_{k,k'}[(\mathbf{W}^{(k,k')} + \Delta\mathbf{W}^{(k,k')})\mathbf{1}^{(k')}(\mathbf{W}^{(k,k')} + \Delta\mathbf{W}^{(k,k')})^\top \odot \mathbf{U}^{(k)}\mathbf{U}^{(k)\top}]\mathbf{U}^{(k)} \\
&\quad -4\sum_{k'}\mu_{k,k'}[(\mathbf{W}^{(k,k')} + \Delta\mathbf{W}^{(k,k')})\mathbf{U}^{(k')}\mathbf{U}^{(k')\top}(\mathbf{W}^{(k,k')} + \Delta\mathbf{W}^{(k,k')})^\top]\mathbf{U}^{(k)} \qquad (4\text{-}62) \\
\frac{\partial\Delta\mathcal{J}}{\partial\Delta\mathbf{V}^{(k)}} &= -2\lambda_k\mathbf{U}^{(k)\top}[\mathbf{W}^{(k)} \odot (\mathbf{R}^{(k)} - \mathbf{U}^{(k)}\mathbf{V}^{(k)})] \qquad (4\text{-}63)
\end{aligned}$$

由此可知，增量方法 XPTrans-inc 的计算复杂度只有 $O(\sum_k m_k n_k r_k + \sum_{k,k'}(w^{(k,k')}m_{k'}(m_k + r_{k'}) + m_k^2(r_k + r_{k'})))$，其中 $w^{(k,k')}$ 是增量重合用户的数量。假设

$n_k, m_k, m_{k'} \gg r_k, r_{k'}$，算法复杂度可以缩减到 $O(wm(m+r))$（平方级时间）。与之前立方级的复杂度相比，速度更快。

4.3　性能评测

本节从以下两个方面评测本章所提出工作的性能：一是跨域行为预测的效果，二是跨平台行为预测的效果。

4.3.1　跨域行为预测性能

本节中给出混合随机漫步算法的实验结果。数据集是有微博域和标签域的社交媒体。主要在两个方面评价算法效果，一是社会化推荐问题，也就是预测正向链接和负向链接的准确度，二是用户冷启动问题，也就是当来的是新用户，他们的训练数据为空，如何更好的给他们做推荐。接下来分别介绍一下评价指标、参数选择和基线算法这三个实验设置环节。

工作中采用最常用评价标准，包括重构错误率，预测准确率和排序评价系数：

- 错误率评价标准：MAE 和 RMSE 可以分别定义为

$$MAE = \frac{1}{N}\sum_{u_i,p_j}(|p_{ij}^{(\mathcal{UP})^+} - \hat{p}_{ij}^{(\mathcal{UP})^+}| + |p_{ij}^{(\mathcal{UP})^-} - \hat{p}_{ij}^{(\mathcal{UP})^-}|) \tag{4-64}$$

$$RMSE = (\frac{1}{N}\sum_{u_i,p_j}(|p_{ij}^{(\mathcal{UP})^+} - \hat{p}_{ij}^{(\mathcal{UP})^+}|^2 + |p_{ij}^{(\mathcal{UP})^-} - \hat{p}_{ij}^{(\mathcal{UP})^-}|^2))^{\frac{1}{2}} \tag{4-65}$$

  其中 $p_{ij}^{(\mathcal{UP})^+}$ 和 $p_{ij}^{(\mathcal{UP})^-}$ 是测试集中用户 u_i 在选择是否采纳消息 p_j 的真实值，$\hat{p}_{ij}^{(\mathcal{UP})^+}$ 和 $\hat{p}_{ij}^{(\mathcal{UP})^-}$ 表示预测结果，N 表示测试集合的大小。所有的 p 值都是真实值，所以要么是 0，要么是 1；而 $\hat{p}$ 值是概率值，也就是在区间 $[0,1]$ 之间。用 MAE 和 RMSE 来评价推荐概率和真实值之间的错误有多大

- 预测准确率：准确率 (precision)，召回率 (recall) 和 F1 系数（即 precision 和 recall 的几何平均数）：

$$precision = \frac{|\{(u_i,p_j)|\hat{p}_{ij}^{(\mathcal{UP})^+} > \hat{p}_{ij}^{(\mathcal{UP})^-}, p_{ij}^{(\mathcal{UP})^+} = 1\}|}{|\{(u_i,p_j)|\hat{p}_{ij}^{(\mathcal{UP})^+} > \hat{p}_{ij}^{(\mathcal{UP})^-}\}|} \tag{4-66}$$

$$recall = \frac{|\{(u_i,p_j)|\hat{p}_{ij}^{(\mathcal{UP})^+} > \hat{p}_{ij}^{(\mathcal{UP})^-}, p_{ij}^{(\mathcal{UP})^+} = 1\}|}{|\{(u_i,p_j)|p_{ij}^{(\mathcal{UP})^+} = 1\}|} \tag{4-67}$$

$$F1 = \frac{2 \times precision \times recall}{precision + recall} \tag{4-68}$$

其中 $p_{ij}^{(\mathcal{UP})^+}$ 和 $p_{ij}^{(\mathcal{UP})^-}$ 是测试集中 u_i 对 p_j 做出采纳或是拒绝行为的真实值，$\hat{p}_{ij}^{(\mathcal{UP})^+}$ 和 $\hat{p}_{ij}^{(\mathcal{UP})^-}$ 表示预测结果。

- 排序评价系数：采用平均准确率（Mean Average Precision，简称 MAP，或 MAP@K）。也就是说，对于前 K 个推荐消息，给出在多个测试样例中平均准确率的平均数。实验中设置 K 为 1，3，5，10 或 20。

需要调整的参数包括信息相似度对用户纽带强度的相对权重在微博域和社交标签域的参数 δ 和 η，用户采纳信息相对于拒绝信息行为的权重 μ，从微博域、标签域的知识到社交域中用户纽带强度的知识迁移过程中的跨域权重 $\tau^{(\mathcal{P})}$ 和 $\tau^{(\mathcal{T})}$，社交域三角形关系形成新纽带强度的权重 $\tau^{(\mathcal{U})}$。算法在 0 到 1 之间对所有参数做贪心搜索，找到最好的参数组合（$\delta, \eta, \mu, \tau^{(\mathcal{U})}, \tau^{(\mathcal{P})}$ 和 $\tau^{(\mathcal{T})}$）来减小预测错误。随机漫步算法对于结构改变具有很好的局部性和鲁棒性特征，这也是随机漫步在很多推荐领域的应用中越来越流行的原因[161,162]。参数设置时可以发现 $\tau^{(\mathcal{U})}$ 小到只有 0.05 的时候，算法会达到最优值，这说明了大多数用户的行为都是只受到自己直接的朋友或是一些间接朋友，而不是陌生人的影响。同时，最好的 $\tau^{(\mathcal{P})}$ 和 $\tau^{(\mathcal{T})}$ 参数设定则表示用户纽带权重的学习既需要微博相似度，也需要标签相似度来完成，详情请看后续实验讨论。

表 4.6　混合随机漫步算法的不同设置以及和二部随机漫步的比较

算法	$\mathcal{G}^{(\mathcal{U})}$	$\mathcal{G}^{(\mathcal{P})}$	$\mathcal{G}^{(\mathcal{T})}$
HRW-Borda	$\mathbf{R}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	$\mathbf{W}^{(\mathcal{T})}$, Borda count
HRW-Popular	$\mathbf{R}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	$\mathbf{W}^{(\mathcal{T})}$, *popularity*
HRW-Cons-Post	$\mathbf{R}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	$\mathbf{W}^{(\mathcal{T})}$, $cons_{post}$
HRW-Cons-Follower	$\mathbf{R}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	$\mathbf{W}^{(\mathcal{T})}$, $cons_{follower}$
HRW-Cons-Followee	$\mathbf{R}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	$\mathbf{W}^{(\mathcal{T})}$, $cons_{followee}$
HRW-All	$\mathbf{R}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	$\mathbf{W}^{(\mathcal{T})}$, all
BRW-R_U-P (TrustWalker)	$\mathbf{R}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	×
BRW-R_U	$\mathbf{R}^{(\mathcal{U})}$	×	×
BRW-W_U-P	$\mathbf{W}^{(\mathcal{U})}$	$\mathbf{W}^{(\mathcal{P})}$	×
BRW-W_U (ItemRank)	$\mathbf{W}^{(\mathcal{U})}$	×	×
BRW-P	×	$\mathbf{W}^{(\mathcal{P})}$	×

实验中要回答这样两个问题：

- 从辅助域中迁移知识解决推荐和冷启动问题，是否比不迁移要好？
- 选取值得迁移学习的信息比用所有信息或是最流行信息要好？

因此，如表4.6所示，用两套基线方法来证明混合随机漫步算法（HRW）更有效。

这些方法从 $\mathbf{W}^{(\mathcal{U})}$，$\mathbf{W}^{(\mathcal{P})}$ 和 $\mathbf{W}^{(\mathcal{T})}$ 中使用社交域、微博域和社交域中的域内链接知识，并学习用户的纽带强度来更新 $\mathbf{R}^{(\mathcal{U})}$。要回答第一个问题，就需要与基线算法中的二部随机漫步算法（BRW）作比较。这些算法不利用标签域信息中的丰富知识来预测用户与微博之间的域间链接。此外仔细地实现先进的矩阵分解方法。

- BRW-R_U-P（TrustWalker[57]）：用微博相似度来更新用户纽带强度来预测用户与微博的域间链接；
- BRW-R_U：只用用户微博二部图, 不用微博相似度来更新用户纽带强度；
- BRW-W_U-P：用社交关系和微博相似度，不改变纽带强度来预测域间链接；
- BRW-W_U（ItemRank[9]）：只用社交关系来计算用户与微博之间的纽带强度；
- BRW-P：只学习了微博相似度来预测用户与微博之间的纽带强度；
- TLLSM（Transfer Learning with Latent Space Matching[154]）：融合用户和信息的知识来做社会化推荐，并对矩阵分解模型添加了正则化约束。

$$\begin{aligned}\min \quad & \|\mathbf{W}\odot(\mathbf{P}^{(\mathcal{UP})^{+}}-\mathbf{UV})\|^2+\|\mathbf{W}\odot(\mathbf{P}^{(\mathcal{UT})^{+}}-\mathbf{XY})\|^2 \\ & +\lambda(\|\mathbf{U}\|^2+\|\mathbf{V}\|^2+\|\mathbf{X}\|^2+\|\mathbf{Y}\|^2) \\ & +\beta\sum_m arctan(\|\mathbf{U}(m,:)-\mathbf{X}(m,:)\|^2) \end{aligned} \tag{4-69}$$

其中 $\mathbf{W}$ 是指示矩阵，λ 和 β 是正则项的参数；$\mathbf{U}$ 和 $\mathbf{X}$ 代表了用户的特征向量；$\mathbf{V}$ 和 $\mathbf{Y}$ 代表了信息的特征向量。

对于第二个问题，可以给出一系列选择可迁移信息的 HRW 方法。

- HRW-Borda：为标签手机所有特征，用 Borda count 的方法投票得到最好的 l_{top} 个标签；
- HRW-Popular：用流行度特征来选取最好的 l_{top} 个标签；
- HRW-Cons-Post：用微博集合一致性来选取最好的 l_{top} 个标签；
- HRW-Cons-Follower：用粉丝群一致性来选取最好的 l_{top} 个标签；
- HRW-Cons-Followee：用关注人群一致性来选取最好的 l_{top} 个标签；
- HRW-All：用所有的标签信息。

接下来给出实验结果。首先比较所提出 HRW 算法和其他基线算法的社会化推荐效果。接着讨论标签域中不同的社交标签的迁移能力以及不同的标签选取方法的推荐效果。第三讨论域间的迁移能力，包括用户纽带强度作为桥接关系、迁移辅助信息域的知识、正向和负向行为信息的比例的重要意义。最后展现算法在解决用户冷启动问题的效果，也给如何利用辅助信息带来启示。在社交场景下证实预测缺失域间链接的效果：随机选取 80% 的用户采纳或拒绝微博的行为作为训练数据，剩余的作为测试数据，而用户采纳社交标签的数据被完全使用。实验中

随机选取 20 次数据并做实验，给出平均值和标准差结果。

算法	MAE	RMSE	准确率	召回率
HRW-Borda	**0.195±1.3e-3**	**0.226±2.6e-3**	**0.912±4.7e-3**	**0.771±3.5e-3**
HRW-Popular	0.278±3.7e-3	0.306±1.8e-3	0.866±6.7e-4	0.728±9.1e-4
HRW-Post	0.215±3.2e-3	0.249±3.5e-3	0.900±4.7e-3	0.758±2.5e-3
HRW-Follower	0.250±4.6e-3	0.285±3.7e-3	0.881±1.9e-3	0.741±9.5e-4
HRW-Followee	0.227±2.2e-3	0.254±2.5e-3	0.891±8.1e-4	0.752±1.1e-3
HRW-All	0.260±3.5e-3	0.296±3.4e-3	0.874±4.4e-3	0.738±2.8e-3
BRW-R_U-P	0.334±3.1e-3	0.357±1.5e-3	0.832±6.3e-4	0.699±4.1e-3
BRW-R_U	0.349±3.5e-3	0.371±1.5e-3	0.831±8.8e-4	0.696±1.5e-3
BRW-W_U-P	0.377±3.6e-3	0.403±3.9e-3	0.813±3.4e-3	0.677±1.7e-3
BRW-W_U	0.390±3.9e-3	0.419±4.0e-3	0.802±3.7e-3	0.668±4.4e-3
BRW-P	0.478±3.5e-3	0.499±4.1e-3	0.754±6.3e-4	0.629±4.4e-3
TLLSM	0.361±2.6e-3	0.385±1.7e-3	0.816±2.7e-3	0.685±4.0e-3

表 4.7　混合随机漫步算法和基线算法预测缺失的用户与微博的域间链接的结果

表4.7比较混合随机漫步算法和不同的基线算法作比较，表中展示出包括MAE，RMSE，准确率和召回率的平均值和标准差。值得注意的是

- 混合随机漫步算法 HRW 比之前的方法都要效果好，而且对初始化不敏感；
- 最终的 HRW 方法，也就是用 Borda count 来选取辅助标签，比起其他所有基线算法要效果好。

通过和二部随机漫步算法 BRW 比较，观察到

- BRW-W_U 比 BRW-P 减小 MAE 达到 18.4%：BRW-W_U 从社交关系中得到用户行为的模式信息，比起简单的随机漫步实现的协同过滤算法要好。BRW-R_U 比 BRW-W_U 减小 MAE 达到 10.5%：因为改变用户纽带权重比起只用二值信息来描述要更加准确。BRW-R_U-P 比 BRW-W_U 减小 MAE 达到 14.3%：既学习域内链接（微博相似度）和域间链接（用户采纳或拒绝信息行为）来更新用户之间的纽带强度。之所以算法效果更好，是因为用户在社交网络中的行为动机是：(1) 用户往往会喜欢和他们过去采纳过的信息更相似的信息；(2) 用户喜欢采纳来自亲密好友的信息。BRW-R_U-P 融合这两个方面的信息来解决社会化推荐问题。
- HRW-All 比 BRW-R_U-P 减小 MAE 达到 22.1%：这与假设的社交网络上用户纽带强度是由多个关系域（微博域和标签域）形成的相一致。所以混合随机漫步方法有效地利用辅助信息来构建带权的用户网络，并且比 BRW 在解决用户、微博之间的链接过于稀疏的问题要好。

- 虽然 TLLSM 能够从多个信息源得到丰富的知识，HRW 比起最先进的矩阵分解模型 TLLSM 在使用辅助信息方面更好，能够降低 MAE 达到 27.9%。用户行为自然而然地形成了用户之间的纽带强度，然而 TLLSM 并没有考虑用户之间的纽带关系。

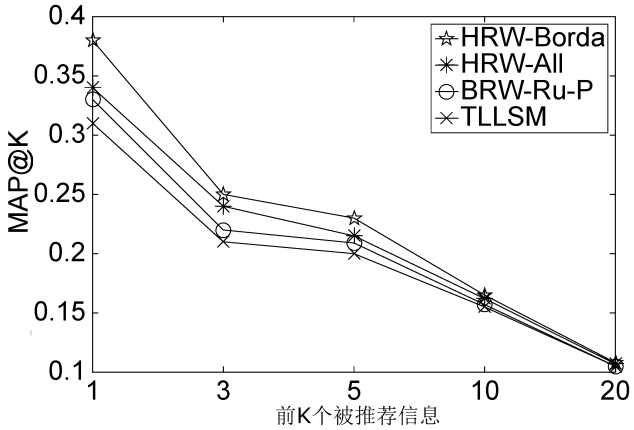


图 4.11　随机漫步算法有更好的 MAP@K 值（平均准确率）：HRW-Borda 比起所有其他算法都要好。当 K 变小的时候，MAP@K 的提升程度会变得更大。

图4.11展现了 HRW-Borda、HRW-All 方法和其他基线算法在推荐前 K 个信息的平均准确率 MAP 的值。可以观察到 (1) HRW 算法比起最好的 BRW 算法 (BRW-R_U-P) 以及矩阵分解方法 TLLSM 有更高的 MAP 数值；(2) HRW-Borda 比起其他的算法有更好的推荐效果。虽然当 K 达到 20 的时候，MAP@20 的提升效果非常小，但是 HRW 算法能够把 MAP@1 提升 16.9%，把 MAP@3 提升 17.4%。实验结果证实了随机漫步算法能够很有更好的社会化推荐的消息排序效果。

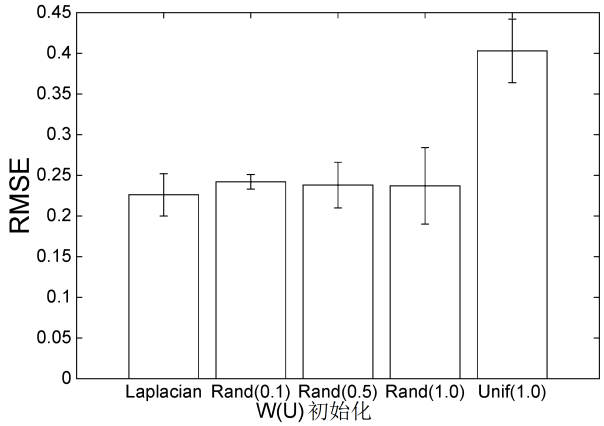


图 4.12　HRW 方法对于用户纽带强度的初始化不敏感：左边四柱是 HRW-Borda 不同 $\mathbf{W}^{(U)}$ 初始化的结果；首先是用拉普拉斯矩阵，接下来三个是在给定范围内随机生成用户之间的纽带强度。最右边设置纽带强度全部为 1（BRW-W_U-P）。

图4.12比较了多种不同的用户纽带强度矩阵 $\mathbf{W}^{(U)}$ 的初始化条件下 HRW-Borda 方法的 RMSE，其中包括 (1) 拉普拉斯 (Laplacian) 矩阵，它是二值社交图的度数矩阵的拉普拉斯变换；(2) Rand(x), $x \in \{0.1, 0.5, 1.0\}$，表示随机 0 到 x 来设置矩阵 $\mathbf{W}^{(U)}$ 中的非零值。(3) Unif(1.0)，表示设置 $\mathbf{W}^{(U)}$ 中所有的值为 1，观察到

- 前四个柱给出几乎相同的 RMSE 数值，说明 HRW 算法对于纽带强度的初始化不敏感。这些错误柱的分布说明当 x 变大的时候，RMSE 的标准车就会更大。默认设置是用拉普拉斯矩阵，会给出最小的 RMSE 值和最小的标准差。
- 最后一个柱说明当所有纽带强度都是 1 的时候，会给出更大的 RMSE 值。这个算法就相当于 BRW-R_U-P，说明忽视掉社交信息会导致预测准确率减低。

实验结果证明了融合社交关系信息能够有效提升推荐效果；推荐效果对于纽带强度的初始化不敏感。

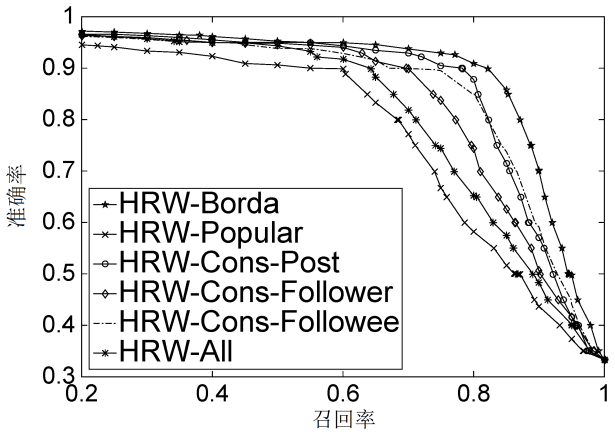
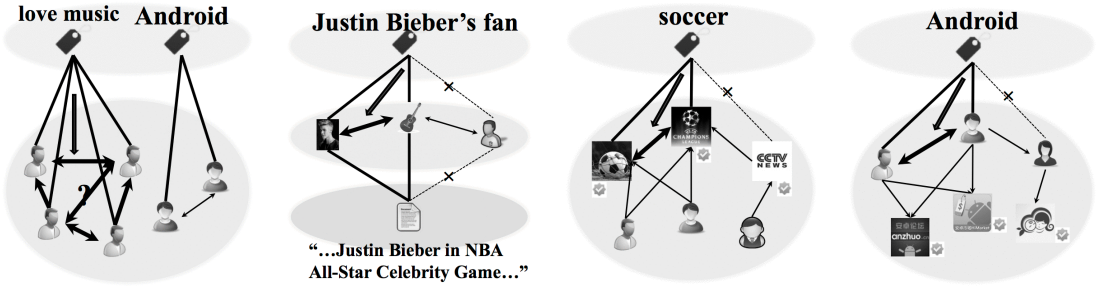


图 4.13　HRW-Borda 比起其他 HRW 方法都能有更好的准确率 -召回率结果。考虑到所有的特征，包括流行度、微博一致性、粉丝群一致性和关注人群一致性，HRW-Borda 通过选取最具有迁移能力的信息能够提升预测效果。

除去和 BRW 方法比较外，实验中还与不同给的随机漫步算法（HRW）做比较，以期得到正确的可迁移信息的选取方法。表4.7中看到 HRW-Cons-Post，HRW-Cons-Follower 和 HRW-Cons-Followee 比起 HRW-Popular 在 MAE 上分别减小 22.6%，10.0% 和 18.3%。图4.13中画出不同算法的准确率 -召回率的曲线，发现 HRW-Borda 能够达到最好的推荐效果，几乎达到了完美。以上的 5 个算法会选出同样数量的可迁移信息（社交标签）。HRW-Popular 假设共享最为流行的社交标签的用户会紧密相连。然而，在图4.14(a) 中看到 “听音乐” 有 815,166 个用户采纳，“安卓粉” 有 10,653 个用户采纳，共享 “安卓粉” 的用户比起共享 “听音乐” 的用户往往有更密集的社交关系。虽然 “听音乐” 的标签更为流行，但是也更容易出现在用户的标签集合中，以至于太难反映用户在交友和采纳微博时候的品味。这里进



(a) 流行度　(b) 与微博行为的一致性　(c) 与粉丝人群的一致性　(d) 与关注人群的一致性

图 4.14　什么样的社交标签能够反映真实的纽带强度？共享流行社交标签的用户往往并没有密切关系，但是微博一致性能够反映出共同的兴趣爱好，社交关系的一致性能够反映出用户的社区特性。HRW-Borda 算法使用混合特征的策略。

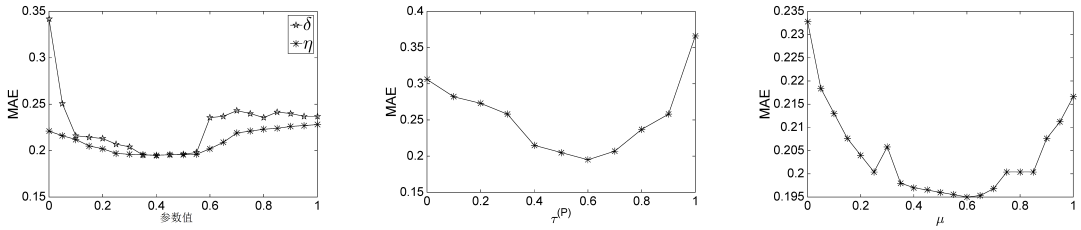
一步样例分析选择什么样的社交标签能够得到最好的迁移效果。

- HRW-Cons-Post 采用微博一致性这一特征。这个方法选取有同样微博的用户共享的标签。例如在图4.14(b) 中，标着 “Justin Bieber 粉” 的用户会都转发 “Justin 在 NBA 全明星名人赛的演出” 这条微博。HRW-Cons-Post 假设他们相互之间会有更强的纽带强度，因为他们在微博内容上有相似的兴趣偏好。
- HRW-Cons-Follower 采用粉丝群一致性这一特征。这个方法选取让用户有更相似的粉丝的标签。图4.14(c) 中展示了很多带有 “足球” 标签的知名账号，比如 “腾讯足球” 和 “CCTV5 欧洲冠军杯” 往往有 13,522 个共同的粉丝（往往都是足球迷）。这两个账号会紧密关联，互相影响。虽然 “央视新闻” 和 “CCTV5 欧洲冠军杯” 都是 CCTV 的公共账号，但是他们没有相同的标签，所以并不经常交互。
- HRW-Cons-Followee 采用关注人群一致性这一特征。这个方法选取有同样的关注人群的标签。图4.14(d) 中有 “安卓粉” 的用户往往与 “安卓论坛” 和 “安卓市场” 这样的知名账号相连，他们相互之间从知名账号处转发新的应用消息。可以看到 HRW-Cons-Followee 比起 HRW-Cons-Follower 有更好的推荐效果，因为 HRW-Cons-Followee 更多的分析普通用户之间的社交纽带强度，而 HRW-Cons-Follower 仅仅反映知名账号之间的社交强度。

在算法 HRW-Borda 里融合了所有上述特征，使用 Borda count 来选取最有迁移能力的社交标签。与 HRW-All 这个采用所有社交标签的方法作对比，HRW-Borda 能够减小 MAE 达到 25.0%。与最好的二部随机漫步算法和过去的因子分解算法相比，HRW-Borda 最终能够让 MAE 提升达到 41.6%。

这里通过分析参数的选取来回答一下三个问题来分析域间的迁移能力。

- 在预测用户、微博域间链接时信息相似度和社交强度是不是都很重要？在微



(a) MAE v.s. δ 和 η　　(b) MAE v.s. $\tau^{(\mathcal{P})}$　　(c) MAE v.s. μ

图 4.15　参数的选取过程的说明：(a) 调整控制社交纽带和信息相似度在更新域间链接的权重，(b) 调整微博域信息和标签域信息对于社交纽带的影响的权重，(c) 调整采纳行为和拒绝信息的行为在社交纽带上的权重。

博域中，δ 是微博相似度带来的社交强度的权重。如果 δ 增大，比起个人兴趣偏好，更有可能会根据社交关系去采纳被推荐的微博。η 是社交标签相似度给社交强度带来的权重。图4.15(a) 展示了随着 δ 和 η 从 0 变化到 1 的推荐效果。$\delta = 0$ 表示不考虑信息相似度的预测算法，$\delta = 1$ 表示不考虑社交强度的预测算法。图中可以观察到在 δ 接近谷底的时候，数值为 0.4。这说明融合社交纽带强度和微博相似度会很容易提升推荐效果。

- 微博域信息和社交标签域信息那个对影响社交纽带强度更有作用？$\tau^{(\mathcal{P})}$ 和 $\tau^{(\mathcal{T})}$ 表示从微博域和社交标签域计算社交纽带强度的权重，其中 $\tau^{(\mathcal{P})}+\tau^{(\mathcal{T})} = 1$。图4.15(b) 给出了从 0 到 1 改变 $\tau^{(\mathcal{P})}$ 的 MAE 数值。当 $\tau^{(\mathcal{P})} = 1$，也就是当不采用社交标签信息的话，MAE 的数值大约是 0.37。当 $\tau^{(\mathcal{P})} = 0$，也就是 $\tau^{(\mathcal{T})} = 1$ 的时候，MAE 的数值大约是 0.31。这说明了社交标签域更容易预测社交纽带强度，因为共享同样的微博的用户共享同样的社交标签。最小的 MAE 在 $\tau^{(\mathcal{P})} = 0.6$ 的时候取得，这说明从社交标签域迁移来的知识能够更有效地预测微博域中的行为。推荐系统应该融合这两种不同的域间链接信息。
- 是不是负向信息，也就是拒绝信息的行为很有作用？μ 是微博域中影响社交纽带强度的正向行为权重（采纳信息行为），那么 $1-\mu$ 是负向行为权重（拒绝信息行为）。图4.15(c) 显示了当 μ 从 0 变化到 1 的过程中 MAE 的曲线。当 $\mu = 1$（只有采纳行为）时，或者 $\mu = 0$（只有拒绝信息行为）时，MAE 都比两种行为信息都使用的时候要大，也就是当 $\mu = 0.6$ 时能达到最好的推荐效果。这说明正向数据和负向数据对于推荐效果有很大的提升。

上述讨论证明了算法的参数设计是有意义的，方法是有有效的。这说明考虑用户行为的多方面来源，更新用户之间的社交纽带强度，并同时使用正向和负向行为信息，能够很好的解决社会化推荐问题。

实验中测试推荐方法在解决用户冷启动问题的效果：(1) BRW-R_U-P 是所有二部随机漫步算法中在预测缺失链接时达到最好的效果；(2) HRW-All 用了所有的社

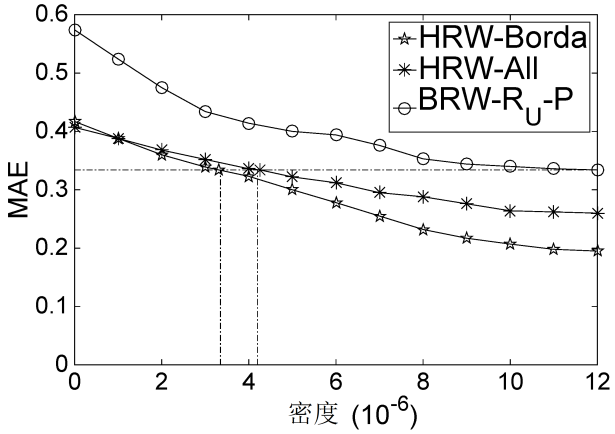


图 4.16　HRW-Borda 只需要 27.6% 的微博信息，HRW-All 只需要 35.5% 的微博历史信息，就能够达到不采纳社交标签行为信息的 BRW-R_U-P 使用 100% 微博行为的效果。混合随机漫步算法对于解决冷启动问题给出了方法。

交标签信息比起 BRW 能够提升预测效果；(3) 最终算法 HRW-Borda 迁移学习最强最有意义的信息，效果最好。控制测试用户的训练数据的密度（也就是每个测试用户每个测试信息的行为数量）。图4.16给出了实验结果。如果训练数据被隐藏，也就是测试用户的训练数据密度为 1.2×10^{-5}，HRW-Borda 能够比最好的基线算法降低 MAE 达到 41.6%（0.195 强过 0.334）。如果隐藏所有的训练数据，也就是密度降低到 0，那么测试用户是应用中的新用户（并没有历史数据和早期行为），HRW-Borda 降低 MAE 达到 27.4%（0.417 强过 0.574）。从图4.16中观察到

- 迁移学习算法 HRW-Borda 只需要 27.6% 的微博信息、HRW-All 只需要 35.5% 的微博历史信息（密度分别为 3.31×10^{-6} 和 4.26×10^{-6}），就能够达到不采纳社交标签行为信息的 BRW-R_U-P 使用 100% 微博行为的效果（密度为 1.2×10^{-5}）。也就是说利用用户的社交标签，算法只需要 3 天的历史行为数据，就能够达到拥有 10 天历史行为数据的推荐效果。所以如果能够动员新用户增加一些社交标签，从社交标签域中迁移的这是能够很好地提升个性化推荐系统的用户体验。
- 当训练数据是空的，HRW-All 能够比 HRW-Borda 要更好一些，因为它能够用更多的信息里估计用户之间的纽带强度。随着社交标签域中的行为增多，用户标签的一致性比起流行度变得重要起来，于是最终选取迁移能力最好的信息的 HRW-Borda 方法能够比 HRW-All 效果更好。

4.3.2　跨平台行为预测性能

本章节通过实验来检测本工作提出的半监督迁移学习方法 XPTrans 的效果和效率。工作中测试两种跨平台行为检测方法：

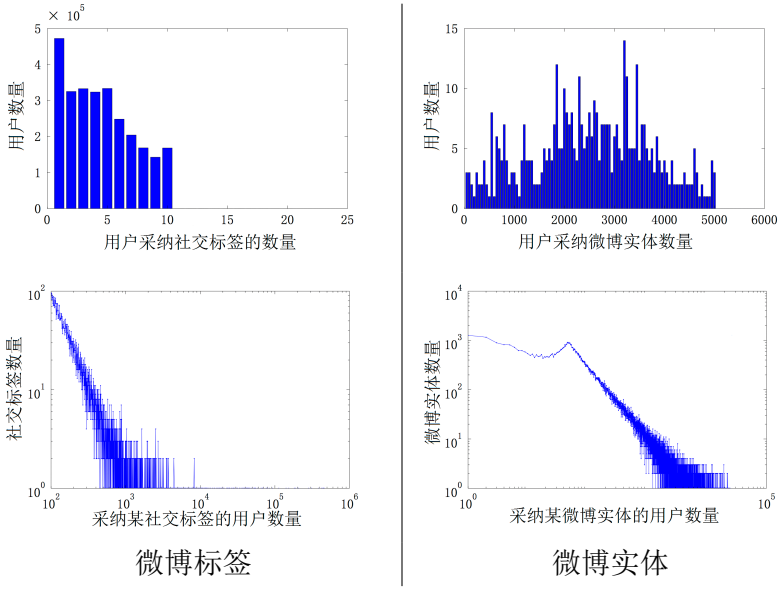


图 4.17　微博数据集的长尾数据分布。

- XPTrans（半监督迁移学习方法）：该方法采用跨平台重合用户作为监督项。
- XPTrans-inc（增量重合用户的半监督迁移学习方法）：用增量的监督数据近似更新聚类特征矩阵。

所有算法都是在 MATLAB 中实现，在 2.40GHz×8 Intel Xeon CPU、64GB 内存和 Windows Server 2008-64 位系统上运行。

如表4.4所示，单一平台的用户行为（评分、采纳信息等）通常很稀疏。图4.17和图4.18中给出每一种数据类型的长尾分布。在实验中使用了新浪微博的标签、微博实体和豆瓣的读书、电影、音乐。实验按照每一对的（辅助平台，目标平台）比照行为预测效果，总共有 $\binom{2}{1}\binom{2}{1}\binom{3}{1}=12$ 对，例如

- 从新浪微博实体迁移预测豆瓣电影评分：是否能从微博内容迁移学习用户的兴趣爱好（如喜欢政治等）来预测用户会给什么样的电影（如《纸牌屋》等）打高分？
- 从豆瓣读书迁移预测微博社交标签：是否能从豆瓣读书迁移学习用户的品味（如《时间简史》等）来预测用户的社交标签（如“科学狂人”等）？

通过样本保持实验（hold-out experiment）来测试跨平台迁移学习的效果。设置实验时采用以下 4 个参数：

- $\alpha_{\mathbf{R}^{(P)}}^{(P\setminus Q)} \in [60\%, 90\%]$：目标平台 P 上不重合行为数据中训练的百分比；
- $\alpha_{\mathbf{R}^{(Q)}}^{(Q\setminus P)} \in [0, 100\%]$：辅助平台 Q 上不重合行为数据中训练的百分比；
- $\alpha_{\mathbf{R}}^{(P\cap Q)} \in [0, 100\%]$：跨平台 P 和 Q 的重合行为数据训练的百分比；
- $\alpha_{\mathbf{U}}^{(P\cap Q)} \in [0, 100\%]$：跨平台 P 和 Q 的最活跃重合用户训练的百分比。

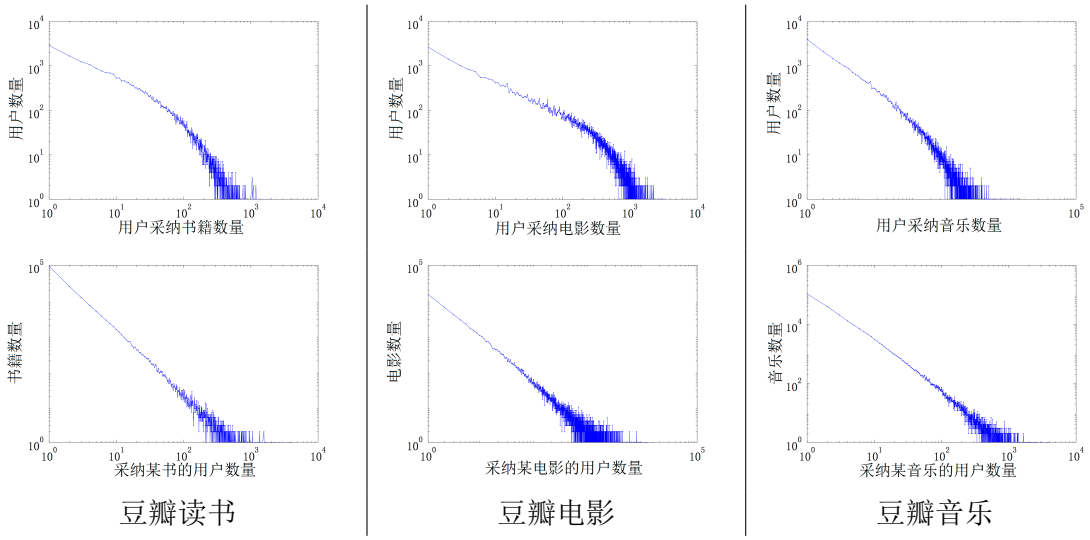


图 4.18　豆瓣数据集的长尾数据分布。

给定上述的实验设置参数，随机选取符合比例的训练数据和辅助平台数据，并作 10 次实验并给出平均的预测结果。

实验中和以下的包括几个先进算法等基线算法作比较：

- CMF （带约束的协同矩阵分解）[92]：该方法并不从辅助平台中迁移学习知识。目标函数是：

$$\mathcal{J} = \|\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{V}^{(P)})\|_F^2 + \mathcal{L}_2(\mathbf{U}^{(P)}, \mathbf{V}^{(P)}) \tag{4-70}$$

- CBT （“编码本” 迁移）[125]：该方法能够把特征空间里的用户评分模式迁移学习到目标域，称之为 “编码本”。然而，这个方法并不考虑监督项，也就是平台之间的重合用户。目标函数如下：

$$\mathcal{J}_1 = \|\mathbf{W}^{(Q)} \odot (\mathbf{R}^{(Q)} - \mathbf{U}^{(Q)}\mathbf{B}\mathbf{V}^{(Q)})\|_F^2 \tag{4-71}$$

$$\mathcal{J}_2 = \|\mathbf{W}^{(P)} \odot (\mathbf{R}^{(P)} - \mathbf{U}^{(P)}\mathbf{B}\mathbf{V}^{(P)})\|_F^2 \tag{4-72}$$

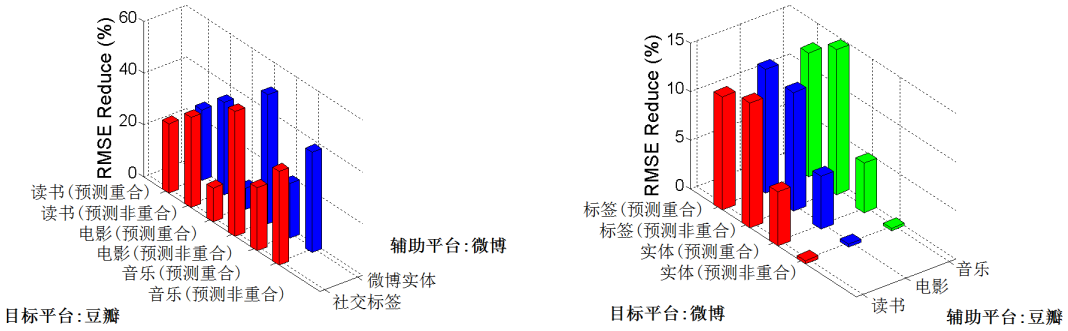
其中 $\mathbf{B} \in \mathbb{R}^{r \times r}$ $(r = r_P = r_Q)$ 是平台间相似度编码，即 “编码本”。

- XPTrans-align：这是本工作中提出的跨平台行为预测方法的一个变种，该变种带有非常强的假设，也就是同一个用户在不同的平台下的行为表征是完全一样的，当然，特征数目也是一样的（$r = r_P = r_Q$）。于是重合用户的特征可以在同一个空间内对齐。目标函数如下：

$$\begin{aligned}\mathcal{J} = & \sum_{i,j} W_{i,j}^{(P)} \left(R_{i,j}^{(P)} - \sum_r U_{i,r}^{(P)} V_{r,j}^{(P)} \right)^2 + \lambda \sum_{i,j} W_{i,j}^{(Q)} \left(R_{i,j}^{(Q)} - \sum_r U_{i,r}^{(Q)} V_{r,j}^{(Q)} \right)^2 \\ & + \mu \sum_{i,j} W_{i,j}^{(P,Q)} \sum_r \left(U_{i,r}^{(P)} - U_{j,r}^{(Q)} \right)^2 \end{aligned} \tag{4-73}$$

	Q: 微博实体 → P: 豆瓣电影				Q: 豆瓣读书 → P: 微博标签			
	RMSE		MAP		RMSE		MAP	
	$P \cap Q$	$P \setminus Q$	$P \cap Q$	$P \setminus Q$	$P \cap Q$	$P \setminus Q$	$P \cap Q$	$P \setminus Q$
CMF[92]	0.779	1.439	0.805	0.640	0.267	0.429	0.666	0.464
CBT[125]	0.767	1.290	0.808	0.676	0.261	0.419	0.675	0.477
*-Align	0.757	1.164	0.811	0.702	0.256	0.411	0.681	0.487
XPTrans	**0.715**	**0.722**	**0.821**	**0.820**	**0.236**	**0.374**	**0.705**	**0.533**
vs CBT(%)	↓6.8	↓44.0	↑1.6	↑21.3	↓9.6	↓10.8	↑4.5	↑11.7
vs *-Align(%)	↓5.5	↓38.0	↑1.3	↑16.8	↓8.0	↓9.0	↑3.6	↑9.4

表 4.8　所提出的 XPTrans 比起其他方法在跨平台行为预测试验中效果都好 (*: XPTrans)。



新浪微博 (社交标签, 微博实体)
→ 豆瓣 (读书, 电影, 音乐)

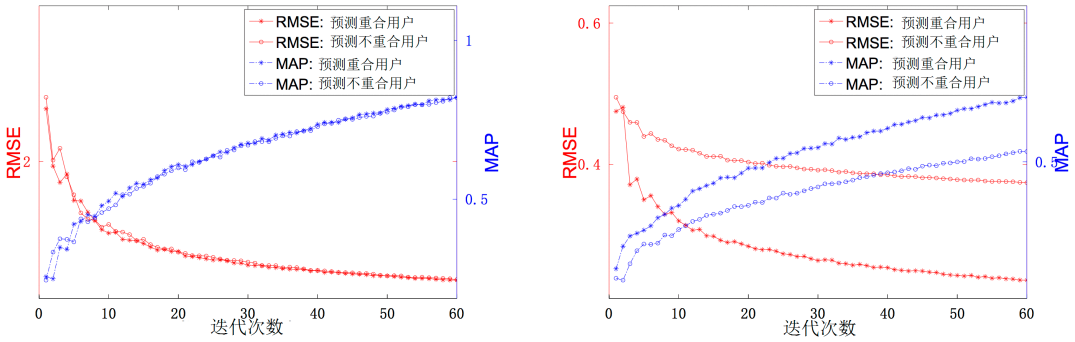
豆瓣 (读书, 电影, 音乐)
→ 新浪微博 (社交标签, 微博实体)

图 4.19　跨平台迁移学习可以通过学习辅助平台信息有效提升目标平台行为预测效果，尤其是在非重合用户上。

通过实现上述算法，调整参数到最优结果，然后与本文提出的方法比较。

实验中采用两个评价标准，分别为均方根误差（RMSE）[131,153] 和平均准确率（MAP）[83]。实验在两个任务上测试性能，一个是预测被随机保持的重合用户 ($P \cap Q$) 的行为，另一种是预测被随机保持的不重合用户 ($P \setminus Q$) 的行为。如果有越小的 RMSE 和越大的 MAP，就意味着预测效果越好。

设置目标平台的训练数据比例为 $\alpha_{\mathbf{R}^{(P)}}^{(P\setminus Q)}$=70%，辅助平台训练数据比例为 $\alpha_{\mathbf{R}^{(Q)}}^{(Q\setminus P)}$=70%。表4.8给出了所提出 XPTrans 算法和基线算法的 RMSE 和 MAP。这里展现从微博实体迁移学习豆瓣评分和从豆瓣读书迁移学习社交标签的结果。算法 XPTrans 能够提升非重合用户的行为预测效果。可以看到，在预测豆瓣电影评分时，XPtrans 比起先进算法能降低 RMSE 达到 44%，比 XPTrans-Align 要降低 RMSE 达 38%；能够比先进算法提升 MAP 达 21%，比 XPTrans-Align 提升 MAP 达 17%。在预测微博社交标签时，XPTrans 能提升将近 10%。结果证明辅助平台信息和重合用户监督项的积极作用。



微博实体到豆瓣电影　　豆瓣读书到微博标签

图 4.20　在将近 50 次迭代之后算法效果基本收敛：在两个迁移学习任务中，RMSE 值都随着迭代次数增加而逐步降低。

图4.19给出每一对目标平台 -辅助平台的 RMSE 降低比例，可以观察到：

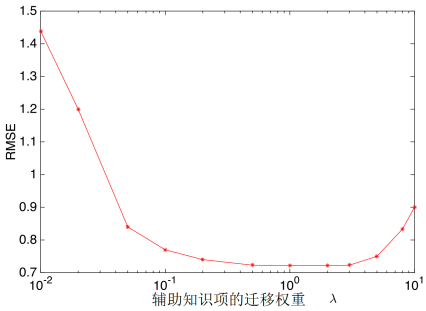
- 和表4.8相比，预测非重合用户行为的效果提升要比预测重合用户的效果提升要大。这说明虽然非重合用户的稀疏度问题更有挑战性，跨平台迁移学习能更有效地解决这一问题。
- 迁移学习微博的行为数据能降低预测豆瓣用户的评分行为的错误率 RMSE 达到将近 40%，这说明跨社交平台融合知识的行为模型是有效的。
- 迁移学习作为辅助平台的豆瓣读书/电影/音乐评分数据能降低预测社交标签的错误率将近 15%，但是预测微博实体的 RMSE 能降低得非常小。这说明预测微博实体的任务非常困难，跨平台迁移学习都很难提升预测效果。

图4.20显示非重合用户和重合用户的行为预测效果都会随迭代次数增加而变好，大约 50 次迭代达到最好。实验中设置最多的迭代次数为 100 以保证收敛。

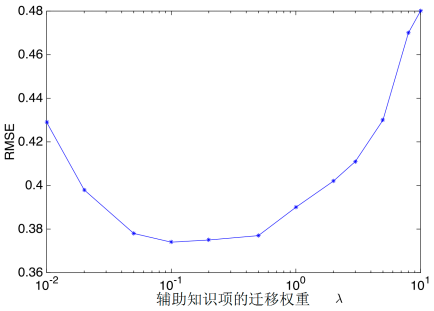
图4.21给出了改变辅助平台参数权重 λ 时对豆瓣电影和微博标签的预测效果。当 $\lambda \in [0.05, 0.5]$ 时，RMSE 能够达到最小。所以这里设置 λ 参数为 0.1。图4.21中给出了改变监督项参数权重 μ 时的预测效果。类似的，这里设置 μ 参数也为 0.1。

这里证明了越多的目标平台训练数据和越多的辅助平台信息能够提升预测效果。图4.22(a) 说明当 $\alpha_{\mathbf{R}^{(P)}}^{(P\backslash Q)}$ 增加时，RMSE 逐步降低。图4.22(b) 说明当 $\alpha_{\mathbf{R}^{(Q)}}^{(Q\backslash P)}$ 增加时，RMSE 也是逐步降低。辅助平台信息的增加所带来的错误率降低比起训练数据增加所带来的要慢，这说明虽然 XPTrans 很有效，但是如果能有更多的目标平台信息依旧比起辅助平台信息更加有价值。

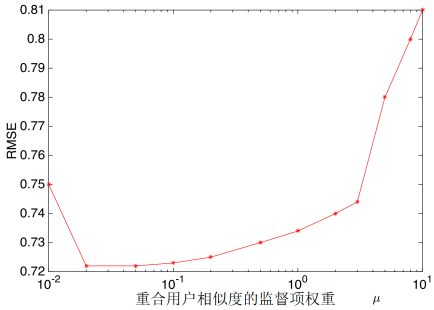
通过跨平台迁移学习，不重合用户的行为预测效果是否能够达到不通过迁移学习的重合用户行为预测效果？图4.8比较了通过改变 $\alpha_{\mathbf{U}}^{(P\cap Q)}$（活跃重合用户的比例）和改变 $\alpha_{\mathbf{R}}^{(P\cap Q)}$（重合用户的行为比例）时的预测效果变化。可以看到在迁移微博实体数据到豆瓣电影预测的情景下，采用 $\alpha_{\mathbf{U}}^{(P\cap Q)}$=26% 的活跃重合用户的行为



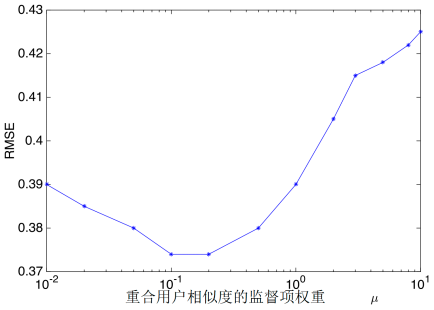
λ：微博实体到豆瓣电影



λ：豆瓣读书到微博标签

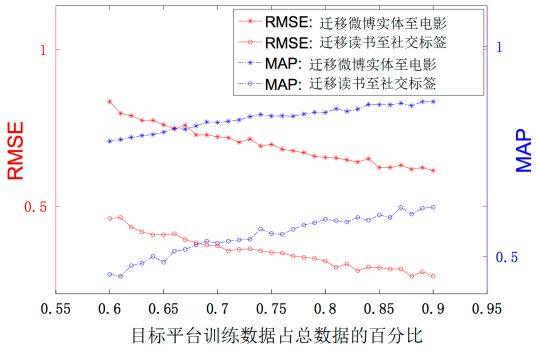


μ：微博实体到豆瓣电影

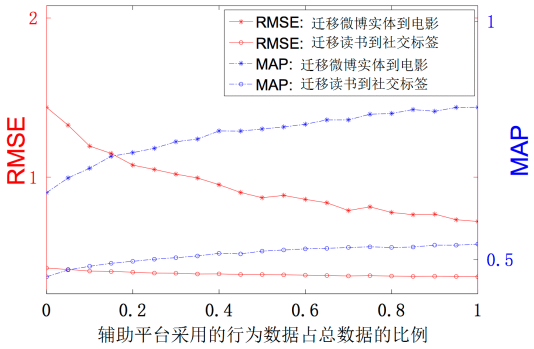


μ：豆瓣读书到微博标签

图 4.21　辅助平台参数 λ 和 μ 的设置：设置 λ=0.1 和 μ=0.1。



(a) 目标平台训练数据比例 $\alpha_{\mathbf{R}^{(P)}}^{(P\backslash Q)}$



(b) 辅助平台训练数据比例 $\alpha_{\mathbf{R}^{(Q)}}^{(Q\backslash P)}$

图 4.22　如果有更多的目标平台/辅助平台训练数据，预测效果能够更好（更小的 RMSE 和更高的 MAP）。

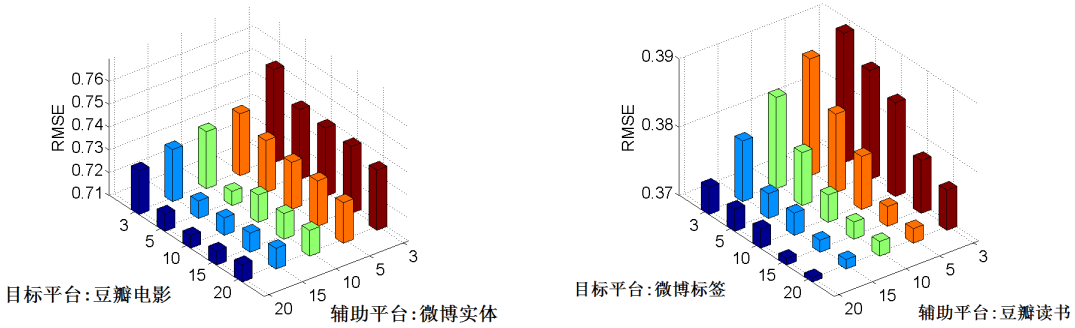
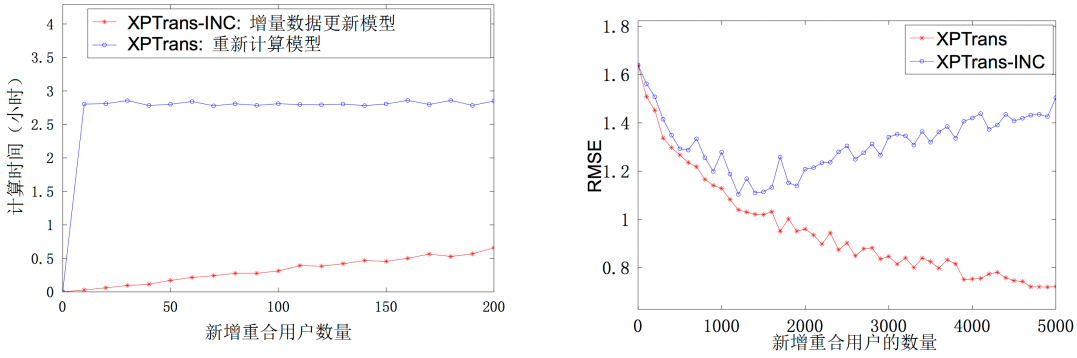


图 4.23　特征越多通常能够提升行为表征，提升预测效果：当 P 是豆瓣电影，Q 是微博实体时，如果 (r_P,r_Q)=(10,20) 或是 (5,10)，RMSE 能够达到最小；当 P 是微博标签，Q 是豆瓣读书时，如果 (r_P,r_Q)=(20,20) 或是 (15,20)，RMSE 能够达到最小。



(a) XPTrans-inc 计算效率很高　　(b) XPTrans-inc 的损失很小

图 4.24　XPTrans-inc 的计算效率高、损失很小：(a) 增量更新算法 XPTrans-inc 的运算速度很快；(b) 当增量用户不多的时候，损失很小。

或是采用 $\alpha_{\mathbf{R}}^{(P\cap Q)}$=60% 的重合用户行为数据，能够使得不重合用户的行为预测效果达到不迁移情况下的重合用户的行为预测效果（同样的 RMSE 和同样的 MAP）。

通过比较每一对目标平台 P 特征数量和辅助平台 Q 特征数量，也就是 $r_P, r_Q \in \{3,5,10,15,20\}$，可以看到行为预测效果。图4.23给出了预测豆瓣电影和微博标签的 RMSE 值。可以看到当 (r_P,r_Q)=(10,20) 或 (5,10) 时，预测电影评分的效果最好，当 (r_P,r_Q)=(20,20) 或 (15,20) 时，预测社交标签的效果最好。由图中可知

- 特征表示越多，通常情况下预测效果也就越好。
- 当 $r_P = r_Q$ 时，并不是总能达到最好的预测效果。可以看到微博实体表征往往需要比起豆瓣电影表征更多的特征。

表4.9和表4.10给出两种案例：(1) 从微博实体迁移知识到豆瓣电影评分，和 (2) 从豆瓣读书评分到微博社交标签预测。通过了解豆瓣读书和豆瓣电影中对应信息的最频繁标签，可以看到跨平台知识迁移的效果：

- **发布什么微博内容的用户会给什么样的电影打高分呢？** (1) 会发布关于“上

$\mathbf{V}^{(Q)}_{:,i}$ 表示的微博实体	$\mathbf{V}^{(P)}_{:,j}$ 表示的豆瓣电影	电影中最频繁的标签			
上海，中国，早安，艺术，教育，文化，香港，纽约，女性，法国，旅行，孩子	《罗马假日》	**经典**	**爱情**	**美国**	浪漫
	《茜茜公主》	**经典**	奥地利	**爱情**	自传
	《飘》	**经典**	**爱情**	**美国**	战争
	《魂断蓝桥》	**爱情**	**经典**	**美国**	战争
北京，乐队，音乐，青年，唱歌，晚安，老师，艺术，表演，朋友，同学，城市	《猜火车》	英国	**青春**	毒品	**故事**
	《阳光灿烂的日子》	**青春**	文化	改革	**故事**
	《老男孩》	**青春**	梦想	激情	中国
	《天堂电影院》	意大利	经典	成长	**故事**
推理，电影，新闻，北京，美国，用户，国家，出品，公司，政府，因特网，论坛	《第九区》	**科幻**	**美国**	**人性**	种族
	《黑客帝国》	**科幻**	**美国**	动作	故事
	《战争之王》	战争	**美国**	故事	凯奇
	《拯救大兵瑞恩》	真正	斯皮尔伯格	**人性**	**美国**

表 4.9　从微博实体迁移知识来预测豆瓣电影评分的案例分析。

海”、“女性” 和 “咖啡厅” 的微博用户往往会喜欢观看经典、关于爱情的、美国影片（上海是中国最国际化和浪漫的城市之一）。(2) 会发布关于 “北京”、“乐队” 和 “音乐” 的微博用户往往会喜欢观看关于青春的故事片。和上海相比，北京更因为长久的历史和文化传统而出名。大多数的中国摇滚乐队都是在北京成立的。(3) 会发布关于 “国家”、“政府” 和 “因特网” 的微博用户往往会喜欢观看科幻、关于人性的、美国影片。一种可能的解释是会谈论政治话题的人往往会关心人类的未来。

- **给什么样书籍打高分的用户往往会有什么样的社交标签？** (1) 会喜欢读经济和商业等方面书籍的豆瓣用户往往会有 “信息技术” 和 “程序员” 的社交标签。世界上的经济学家并不多，却存在着大量的程序员、码农。程序员不仅仅会阅读科技方面的书籍，也会有广泛的兴趣去关注创业公司和高新科技。(2) 会喜欢读关于爱情和青春等方面小说的豆瓣用户往往会有 “吃货” 和 “学生” 的社交标签。因为吃货和学生通常是喜欢读、也有时间读青春爱情小说的年轻人。(3) 会喜欢读日本动漫的豆瓣用户会有 “猫” 和 “猫奴” 的社交标签。他们会喜欢可爱的东西，而猫和日本动漫都会有很可爱的元素。

在迁移微博实体的知识来预测电影评分时，先是随机从 12,027 个重合用户中选取 5,000 个，然后在矩阵 $\mathbf{W}^{(P,Q)}$ 中注入增量的用户匹配信息，并且比较 XPTrans 和增量更新的近似算法 XPTrans-inc 的预测效果。图4.24(a) 展示了算法的运行时间。XPTrans-inc（不到 1 小时）比起 XPTrans（大约 3 小时）要快很多。图4.24 (b) 证实了 XPTrans-inc 在增量用户数量不到 1,000 的有效性：精度损失很小。然而，

$\mathbf{V}^{(Q)}_{:,i}$ 表示的书籍	书籍的最频繁标签			$\mathbf{V}^{(P)}_{:,j}$ 社交标签
《浪潮之巅》	因特网	信息技术	**商业**	**信息技术**，计算机，用户体验，**程序员**，安卓，跑步，广告
《货币战争》	金融	**经济**	货币	
《长尾效应》	**经济**	因特网	**商业**	
《Don’t Make Me Think》	用户体验	网页设计	Web	
《You Are My Sunshine》	**小说**	**爱情**	**青春**	宅男，**吃货**，**美食**，90 后，五月天，动漫
《梦里花落知多少》	郭敬明	**小说**	**青春**	
《致我们终将逝去的青春》	**青春**	**小说**	网络	
《龙珠》	**动漫**	**日本**	孩子	动漫，艺术，日本剧，**猫**，**猫奴**，ACG，J-POP
《死亡笔记》	**动漫**	推理	**日本**	
《灌篮高手》	**动漫**	篮球	**日本**	

表 4.10　从豆瓣读书迁移知识来预测微博标签的案例分析。

当增量用户比例超过总量的 5% 时，预测系统需要重新运行 XPTrans 做更新。

4.4　本章小结

本章节着重解决社会化推荐中的高稀疏度和冷启动问题，重新从迁移学习的角度思考这个问题，通过迁移辅助域知识来解决目标域稀疏度问题，提出了基于星状图的新颖的混合随机漫步算法，该算法可以融合复杂的多元异质的链接结构。大规模的真实社交媒体上充分实验表明所提出的方法能够大大提升推荐效果。混合随机漫步算法还能从社会标签域提取知识，通过更新社交域中纽带强度来迁移到微博域中。和之前只是用微博域中知识的方法相比，只需要目标域中的 27.6% 的知识就可以利用迁移学习达到与完全使用目标域但不适用辅助信息域比同样的效果。这一方法给解决高稀疏度和冷启动问题提供很有意义的参考。

本章节中还进一步提出了用辅助平台行为数据来解决目标平台稀疏度的 “跨平台行为预测” 问题。工作中研究两个真实社交平台中的 5 个不同的信息域。平台间的重合用户对于跨平台知识迁移具有非常重要的意义。由此，本文给出了半监督迁移学习算法 XPTrans 来更精准地在目标平台上预测用户行为。实验证明 XPTrans 可以有效地从辅助平台迁移知识提升目标平台的预测效果。XPTrans 支持购物应用从社交媒体中迁移知识，支持健康医疗应用从可穿戴设备中迁移知识。

第 5 章　社交媒体可疑行为分析方法和评价指标

本章从同步性、密集连接模式和跨维度密集性的角度介绍社交媒体可疑行为分析方法和评价指标。首先介绍基于同步性的可疑行为检测算法，接着介绍基于密集连接模式的可疑行为检测算法，然后介绍跨维度行为可疑程度的通用指标，第四小节介绍性能评测结果，并在最后小结本章内容。

5.1　基于同步性的可疑行为检测算法

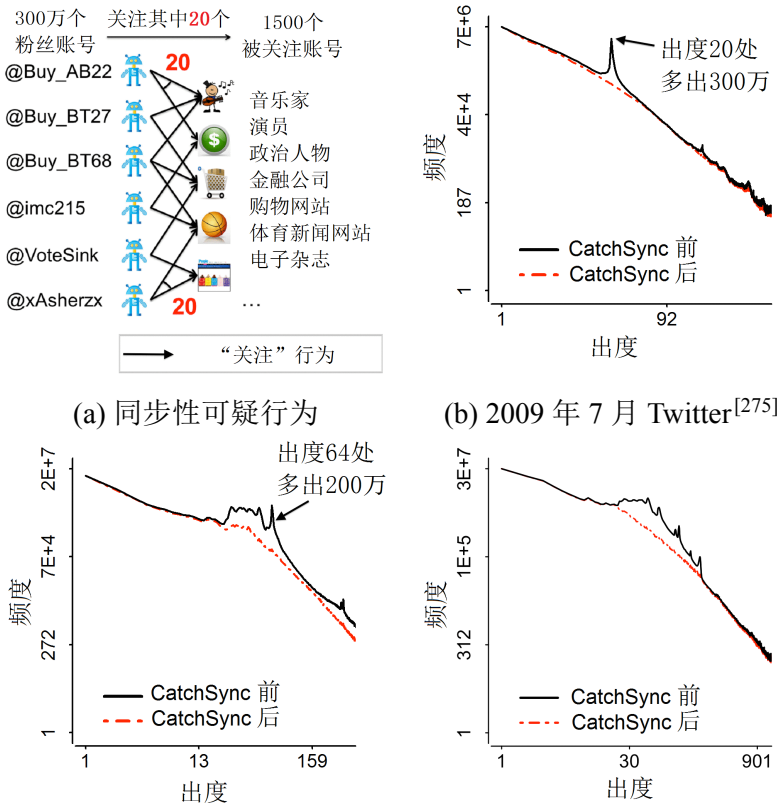
本节介绍基于同步性的可疑行为检测算法。内容包括引言、相关工作、可疑行为的同步性分析以及基于行为同步性的可疑用户检测算法。

5.1.1　本节引言

在社交媒体上，用户相互关注的行为会形成大规模的有向图。给定一个含有百万节点的有向图，是否能够从图结构中知道哪些节点是可疑的？在许多真实应用中欺诈者操纵网络结构以获得利益。比如 Twitter 的 “谁关注谁” 的有向图中，欺诈者因产生很多虚假粉丝账户，并使他们关注顾客，让顾客们看上去合法或是受欢迎而得到钱款 ①。这种现象破坏了整个社交媒体的流行性和可信度，会给诚实用户带来非常不愉快甚至是危险的感受。这种操纵 Twitter 网络的攻击需要同步地给有向图增加原本不会存在的边。传统的在 Twitter 上检测可疑行为的方法往往是分析用户的微博内容和个人信息内容[183,185,276]，并不考虑同步性可疑行为造成的结构上扭曲，所以很容易错失这些可疑行为。本工作设计了从图中的异常结构特征入手，检测用户节点是否可疑的图挖掘算法。在现实生活中该图挖掘算法能检测到多种多样应用中类似的可疑行为。例如，存在着一大群机器同时发出成千上万的请求，对网站进行分布式服务攻击（Distributed Denial of Service，简称 DDOS），在 “谁访问谁” 的图中形成同步性的行为模式。Amazon 和 Yelp 等在线网站中的垃圾散播者会操纵 “谁给谁打分” 的图，为了给特定的商品或者是餐馆修改评分，添加很多并不应当存在的边。Facebook 的主页拥有者会付款给垃圾传播者来 “喜欢” 他们的页面，提升自己表面知名度的同时，扭曲了 “谁喜欢哪个页面” 的图。

本文关注于检测 Twitter 中的僵尸粉：他们成组的发生同步性行为，使顾客得

① Buy Twitter Followers. http://www.buy-followers.org; Buy Twitter Accounts. http://www.buytwitteraccounts.org



(a) 同步性可疑行为　(b) 2009 年 7 月 Twitter[275]

(c) 2011 年 1 月腾讯微博　(d) 2011 年 11 月腾讯微博

图 5.1　僵尸粉的同步性行为及其留下的踪迹：(a) 可以看到百万级的 Twitter 账号关注同一组的顾客，形成了同步性行为；(b) 同步性行为在出度分布上形成了尖峰异常，在移除值得怀疑的账号之后，出度分布变得平滑。在腾讯微博上也取得了同样效果，无论是在 (c) 2011 年 1 月份的数据还是 (d) 2011 年 11 月份的数据。

到虚假知名度。图5.1(a) 展示了一组可疑的粉丝团体和他们所关注的人。这些粉丝总共有 300 万，从 1500 个的用户群体中选择刚好 20 个去关注，形成了一种罕见的同步性连接结构。图中所示的 Twitter 注册名相似性（@Buy_AB22, @Buy_BT27, @Buy_BT68）等附加信息是怀疑他们用脚本产生的有力证据。本工作从新颖的行为视角、而非内容角度来看待这一问题。从细节上说，包括僵尸粉、机器人等的可疑节点在网络中会形成这样的行为模式：(1) 同步性行为（同等节奏地发生），他们通常会同时关注同等数量，比如 20、64 或者 100 个目标用户；(2) 异常的/罕见的行为，他们的行为模式和大多数节点非常不同。本文给出快速有效的方法 CatchSync 来衡量两大行为特征，即一组用户节点的同步性和正常性：通过观察在同步性 -正常性绘图（Synchronicity-Normality Plot）中的可疑节点，实现有效检测。本工作中主要关注三大真实的、完整的社交网络数据，包括 2009 年的 Twitter、2011 年 1 月的腾讯微博和 2011 年 11 月的腾讯微博（分别简记为 TwitterSG、WeiboJanSG 和 WeiboNovSG）。这些社交媒体都有百万级的用户节点

和上十亿级的关注边数。

图5.1给出 CatchSync 方法的有效性证明。如之前提及的，社交媒体数据的分布会因为大规模的可疑粉丝而被严重扭曲。这里用 “对数 -对数” 规模绘制出 Twitter、WeiboJan 和 WeiboNov 的出度分布（黑色线条）。这些分布应该是顺滑的、符合幂律的分布[258]。但是图中显示了可疑粉丝产生的尖峰异常[269]。例如在图5.1(a) 中，有 300 万的 Twitter 粉丝关注着 20 个相近的用户，并在图5.1(b) 中的出度分布上产生了一个出度为 20 的尖峰。在移除了 CatchSync 所标记的可疑节点，出度分布变得更顺滑，并且接近于幂律分布（红色线条）。类似的现象也可以在腾讯微博数据，即图5.1(c-d) 中看出。

本工作的主要贡献在于。提出了基于同步性的 CatchSync 方法来捕捉大规模图中的可疑行为。该方法具有使用者想要的下列特性：

- 有效性：确实能够观察到产生可疑行为的源用户 -目标用户群。工作中用多个强有力的证据证实了方法的有效性，包括出度分布和特征空间重新回归正常，正确地将人工标注数据上的正常、异常用户分类，以及重点样例分析。
- 可扩展性：该方法计算复杂度与图中边数呈线性关系，适用于因特网规模的图。
- 无需参数性：使用者并不需要确定额外的参数信息，比如密度下限、组群数量和组群大小。
- 无需附加信息：CatchSync 方法并不需要附加信息。这个方法只与拓扑结构有关，不需要手动标注信息作为训练，也不需要节点属性知识。后续会介绍到，本方法可以融合这些知识以期更好的效果。

5.1.2　相关工作

本小节中综述三方面的相关工作：基于图的异常检测、子图挖掘算法和社交垃圾信息传播者检测。表5.1已经从方法的有效性、参数设置和附加信息三个角度讨论了其中大部分算法，并展示本文提出 CatchSync 算法的很多优势。

研究者们已经提出了许多基于图的异常检测算法[169,170,292]。AutoPart[200] 能够把相似的节点归为同样的类，并且把指向这些节点的边标记为异常边。然而实际情况是常常没有足够的可疑节点和正常节点间相似性信息。基于图的最近工作通过分析结构上的异常现象提出了如何检测可疑节点和边的算法，这些方法能够把置信度值在可疑节点之间传播[175,203,212,245]。OutRank[204] 是基于图上随机漫步的算法通过学习节点之间相似性来检测异常。OddBall[205] 给出与 “相近无向子图” 相关的密度、权重、排序和特征值规则，并假设近完全图和星状图是可疑的。NetProbe[176] 采用信任传播 (belief propagation) 算法借助已知的可疑节点信息，从

		检测同步性？	无需参数？	无需附加信息性？
提出的 CatchSync		√	√	√
基于图的异常检测	AutoPart	×, k 个节点组	√, 自动选取 k	√
	OutRank	×, 高分节点	×, 需要阈值	√
	Oddball	×, 近乎全连图	√	√
	CopyCatch	√, 时间密集	×, 需要种子	×, 时间戳
	NetProbe	×, 欺诈节点	×, 传播	×, 已知的欺诈点
子图挖掘	METIS	×, k 个等大子图	×, k	√
	SpokEn	×, 密集相连	×, 特征向量	√
	DSE	×, d_G-密度的	×, 密度 d_G	√
垃圾信息检测	SPOT	×, 垃圾传播者	√	×, 文本特征
	SybilRank	×, 攻击者	×, 种子	×, 早期非攻击者

表 5.1　将 CatchSync 与已有的算法作比较：CatchSync 能够不受组数、子图密度、起始信任度的限制。该算法不需要任何参数设置或者是附加信息。

图中检测所有的可疑节点。CopyCatch[190] 能够从 Facebook 中可疑的 “喜欢 (Like)” 页面标记检测出时间信息形成的二部密集图；这个方法需要边形成的时间作为附加信息。可疑用户的伪装算法能够让行为密度并不是非常高。本文所提出的方法是与上述不同的，本工作寻找形成可疑的子图结构的同步性行为。

研究者还开发了一系列的子图挖掘算法[201]。METIS[197] 通过减小不同子图之间边的数量，能够把图中的节点分割为 k 个相同大小子图。CloseGraph[217] 通过剪枝算法来减小不必要的子图，设计了频繁图模式的挖掘算法。Crochet[218] 分析了寻找跨图的近完全图的算法，子图中每个节点都与至少 γ 比例的其他节点相连。CloseMine[174] 分析了基于图的频繁子图分类和闭合频繁子图分类算法的关系。MUSE[224] 在不确定图的数据库中挖掘子图模式。最近有一些社区发现的方法广泛应用于复杂网络科学中。社区可疑往往被看作图中独立的组成部分[65,238]。EigenSpokes[239] 找到了图中的奇异值向量表示一种特殊的 “spoke” 模式，基于 spoke 模式的算法能够找到紧密社区。D-core 算法[225] 能够检测密集连接的社区，并拓展成经典的图理论中的 k-密集无向图到有向图场景。密集子图的一些研究成果说明不是每一个子图中的节点都属于这个社区[227]。DSE[228] 展现出知名的最密集子图通常是大规模图，却有较小的边密度和大直径，于是给出了局域搜索的近似算法。子图挖掘算法的弱点是它们通常需要参数，比如说密度和类别数量作为输入。另一方面是可疑的节点通常很容易就能绕开高密度的检测算法，比如削减目标节点的数量或者是源节点的规模。

近年来，研究者们提出了很多社交垃圾信息传播者的算法[182,276]。社交蜜罐

（social honeypot）的方法[180,193] 能够从社交媒体社区中得到垃圾传播者的大量个人信息，用统计方法得到了垃圾信息传播者的分类器。SPOT[183] 从微博内容中学习基于文本内容的特征以及可疑链接并检测可疑的 Twitter 用户个人信息，并给他们打出可疑程度的分数。SybilRank[185] 依靠社交图的属性来通过学习这些节点的属性给他们排序。SSDM[191] 是一种基于社交逻辑性的用社交媒体信息和基于内容信息检测垃圾传播者的方法。被购买的 Twitter 粉丝账户与随机选出的 Twitter 用户在交互行为和内容分享的模式上差别很大[192]。然而，本工作与过去的方法在检测攻击时不同，并非从内容表象这一并非不得不发生的事情出发，而是考虑行为模式这一僵尸粉无法逃避的事实。工作中发现 Twitter 或是类 Twitter 的社交媒体上僵尸粉往往有着同步性的用户行为模式。

总之，本文提出的新方法 CatchSync 是无需调整参数、无需附加信息的方法，这个方法检测具有同步性行为的可疑节点非常有效。

5.1.3　可疑行为的同步性分析

本小节会提出同步性行为检测问题，目标是在有向图中检测可疑节点，问题的定义如下：给定在节点集合 $\mathcal{U}$ 的 N 个节点的有向图，寻找诸如虚假粉丝、机器人的可疑源节点集合 $\mathcal{U}_{sync}$ 和诸如被关注的人、目标主机的目标节点集合 $\mathcal{V}_{sync}$，这些节点形成了同步而又异常的连接模式。

符号	含义/描述
$\mathcal{U}$; N=$\|\mathcal{U}\|$	节点集合; 节点数量
$\mathcal{I}(u)$; $\mathcal{O}(u)$	u 的源节点集合; u 的目标节点集合
$d_i(u)$=$\|\mathcal{I}(u)\|$	u 的入度（源节点数量）
$d_o(u)$=$\|\mathcal{O}(u)\|$	u 的出度（目标节点数量）
$hub(u)$; $aut(u)$	u 的枢纽度; 权威度
$sync(u)$; $norm(u)$	u 目标节点的同步性; 正常性
$\mathbf{p}(u)$	u 的 k 维度的特征向量
$c(u,v)$	u 和 v 在特征空间中的相似性

表 5.2　数学符号和定义描述

同步性是指节点相互之间有非常相似的行为模式，异常性是说这些行为模式与图中主要节点的模式非常不同。表5.2中给出数学符号及其定义描述。解决上述问题的步骤如下：首先给出目标节点设计特征空间。第二步定义同步性和正常性来量化节点行为模式。同时对同步性 -正常性图的形状给出通用的理论证明，最后采用基于距离的异常检测算法找到图中的可疑节点。

过去基于图的数据挖掘工作中从节点的行为模式中得到很多特征方面的启示，包括 (a) 出度和入度，(b) HITS 值，包括枢纽度（hubness）和权威度（authoritativeness），(c) 中介性（betweenness）和核心性（centrality），(d) 带权图中的节点的向内权重和向外权重，(e) 图的邻接矩阵的第 i 个左奇异值向量和右奇异值向量。用 $\mathbf{p}(u) \in \mathbb{R}^k$ 定义了节点 u 的 k 维度特征向量。从图结构中抽取反映节点行为特征的特征向量，这些特征可以是上述所有特征或者是任意维度的特征。本文中选择度数和 HITS 值，除却容易计算和绘图外，下述特征在异常检测中能够保证可解释性：

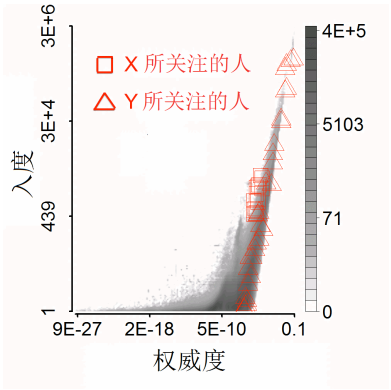
- 度数（出度和入度）：用 $\mathcal{I}(u)$ 定义 u 的源节点集合，用 $\mathcal{O}(u)$ 定义 u 的目标节点集合。节点 u 的入度 $d_i(u)$ 是源节点的数量，也就是 $\mathcal{I}(u)$ 的大小。节点 u 的出度 $d_o(u)$ 是目标节点的数量，也就是 $\mathcal{O}(u)$ 的大小。在社交媒体，大的入度是说存在许多粉丝账户，而且这些顾客从这种卖粉丝的服务中得到相似的入度值，而这些入度值比起普通用户的要大很多，但比起真实知名的名人要小很多。机器人或者是僵尸粉通常有相似的出度值，因为这些服务很容易会让机器人账户关注同样数量的顾客。当机器人账户的出度值更小，他们不仅更聪明了，但还是有同步性行为。另外，度数值与之前发现的尖峰有关。
- HITS 值（枢纽度和权威度）：参照 Kleinberg 的著名工作[283]，用 $hub(u)$ 定义节点 u 的枢纽度，用 $aut(u)$ 定义节点 u 的权威度。把 “粉丝 -关注的人” 网络构建成邻接矩阵，第一个左奇异值向量含有粉丝节点的枢纽度，第一个右奇异值向量含有关注的人的权威度。一个所关注的人如果有很多粉丝，那么他会有很高的权威度。连接了知名名人的粉丝比起普通用户会有更高的枢纽度。僵尸粉通常会比有同样出度值的用户有更小的枢纽度，因为顾客往往比起知名的名人要不那么出名。顾客比起有同样入度值的用户有更小的权威度，因为他们大多数的粉丝都是在网络中并不重要的僵尸粉。

这两组特征代表了社交媒体中的用户行为模式。实验中这些特征能够很好地找到可疑节点。要注意的是如果存在附加信息可以使用，这个方法是能够很容易的融入这些附加特征的，而检测效果会更好。

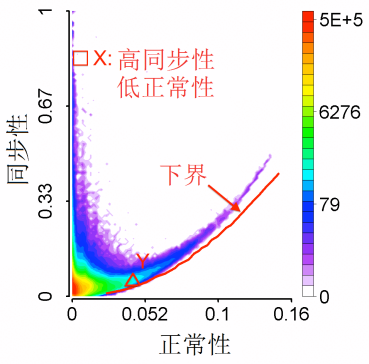
绘图名	描述
OutF-plot	源节点的特征热度图（出度 vs 枢纽度)
InF-plot	目标节点的特征热度图（入度 vs 权威度）
SN-plot	源节点的目标节点群的同步性 -正常性图

表 5.3　绘图名和描述

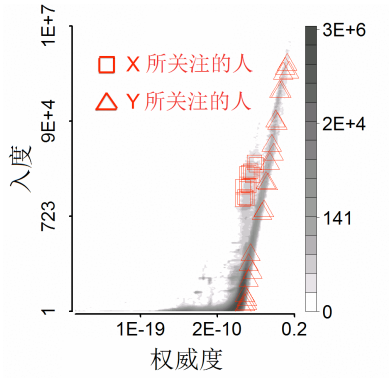
接下来给出特征空间中一些绘图的定义。给定源节点 u，画出用出度 $d_o(u)$ vs 枢纽度 $hub(u)$ 的对数下二维特征空间，记作 “OutF-plot”。相似的，给定目标节点 u，画出入度 $d_i(u)$ vs 权威度 $aut(u)$ 的对数下二维特征空间，记作 “InF-plot”。表5.3总结了上述绘图的名字和描述。



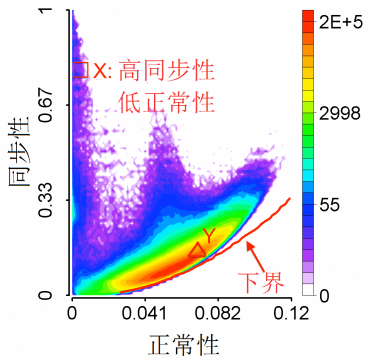
(a) TwitterSG 中的 InF-plot



(b) TwitterSG 中的 SN-plot



(c) WeiboJanSG 中的 InF-plot



(d) WeiboJanSG 中的 SN-plot

图 5.2　“同步性 -正常性” 图：源节点 X 因为他的目标用户在 InF-plots 的 (a) 和 (c) 中特别相似，所以 X 产生同步性异常行为；而 Y 的目标用户不同。X 在 InF-plots 的 (b) 和 (d) 中有比较大的同步性和小的正常性，而 Y 则更接近抛物线的下界。

特别的，图5.2(a) 和图5.2(c) 是 TwitterSG 和 WeiboJanSG 中的 InF-plots。在 TwitterSG 中用 X 标记为可疑的粉丝之一，用 Y 标记为和 X 有同样的出度的普通用户。把它们的目标节点（关注的人）标记在图5.2(a) 中，发现 X 的目标节点在 InF-plot 中相近，但 Y 的目标节点不相似。换句话说，X 的目标节点有相似的入度和权威度，但是 Y 的目标节点有的是非常知名，有的就是普通用户，所以在特征空间中非常不同。在 WeiboJanSG 中能观察到相似现象。图5.2(c) 中看到 X 的目标用户落在距离主体部分很远的小点簇中。这些点往往有从 1,000 到 100,000 的入度值，但是这些人并不如同样入度值下的普通用户一样有名（这些人偏左）。由此提出了研究源节点行为模式的两个概念：(a) “同步性” $sync(u)$ 来量化 u 的目标节点

在特征空间（入度 vs 知名度）的同步性特征；(b) “正常性” *norm*(*u*) 来量化 *u* 的目标节点在特征空间（出度 vs 枢纽度）的正常性特征。定义 **p**(*v*) 来表示目标节点 *v* 归一化的特征向量，定义 *c*(*v*,*v*′) 为目标节点 *v* 和 *v*′ 在特征空间（InF-plot）中的相似程度。于是有

$$c(v, v') = \mathbf{p}(v) \cdot \mathbf{p}(v') \tag{5-1}$$

要快速计算每一对节点的相似度，把特征空间分割为 *G* 个格子 (grid cell)，并且把每一个节点映射到特定的格子中。如果两个节点在同一个格子中，他们也就有相似的特征向量，在特征空间中是相近的。于是有

$$c(v, v') = \begin{cases} 1 & \text{如果节点 } v \text{ 和 } v' \text{ 在同一个格子} \\ 0 & \text{否则} \end{cases}$$

由此定义同步性和异常性。

定义 5.1： 同步性和异常性 定义节点 *u* 的同步性为 *u* 的每一对目标节点 (*v*,*v*′) 之间的相似程度：

$$sync(u) \quad = \quad \frac{\sum_{(v,v') \in O(u) \times O(u)} c(v, v')}{d_o(u) \times d_o(u)} \tag{5-2}$$

定义节点 *u* 的正常性为 *u* 的一个目标节点和全网络的某一个随机节点 (*v*,*v*′) 之间的相似程度：

$$norm(u) \quad = \quad \frac{\sum_{(v,v') \in O(u) \times \mathcal{U}} c(v, v')}{d_o(u) \times N} \tag{5-3}$$

同步性和正常性值是从 0 到 1 的。由此知道可疑的源节点 *u* 往往是有相当大的 *sync*(*u*) 和极其小的 *norm*(*u*)：

- 极其大的 *sync*(*u*): 很大的 *sync*(*u*) 值说明在特征空间里会存在一大群节点和节点 *u* 有同样的特征。如果粉丝节点 *u* 有一个很大的 *sync*(*u*) 值，那么 *u* 和其他节点一样会有相似的关注行为（相似的出度和相似的枢纽度）。也就是与同一组关注的人（节点）连接。僵尸粉往往会有极大的同步性值。
- 极其小的 *norm*(*u*): 很小的 *norm*(*u*) 值说明在特征空间里节点 *u* 相比于大多数节点是异常节点。如果粉丝节点 *u* 有较小的 *norm*(*u*) 值，相比于社会媒体中的大多数粉丝节点来说，*u* 会有很不同的关注行为模式（很不同的出度和不同的枢纽至）。也就是说节点 *u* 几乎不与大多数节点相联系。僵尸粉往往给出很小的正常性值。

数学标记	定义描述
G	特征空间中的格子数目
$g = 1, \ldots, G$	特征空间中的格子 ID
F	前景点的总数量
B	背景点的总数量
f_g ($\hat{f}_g$)	格子 g 中归一化的前景点数量
b_g ($\hat{b}_g$)	格子 g 中归一化的背景点数量
$\vec{f}$ ($\vec{\hat{f}}$)	G 长度的概率向量，描述归一化的前景点数量
$\vec{b}$ ($\vec{\hat{b}}$)	G 长度的概率向量，描述归一化的背景点数量
n	和 $\vec{b}$ 中的背景点相比，在 $\vec{f}$ 中的前景点的正常性
s	和 $\vec{f}$ 中的前景点相比，在 $\vec{f}$ 中的前景点的同步性

表 5.4　定理中使用的数学标记和定义描述

对于一个源节点 u，称 InF-plot 中的 u 的目标节点为“前景点”，称图中所有点为“背景点”。由此可以给出 SN-plot 标准形状的定理，这是可疑节点检测算法有效性的基础。表5.4中给出了数学标记和描述。

定理 5.1：　对于任何的前景点/背景点的分布来说，在同步性 -正常性图中存在二次曲线的下界。

证明 为了找到给定正常性后的同步性的下界，给出下面这个问题。给定 G 个格子，以及前景点和背景点的数目的概率向量 $\vec{f} =< f_1, \ldots, f_G >$ 和 $\vec{b} =< b_1, \ldots, b_G >$ 其中 f_g 和 b_g 是前景点和背景点在给定格子 g ($g = 1, \ldots, G$) 中的点数目（$f_g \le b_g$），以及向量 $\vec{f}$ 的正常性 n，找到最小化同步性 s 的向量 $\vec{f}$ 值。

要注意的是 (1) 同步性 s 是前景点的同步性，也就是前景点向量和它们本身的点积：$s = \sum_g \frac{f_g^2}{F^2}$；(2) 正常性 n 是前景点向量和背景点向量的点积 $n = \sum_g \frac{f_g b_g}{FB}$。

记 $B(F)$ 为总数量：$\sum f_g = F$ and $\sum b_g = B$。记 $\hat{b}_g = b_g/B$ 和相似度 $\hat{f}_g = f_g/F$. 那么概率向量的总和为 1。问题的定义可以更新为：给定含有 G 个值的概率向量 $\vec{\hat{b}}$，找到概率向量 $\vec{\hat{f}}$，当给定正常性 $n = \vec{\hat{f}} \cdot \vec{\hat{b}} = \sum(\hat{f}_g * \hat{b}_g)$ 和最小的同步性 $s = \vec{\hat{f}} \cdot \vec{\hat{f}} = \sum \hat{f}_g^2$，并且找到最优化的向量 $\vec{\hat{f}}_{opt}$，以及最小的同步性值 $s_{\min}$。

拉格朗日乘积 (Lagrange multiplier) 的方法被广泛使用在存在等式约束下的函数最小值（最大值）。这里拉格朗日函数是

$$\mathcal{F}(\hat{f}_g, \lambda, \mu) = (\sum_g \hat{f}_g^2) + \lambda(\sum_g \hat{f}_g - 1) + \mu(\sum_g (\hat{f}_g * \hat{b}_g) - n) \tag{5-4}$$

函数的梯度为

$$\frac{\partial \mathcal{F}}{\partial \hat{f}_g} = 2\hat{f}_g + \lambda + \mu\hat{b}_g = 0 \quad g = 1, \ldots, M \tag{5-5}$$

以及两个初始的约束条件

$$\frac{\partial \mathcal{F}}{\partial \lambda} = \sum_g \hat{f}_g - 1 = 0 \tag{5-6}$$

$$\frac{\partial \mathcal{F}}{\partial \mu} = \sum_g (\hat{f}_g * \hat{b}_g) - n = 0 \tag{5-7}$$

根据上述等式，两边加起来就得到

$$2 + G\lambda + \mu = 0 \tag{5-8}$$

如果在两边同时乘以 $\hat{b}_g$，再加起来为

$$2 * n + \lambda + \mu s_b = 0 \tag{5-9}$$

其中叫 s_b 为背景点的同步性：$s_b = \sum_g \hat{b}_g^2 = \sum_g \frac{b_g^2}{B^2}$. 解出 μ 可知

$$\mu = -2 - G\lambda \tag{5-10}$$

那么代入 μ 就知道 λ：

$$\lambda = 2(s_b - n)/(1 - G * s_b) \tag{5-11}$$

代入 μ 和 λ 就可以解出 $\hat{f}_g$，或是在等式两边同时乘以 $\hat{f}_g$ 再加起来，就得到：

$$2 * s + \lambda + \mu n = 0 \tag{5-12}$$

于是最优化的 s_{opt} 满足

$$s_{opt} = 1/2(-\lambda - \mu n) \tag{5-13}$$

如果 Hessian 矩阵在这一点上是正数矩阵，那么函数就是凸函数并且存在局域最小值。对于前 M 个值，Hessian 矩阵是以 2 为对角元的对角矩阵，而其他地方都是 0. 最终可知同步性的下界是

$$\boxed{s_{min}(n) = (-Gn^2 + 2n - s_b)/(1 - Gs_b)} \tag{5-14}$$

那么给定正常性 n，同步性最小值 s_{min} 是 n 的二次函数。 □

讨论如果正常性的值变小接近 0 的时候的最小同步性。如果 $n = 0$，知道 $s_{min}(0) = \frac{-s_b}{1-Gs_b}$。因为背景点的同步性是 $s_b = \sum_g \hat{b}_g^2 = \sum_g \frac{b_g^2}{B^2} \in [0, 1]$, 有 $s_{min}(0) \in$

$[0, \frac{1}{G}]$. 如果 G 是很大的数字，比如 100 或者 1,000，那么 $s_{min}(0) \to 0$. 二次函数接近形式 "$y = ax^2$"。图5.2(b) 和图5.2(d) 给出 TwitterSG 和 WeiboJanSG 中源节点的 SN-plots。根据图5.2(a) 和5.2(c) 知道，因为源节点 X 会有同步性的异常行为，而 Y 并没有。那么比起节点 Y 来说，X 有大很多的同步性和小很多的正常性。证明中给出了红色的抛物线是给定正常性下同步性程度的理论下界。所以 Y 靠近抛物线，而 X 离下界很远。下一步就是如何从 SN-plots 中检测到和节点 X 一样的可疑节点。根据前面的定理，定义了如何在 SN-plots 中检测异常检点。节点离下界很远，那么这个点就是异常节点。可以定义过剩值 $r_{source}(u)$ 为

$$r_{source}(u) = sync(u) - s_{min}(norm(u)). \tag{5-15}$$

过剩值能够描述点的同步性距离理论下界有多远，也就表示这个节点有多可疑。细节上讲，给定节点 u，计算同步性值 $sync(u)$ 和正常性值 $norm(u)$。给定正常性值 $norm(u)$，根据上式知道同步性值 $s_{min}(norm(u))$ 的下界。通常，一个普通节点会有较小的同步性值，也就是说 $sync(u) - s_{min}(norm(u)) = \epsilon$ （ϵ 是比较小的数）。如果点 u 有着非常大的同步性值，也就是说，如果 $sync(u) \gg s_{min}(norm(u))$，也就是说 $r_{source}(u) \gg 0$，那么 u 是一个可疑节点，并且行为模式非常同步。可疑源节点集合 $\mathcal{U}_{sync}$ 包括过剩值距离均值 $\alpha = 3.0$ 倍标准差的节点

$$\mathcal{U}_{sync} \leftarrow \{u : r_{source}(u) > \mu[r_{source}] + \alpha \times \sigma[r_{source}]\} \tag{5-16}$$

用 $\mu[r_{source}]$ 标记所有源节点的平均过剩值，用 $\sigma[r_{source}]$ 标记过剩值的标准差。相似的，用 $r_{target}(v)$ 标记目标节点 v 的过剩值，能够描述 v 的粉丝群有多可疑。接着给出可疑目标节点集合 $\mathcal{V}_{sync}$：

$$\mathcal{V}_{sync} \leftarrow \{v : r_{target}(v) > \mu[r_{target}] + \alpha \times \sigma[r_{target}]\} \tag{5-17}$$

α 的默认值是根据 Tax 的经典异常检测工作[168] 选取的。实验中证实了 CatchSync 算法的效果对于选择 α 完全不敏感。

5.1.4　基于行为同步性的可疑用户检测算法

本小节在算法6中给出 CatchSync 的实现方法并分析复杂度。首先，给目标节点生成特征空间，然后根据目标节点在特征空间的相对位置，计算源节点行为的同步性和正常性。最后，用基于距离的异常检测算法来检测同步性 -正常性图中的异常节点。

更准确的说，给源节点选取了 "出度 vs 枢纽度" 的二维特征空间，给目标节点选取了 "入度 vs 权威度" 的二维特征空间。图的邻接矩阵形成的第一个左 (右)

Algorithm 6 CatchSync: 在大规模图中检测同步性行为中的可疑节点

Require: 有向图，含有 N 个节点的节点集合为 $\mathcal{U}$

1: 第一步：构建目标节点的二维特征空间。
2: **for** 每一个目标节点 v **do**
3:　计算入度 $d_i(v)$ 和权威度 $aut(v)$
4: **end for**
5: 给出 InF-plot：$d_i(v)$ vs $aut(v)$
6: 第二步：构建源节点的同步性 -正常性图。
7: **for** 每一个源节点 u **do**
8:　计算同步性 $sync(u)$ 和正常性 $norm(u)$
9: **end for**
10: 给出 SN-plot：$sync(u)$ vs $norm(u)$
11: 第三步：采用基于距离的方法检测可疑源节点集合 $\mathcal{U}_{sync}$ 和目标节点集合 $\mathcal{V}_{sync}$。
12: **输出:** 具有同步性异常行为的源节点集合 $\mathcal{U}_{sync}$ 和目标节点集合 $\mathcal{V}_{sync}$。

奇异值向量是枢纽度 (权威度)。计算这些值的算法在这里被省略。在第二步里把 InF-plot 画成格子，并且找到对数空间里的基数 b，并确定格子长度 L，那么格子的位置是从 b^{aL} 到 $b^{(a+1)L}$，对于每一个整数 a。设定 $L=1$ 和 $b=2$，即格子位置是 2 的幂数，也就是把枢纽度和权威度分为 $2^0, 2^{-1}, 2^{-2}, \ldots$，把出度和入度分为 $2^0, 2^1, 2^2, \ldots$。可知计算度数和 HITS 值的复杂度是与边数量 E 呈线性关系的。第二步计算同步性和正常性的方法也是与 N 呈线性关系的。标记格子数量为 G，那么时间复杂度为 $O(E+NG)$。所以，可扩展的 CatchSync 能够处理大规模有向图。

用 $p(v)$ 标记归一化的特征向量 $\mathbf{p}(v)$，而这些特征是对数分布的。如果特征空间里的格子长度是 b 的幂数，比如说 $b=2$，那么就存在两个整数 a 和 a' 来估计 $p(v)$ (节点 v 的特征) 和 $p(v')$ (节点 v' 的特征)，因为

$$b^{aL} \le p(v) < b^{(a+1)L}, b^{a'L} \le p(v') < b^{(a'+1)L} (a \le a' < 0) \tag{5-18}$$

相似度定义为

$$c(v,v') = \begin{cases} 1 & \text{如果 } a = a' \\ 0 & \text{否则} \end{cases} \tag{5-19}$$

那么相似度公式为

$$c_0(v,v') \in [b^{(a'-M)L}/b^{(a+1-M)L}, b^{(a'+1-M)L}/b^{(a-M)L}) = [b^{(a'-a-1)L}, b^{(a'-a+1)L}) \tag{5-20}$$

所以近似方法的错误为

$$\|c(v,v')-c_0(v,v')\| \le \begin{cases} 0 & \text{如果 } a=a' \\ b^{(M-|a-a'|)L} & \text{否则} \end{cases} \tag{5-21}$$

那么从特征空间的点分布知道，所选取的一对节点往往在一个格子中，所以近似错误是非常小的。

5.2　基于密集连接模式的可疑行为检测算法

本节介绍基于密度连接模式的可疑行为检测算法。内容包括引言、相关工作、密集行为所对应连接矩阵的特征子空间分析以及基于特征子空间的密集行为检测算法。

5.2.1　本节引言

给定社交网络、专利引用网络和电话网络等多种大规模应用的网络拓扑结构（图），如何能抓住可疑的用户行为？如何能找到惊人的、难以预知的连接模式？有很多工作已经研究了通信商的欺诈行为[274]、Ebay 中的虚假评价[175] 和 Facebook 上虚假的页面 “喜欢”[190]，而这里所研究的是常见的异常行为模式，并尝试开发一种通用的有效方法从不同的应用中检测出这类行为。

图5.3中展示出密集行为的三个案例：(a) 在 Facebook 或是 Twitter 的类似的可以被表示为无向图/有向图的社交网络中，许多售粉公司都设置了百万级的僵尸粉一起行动，共同关注同一群人（顾客）来提升他们的市场价值。所以，虽然这些被关注的人并不知名，但是他们会最终有很多粉丝。这些粉丝是花钱雇佣来的，或者是用脚本创造出来的。这种密集行为会在图对应的邻接矩阵中形成大的、密集的块。(b) 在论文引用的网络中，在同一个研究问题或者是同一个项目里的研究者们往往会互相引用对方的文章，即使这些文章与他们的工作毫不相关。(c) 在诸如 IMDb，MovieLens 和 Netflix 等电影参演的网络中，男演员/女演员/导演经常与关系已经很好的朋友一起合作参演电影，这样会更容易在工作中交流，更容易理解电影中演员的形象。这些网络是可以用二部图来描述的。要注意的是密集行为是说一组演员/导演与一组电影之间的交互，并最终在理解矩阵里面形成个密集子矩阵（块状）。在基于图的应用中，密集行为模式是非常常见的，所以一个很重要又很有趣的问题是：如何来找到几乎满员的密集块状子矩阵，也就是如何找到密集行为的链接？

这个问题做起来并不那么容易。就社交网络为例，其中有很多帮助顾客提升粉丝数的僵尸粉公司存在。这类行为扭曲社交媒体的网络结果，导致正常用户体

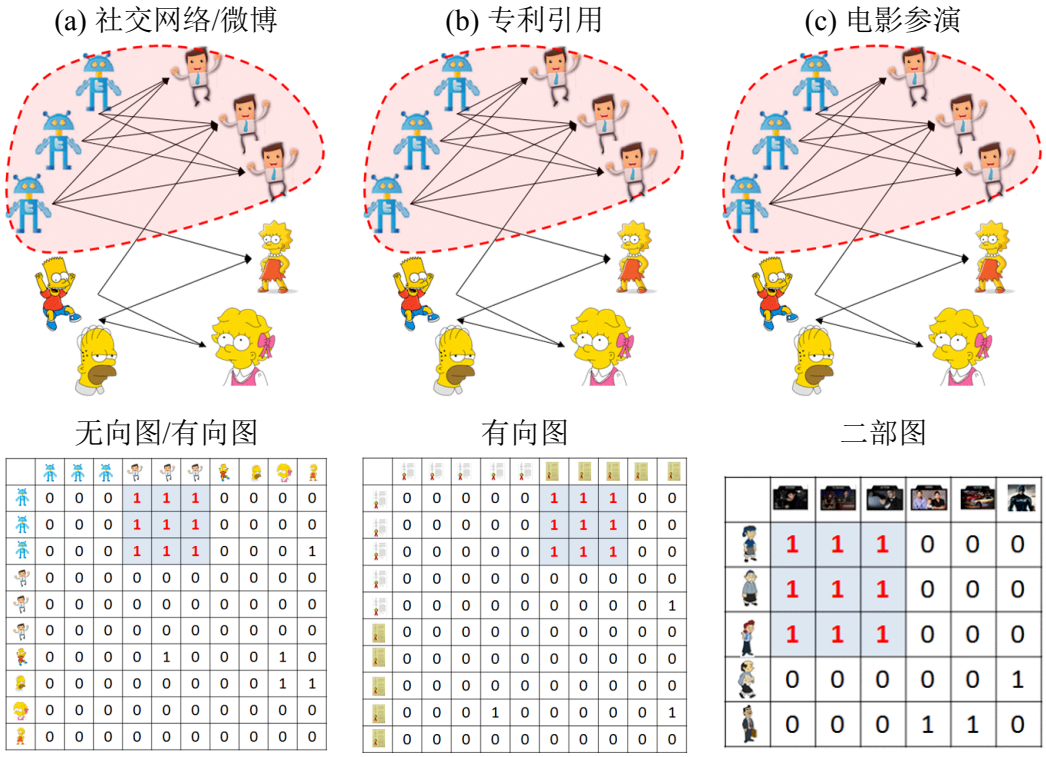


图 5.3　密集的可疑行为：在许多基于图的应用中，比如社交网路、专利引用和 IMDb 中，密集行为模式是常见的。这种特别的行为在图（以及邻接矩阵）中形成过于大、过于稠密的块状结构。

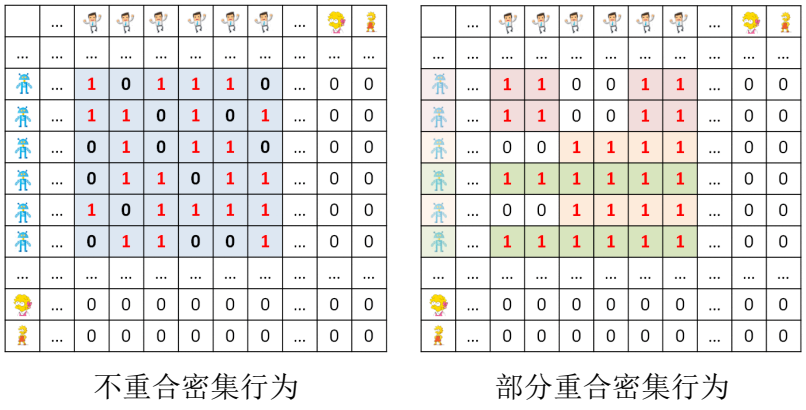
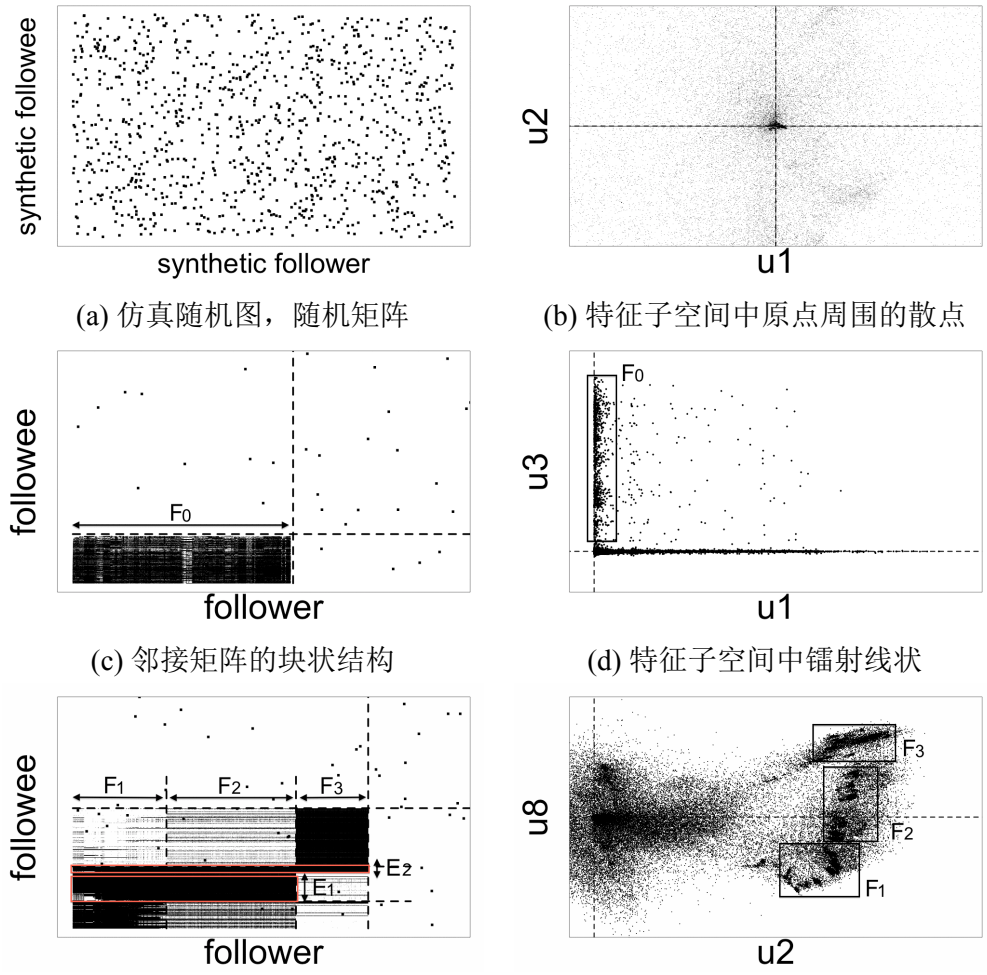


图 5.4　如何检测不重合密集和部分重合密集行为？(a) 如何从邻接矩阵中检测到密集却不是 100% 密集的块状结构？(b) 事实上会存在三组不同的僵尸粉关注他们的顾客！

验受到严重伤害。僵尸粉公司会开发出各种办法来绕开检测。一种是形成密度较低的块。比如在图5.4(a) 中就提出了一个更难的问题：如何检测到密集却不是 100% 密集的块状结构？什么情况下一个块过于大过于密集，以至于在图中很难出现？图5.4(b) 中给出了多组僵尸粉与顾客相连接的案例。僵尸粉群往往很分享顾客，而他们形成的密集行为会产生部分重合。如何检测部分重合的密集行为？

近年来的一些研究把社交图数据转为连接模式来研究社区结构[227,236,238] 以及



(a) 仿真随机图，随机矩阵　(b) 特征子空间中原点周围的散点

(c) 邻接矩阵的块状结构　(d) 特征子空间中镭射线状

(e) 邻接矩阵的阶梯状结构　(f) 特征子空间中珍珠状

图 5.5　密集行为会在邻接矩阵中形成特定的连接模式和奇特的特征子空间的形状：在仿真图中，在特征子空间中粉丝都在原点周围分散。在微博数据中，粉丝组 F_0 的不重合密集行为会在邻接矩阵中形成密集块，在特征子空间中形成镭射线。粉丝组 F_1-F_3 的重合密集行为会形成阶梯状结构和珍珠状的子空间分布。

聚类属性[211,251]。然而，并没有任何分析能够指出如何检测出特殊行为的方式方法。本文在腾讯微博的完整有向图数据上做研究。这份数据是 2011 年 1 月爬取得到，含有 1.17 亿的用户和 33.3 亿的社交关系。在微博图中研究用户的关注行为时讨论了不同种类的密集行为。比如图5.5(a-b) 中的无密集行为，图5.5(c-d) 中的不重合密集行为，图5.5(e-f) 中的部分重合密集行为。在邻接矩阵中寻找连接模式并检查特征子空间中对应的形态。

图5.5(a)、5.5(c) 和 5.5(e) 中展现了链接关系，也就是用黑点描述邻接矩阵中的非零值，所在 X 轴是粉丝编号，所在 Y 轴是被关注人的编号。密集行为形成的密集黑块用虚线高亮出来。图5.5(b)，5.5(d)，5.5(f) 中画出了粉丝节点的一对矩阵的左奇异向量值。这些图能够可视化特征子空间，虚线分别是 X 轴和 Y 轴。借助

名词	描述
阶梯状	多组密集行为交叠发生，形成重叠的块
镭射线状	沿数轴的一簇线状点
珍珠状	在同等半径下的珍珠项链（开环）状点簇

表 5.5　本文用到的形容形状的名词和描述

行为模式	邻接矩阵连接模式	特征子空间
不含密集行为	散点	环绕在原点周围
不重合密集行为	“块状”	“镭射线状”
部分重合密集行为	“阶梯状”	“珍珠状”

表 5.6　如何用特征子空间图的形状来推测奇特的行为模式？密集行为模式表示图中奇特的连接模式，而特征子空间能以奇特的形状来反映这种奇特的连接模式。

表5.5中的名词表征复杂模式可以讨论如下：

- 不含密集行为：根据 Chung-Lu 模型[260] 仿真了不含密集行为的随机幂律图。图5.5(a) 中的邻接矩阵并不含有大的、密集的块。本工作研究了每一对二维的特征子空间，看到在图5.5(b) 中原点周围的粉丝。
- 不重合的密集行为：在腾讯微博中存在一组僵尸粉 F_0 关注同一组人。那么图5.5(c) 所示，邻接矩阵中就会有一个大的密集的块（83,208 个粉丝，密度为 81.3%）。图5.5(d) 画出了第 1 个和第 3 个左奇异向量形成的特征子空间。粉丝组 F_0 在 Y 轴一侧形成镭射形状的点簇。
- 部分重合的密集行为：在邻接矩阵中会看到更惊奇的连接模式，也就是如图5.5(e) 中的阶梯状（10,052 个粉丝，密度为 43.1%）。僵尸粉组 F_1-F_3 的密集行为分别形成三个密度超过 89% 的密集块。然而不同于不重合密集行为，F_1 和 F_2 有同样的关注人群 E_1，而 F_1 和 F_3 有同样的关注人群 E_2。重合的密集行为的邻接矩阵的第 2 个和第 8 个左奇异向量形成了特征子空间，如图5.5(f) 所示，含有多个微小的簇以同样的半径围绕着原点。如同不完整的球状，又像珍珠项链，称之为 “珍珠状” 模式。

本工作给出的方法如表5.6所示，讲述了如何用特征子空间来推测可疑的行为模式。解决思路是密集行为模式会在图中形成特别的连接模式，邻接矩阵的特征子空间又能用奇特的星状来反映这种奇特的连接模式。后续小节解释方法的细节。观察可知所提出的 LockInfer 方法能有效地学习关系图的连接模式并检测出密集行为。贡献点如下：

	(a.1) 检测 “块” 密集行为	(a.2) 检测 “阶梯” 密集行为	(b) 种子选取（解释 “镭射线” 和 “珍珠状”）	(c) 可扩展（$\leq O(E)$）
METIS	×	×	×	√
Crochet	×	×	×	√
AdjCluster	√	×	×	×
SpokEn	√	×	×	√
CopyCatch	√	×	×	√
LockInfer	√	√	√	√

表 5.7　传统算法并不具有这些特性：(a) 检测形成 (a.1) 块状的密集行为和 (a.2) 阶梯状的密集行为，(b) 解释 “镭射线” 和 “珍珠状” 现象，(c) 对于大规模图来说可扩展。表中展现出所提出的 LockInfer 方法的优势。

- 解决思路： 根据不同类型的仿真密集行为在奇异向量中留下的痕迹，总结出一系列的诊断方法。这些方法能够让数据科学家和实践者能够从奇特的连接行为中发现可疑的用户行为。
- 检测算法： 提出了快速可扩展算法，利用上述解题思路自动地找到密集行为。算法的运行时间与图中的边数呈线性关系。在仿真实验和真实数据（包括腾讯微博和 IMDb 数据）上分别证实了算法的有效性。

5.2.2　相关工作

本小节分四组介绍相关的研究工作：子图挖掘，图分割，谱聚类和社区检测。表5.7给出本工作的 LockInfer 算法与采用特征向量和特征子空间的传统图挖掘方法相比，既有效、又有可解释性，还兼具可扩展性。

大量的子图挖掘算法已经用于超链接数据[284]，计算生物学和计算机网络[206]。有一系列的图挖掘算法能找到类闭环（quasi-clique）模式，比如 Crochet[218,219]，找到频繁子图[217]，周期子图[221] 和密集子图[204,226]。不重合的密集行为形成密集的二部子图，而不是频繁子图和闭环；由部分重合的密集行为所形成的子图并不足够密集，所以寻找密集子图的算法找不到这种行为。

典型的图分割问题是把图分为若干个组成部分，这些组成部分是有同样的大小，而每一对组成部分之间的链接少到几乎不存在，如表中给出的 METIS 等算法[198,200,202,205]。然而当密集行为存在重合部分，图中就并不存在好的分割办法[236]。CopyCatch[190] 是采用信任传递算法的基于二部子图[213] 的检测工作。但是相关方法没有介绍如何选出合适的种子，而本工作给出选取种子的方法。

谱聚类算法已经被广泛用于大规模图中[250]。谱聚类算法使用表征数据的矩

阵的特征向量进行数据聚类[241]。二部图子空间分割算法[99] 则是使用归一化的拉普拉斯矩阵的第二个特征向量。于是谱特征子空间得到了广泛关注。SpokEn[239] 展现出手机的通信网络拓扑图形成的奇异向量如果被成对绘制，会在特定的轴旁形成独立的线。这代表了异常紧密的社区。然而真实存在的密集社区或是密集行为并不一定沿着轴[264]。AdjCluster[240] 给出了在特征子空间中正交之线存在的理论研究。已有工作不能够解释弯转的镭射状线，更没有讨论过珍珠形状。

研究人员已经开发出很多社区检测算法[231,237]。其中一种思路是让一类用户集合的节点，内部连接程度要比外部链接程度要大[236]。研究者们还给出如何从网络拓扑结构区优化模式度来推测社区结构[230,235]。通常期望社区里的用户有密集的内部链接，而不同社区之间的用户有很少的链接。然而密集行为模式并不会符合这样的假设，因为可疑用户完全可以与不同组成部分中的目标用户相连。

总结来说，上面所讨论的方法中没有一个能够指导研究者如何理解真实网络拓扑，解释奇特的特征子空间形状，并检测出不重合和重合的密集行为。而本文所提出的 LockInfer 针对这些问题研究。

5.2.3　密集行为的特征子空间分析

在这一小节，首先介绍 “密集块” 的定义和理论上的密度阈值，然后介绍如何绘制特征子空间。通过讨论不同类密集行为，给出行为形成的密集块性质，并给出一系列从特征子空间的模式和连接模式来判断密集行为类型的规则。

用 (S, T) 表示源节点集合 S 与目标节点 T 形成的子图。通过节点重排序之后，邻接矩阵中就会出现 “块” 的形状。用 $d(S, T)$ 表示块密度即非零值比例。那么密集块可疑定义为实际密度比均一假设下要高的块。正式的定义如下：

定义 5.2 (密集块)：　在邻接矩阵 A（大小为 $M \times N$，密度为 D），一个大小为 $m \times n$ 的块 (S, T) 可以被叫做 “密集块”，当且仅当密度 $d(S, T)$ 比均一密度 $\hat{d}$ 要高，即 $d(S, T) \geq \hat{d}$，其中 $\hat{d}$ 是在公式5.1定义的阈值密度。

直觉上理解这个定义是说，大又密集的块表示了密集行为，所以看上去非常可疑。密度阈值 $\hat{d}$ 可以被如下估计：

引理 5.1：　阈值密度 $\hat{d}$ 是指一个密集块几乎不可能出现，也就是在稀疏矩阵中出现的次数估计小于 1。那么阈值密度为

$$\hat{d} = \frac{1}{log(D)}\left(\frac{1}{n}log\frac{m}{M} + \frac{1}{m}log\frac{n}{N}\right). \tag{5-22}$$

证明　用 A 来表示大小为 $M\times N$，密度为 D 的基于 Erdös-Rényi 模型的随机图的邻接矩阵，其中 $A_{i,j}$ 是概率为 D 的独立伯努利随机变量。所以 A 中的大小为 $m\times n$，密度为 $d \geq \hat{d}$ 的密集块是

$$\begin{aligned} X(A,m,n,\hat{d}) \quad = \quad & \{(S,T) : \sum_{i\in S}\sum_{j\in T} A_{i,j} \geq mn\hat{d}, S = \{i_1,...i_m\}, T = \{j_1,...j_n\}, \\ & 1 \leq i_1 < \cdots < i_m \leq M, 1 \leq j_1 < \cdots < j_m \leq N\} \end{aligned} \tag{5-23}$$

定义 $Y=\sum_{i\in S}\sum_{j\in T} A_{i,j}$，期望是 $\mu=E[Y]=mnD$，那么密集块数量的期望值为

$$E[\mid X(A,m,n,\hat{d}) \mid] = \binom{M}{m}\binom{N}{n} Pr[Y \geq mn\hat{d}]. \tag{5-24}$$

根据 Chernoff 边界和 Stirling 近似公式，可知

$$Pr[Y \geq mn\hat{d}] \leq \left(\left(\frac{D}{\hat{d}}\right)^{\hat{d}}\left(\frac{1-D}{1-\hat{d}}\right)^{1-\hat{d}}\right)^{mn}, \tag{5-25}$$

$$\binom{M}{m}\binom{N}{n} \sim \frac{1}{2\pi\sqrt{mn}}\left(\frac{M}{m}\right)^m\left(\frac{N}{n}\right)^n. \tag{5-26}$$

因此，密集块数量的期望值的对数值为

$$\begin{aligned} & logE[\mid X(A,m,n,\hat{d}) \mid] \\ & \leq -mn\left(\hat{d}\cdot log\frac{\hat{d}}{D} + (1-\hat{d})\cdot log\frac{1-\hat{d}}{1-D}\right) - m\cdot log\frac{m}{M} - n\cdot log\frac{n}{N} - log(2\pi\sqrt{mn}) \\ & \leq -mn\hat{d}\cdot log\frac{\hat{d}}{D} - m\cdot log\frac{m}{M} - n\cdot log\frac{n}{N} \\ & = -mn\hat{d}\cdot(log\hat{d} - logD) - m\cdot log\frac{m}{M} - n\cdot log\frac{n}{N} \\ & \approx mn\hat{d}\cdot logD - m\cdot log\frac{m}{M} - n\cdot log\frac{n}{N} \end{aligned} \tag{5-27}$$

其中密集块的密度 $\hat{d}$ 通常比数据密度 D 要更高，也就是说 $\hat{d} \gg D$。所以，边界密度为

$$\hat{d} = \frac{1}{log(D)}\left(\frac{1}{n}log\frac{m}{M} + \frac{1}{m}log\frac{n}{N}\right). \tag{5-28}$$

图中每一个密度大于 $\hat{d}$ 密集块的数量的期望值小于 1，即很难存在。在图5.6中画出了在大小为 $1M\times 1M$、边数为 $3M$ 的随机图中，块的存在数量的期望值。一个大小为 100×100 的块在密度 $\hat{d} = 2\%$ 时是密集块。　□

“特征子空间图” 尝试通过可视化的方法展现出可疑的行为模式。用 A 来表示大小为 $N \times N$ 的社交网络邻接矩阵。每一个用户都可以被表示成一个 N 维的数据

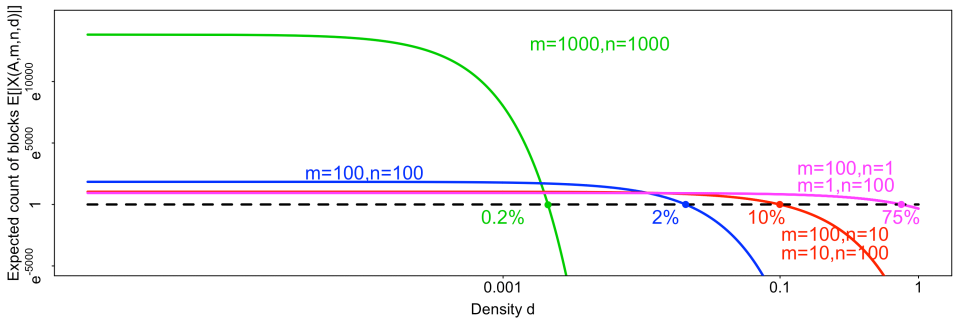


图 5.6　如果 $d \geq 2\%$，一个大小为 100×100 的块是密集块：这种块在大小为 $1M \times 1M$ 的随机图中的期望值会随着密度升高而降低。

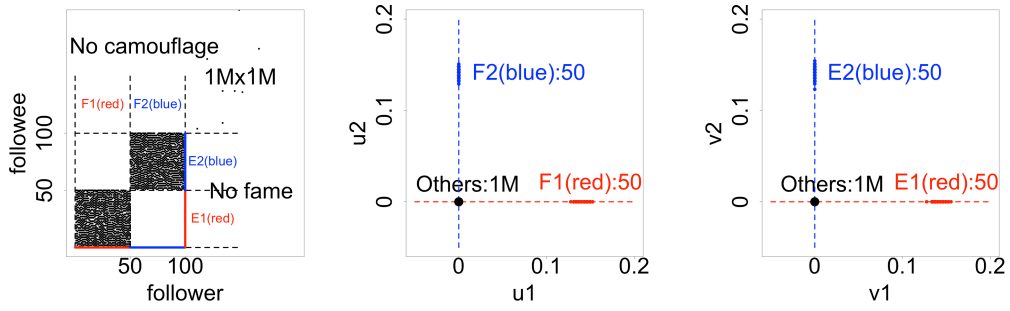
点，特征子空间图就是把这些在 N 维度的点在合适的二维子空间中展现出来。准确来说，子空间是由两个奇异值向量组成的。

k 维度的奇异值分解（SVD）是把形式为 $A = U\Sigma V^T$ 的矩阵因子化，其中 Σ 是由前 k 大的奇异值组成的、大小为 $k \times k$ 的对角矩阵，U 和 V 是大小为 $N \times k$ 的正交矩阵，其中分别包含左奇异向量和右奇异向量。$u_{n,i}$ 是矩阵 U 的第 (n,i) 个元素，相似的，$v_{n,i}$ 是矩阵 V 的元素。$u_{n,i}$ 是第 n 个粉丝在第 i 个左奇异向量中的值。定义 (i,j)-左特征子空间图为点集 $(u_{n,i}, u_{n,j})$ 形成的散点图，这就是 N 个粉丝的第 i 和第 j 个左奇异向量的映射。可以同样定义 N 个用户作为被关注人的情况，所以 (i,j)-右特征子空间图就是点集 $(v_{n,i}, v_{n,j})$ 的散点图。这个图能够可视化所有点，如果恰当使用，是可以解释很多邻接矩阵的内在性质的。

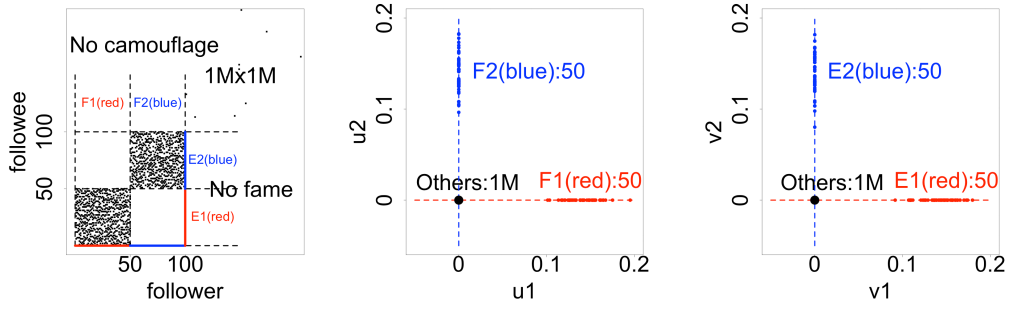
如同在图5.5(a-b) 中介绍的，给定一个随机幂律分布图，特征空间会是在原点周围的一些云一样的点集合。然而，在腾讯微博数据中看到了如镭射线状和珍珠状的特别形状。本工作中想要研究的就是：是什么样的用户行为导致了这些特别形状在特征子空间中出现？简短的答案是不同种类的密集行为。接下来会详细介绍密集行为类型和这些特征形状之间的关系。

在枚举所有密集行为的类型，首先要给出 “伪装” 和 “伪知名” 的概念。如果粉丝集合 F 存在着密集关注偶像集合 E 的利益驱动的动机，那么他们也可以进行 “伪装”，也就是关注额外的一些不在集合 E 中的用户；那些 E 中的用户也可能会 “伪知名”，也就是被一些额外的不在集合 F 中的粉丝关注。

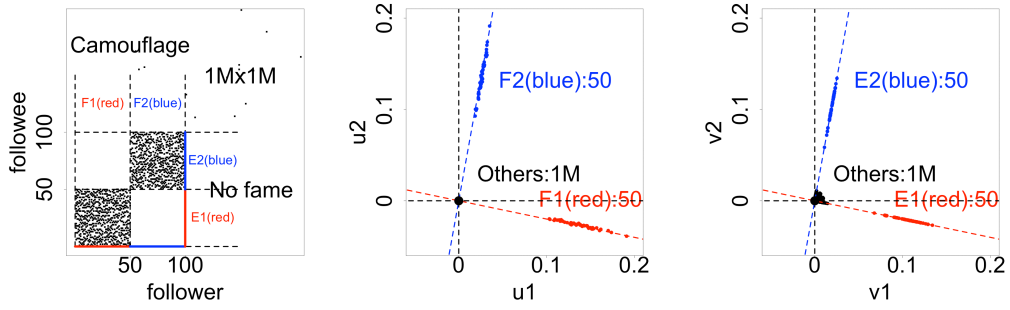
根据这些概念可以构造仿真数据研究密集行为。首先产生一个大小为 $1M \times 1M$ 的随机幂律分布图，然后注入两组不同的存在密集行为的粉丝集合。细节上说，注入集合 F_1（50 个新的粉丝）一起关注集合 E_1（50 个新的被关注的人）。相似地，注入另一组集合 F_2 一起关注集合 E_2。图5.7中左侧用黑点画出邻接矩阵中的非零元素，能看到两个大小为 50×50 的非重合密集块。不重合的密集行为的属性设置如下：



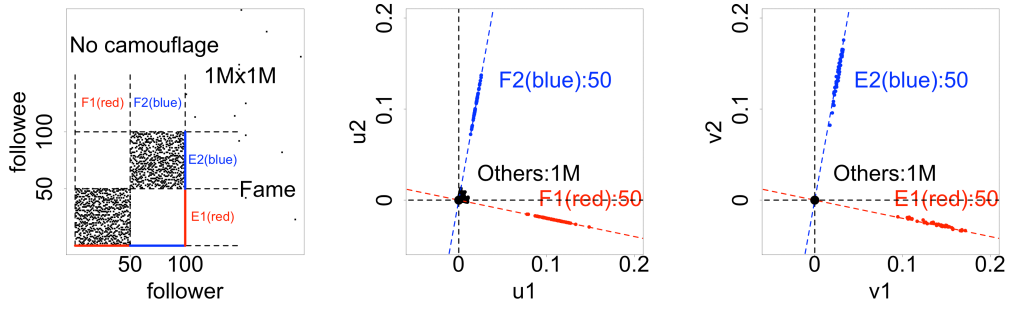
规则 1 (短 “镭射线”): 两个密集块，90% 的高密度，不含 “伪装”，不含 “伪知名”



规则 2 (长 “镭射线”): 两个密集块，50% 的低密度，不含 “伪装”，不含 “伪知名”



规则 3 (旋转的 “镭射线”): 两个密集块，含有 “伪装”, 不含 “伪知名”



规则 3 (旋转的 “镭射线”): 两个密集块，含有 “伪知名”, 不含 “伪装”

图 5.7　规则 1-3 (“镭射线”)：邻接矩阵中不重合的密集块。

- 密度：高密度是指新注入的粉丝关注 90% 的对应的注入的偶像；低密度是指比例为 50%。
- 伪装：伪装是让注入的粉丝关注 0.1% 额外的偶像；如果没有伪装，那么它只关注对应的偶像。
- 伪知名：伪知名是让注入的偶像还会被 0.1% 额外的粉丝关注；如果没有伪知名，那么它只被新注入的粉丝关注。

在图5.7的中间和右侧分别给出了左奇异向量和右奇异向量构成的特征子空间。于是能看到不同类型的非重合密集行为存在下述的可疑踪迹：

- 规则 1 (短 “镭射线”): 如果注入粉丝的密集行为非常紧密，邻接矩阵中会有一个或者多个密度高达 90% 的不重合块。特征子空间图会展现出短 “镭射线”：向原点延伸过去的贴近轴的线状密集点集。
- 规则 2 (长 “镭射线”): 如果注入的粉丝行为密集，但较为松散，邻接矩阵中有若干个密度为 50% 的不重合块。特征子空间图展现出长 “镭射线”：镭射线会贴近轴，并且向原点伸长。
- 规则 3 (旋转的 “镭射线”): 如果注入的粉丝有 “伪装” 或者是注入的偶像有 “伪知名” 的问题，邻接矩阵会在密集块以外形成稀疏的额外链接。和规则 1、2 不同的是，向原点延伸的镭射线会以某个角度旋转，叫做旋转的 “镭射线”。

另一方面，如果注入的粉丝密集地关注对应的偶像集合，这些偶像集合存在部分的重合，称之为部分重合的密集行为。仿真数据是在随机图中注入 3 个粉丝集合 F_i ($i = 1, \ldots, 3$) 和 5 个偶像集合 E_i ($i = 1, \ldots, 5$)。每一个粉丝集合都含有 1000 个粉丝，每一个偶像集合都含有 10 个偶像。F_1 的粉丝集合关注集合 E_1 到 E_3 的偶像；F_2 的粉丝集合关注集合 E_2 到 E_4 的偶像；F_3 的粉丝集合关注集合 E_3 到 E_5 的偶像。图5.8中可以看到邻接矩阵和特征子空间之间的关系。

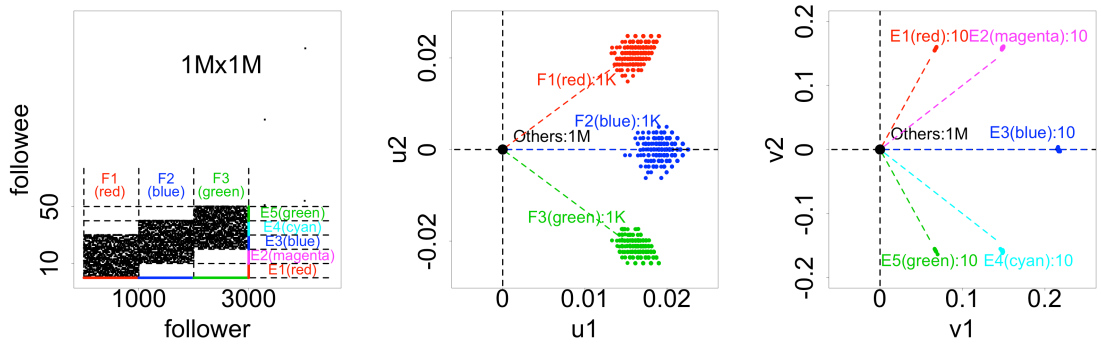
- 规则 4 (“珍珠状”): 重合的密集行为在邻接矩阵中形成 “阶梯状” 块，因为粉丝集合会连接若干个偶像集合，多个密集块之间是存在重合的。特征子空间中显示出离原点距离相近的球状，或者叫 “珍珠状” 的点簇。

图5.8(b) 给出 3 组 F_1 到 F_3 的粉丝集合在特征子空间中形成的 3 个点簇构成的珍珠状。图5.8(c) 给出 5 组 E_1 到 E_5 的偶像集合在特征子空间中形成的 5 个点簇构成的珍珠状。具有相近的或者是重合的偶像的注入粉丝会在特征子空间中靠近。

5.2.4　基于特征子空间的密集行为检测算法

本工作中所给出密集行为的检测算法 LockInfer 有下面两个步骤：

- 种子选取： 根据上一个小节给出的规则 1 到 4，选择具有密集行为的粉丝节点，并叫做 “有密集行为” 的粉丝。



(a) 邻接矩阵　　(b) 左特征子空间　　(c) 右特征子空间

图 5.8　规则 4 (“珍珠状”): 三个部分重合的密集块形成的 “阶梯状” 块。

- 密集值传递： 在粉丝集合和偶像集合之间传递 “有密集行为” 的值，简称密集值。每次用高于密度阈值定理给出的密度阈值 $\hat{d}$ 选取有密集行为的粉丝用户，去掉并不足够密集的用户，接下来是对偶像集合做同样的处理。

算法 7中还给出了每一步骤的细节。

Algorithm 7 LockInfer 算法：大规模图中检测密集块

Require: 邻接矩阵 A，密集块的最小规模 $m_{min}\times n_{min}$ 和密度阈值 $\hat{d}$

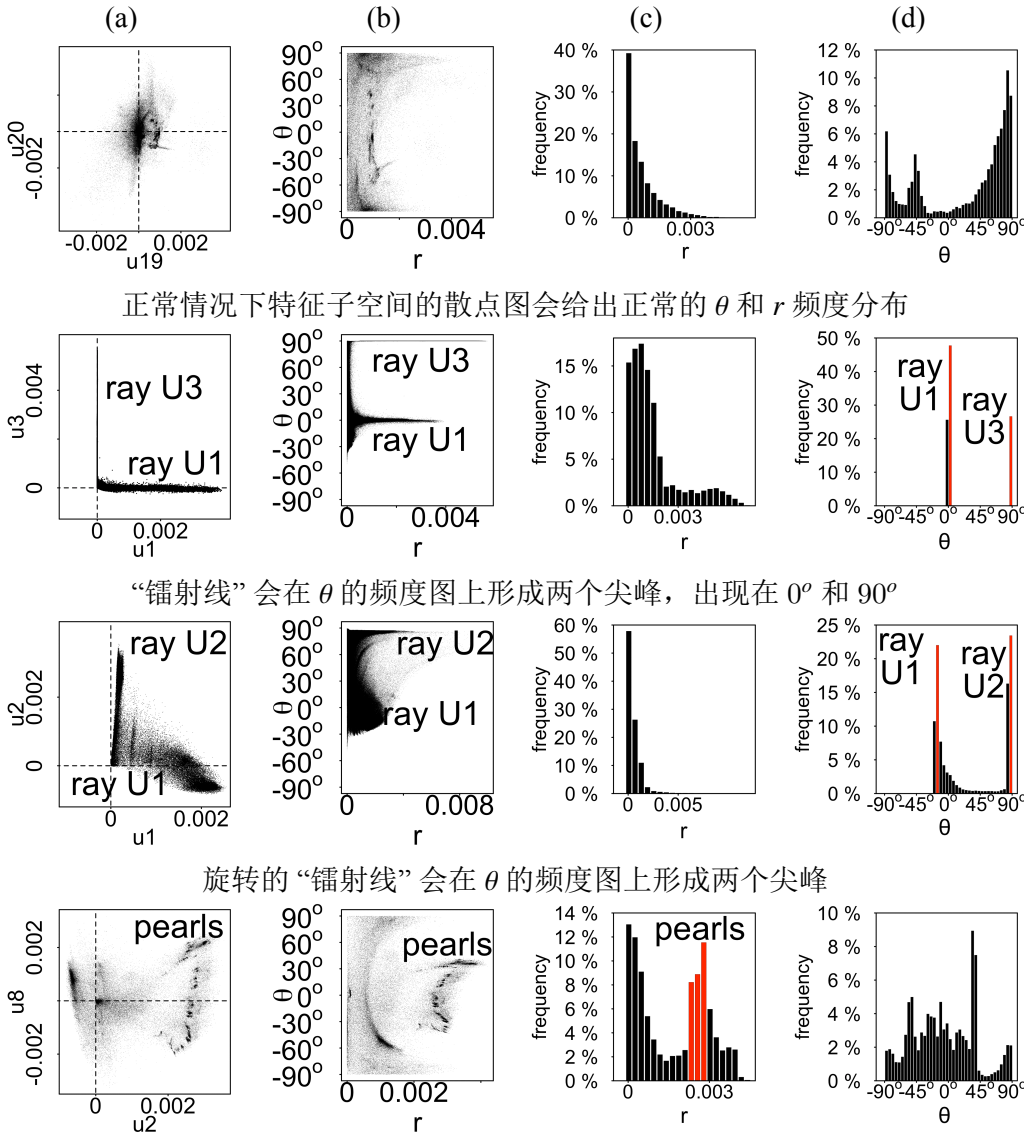
1: 如算法 8，$Seeds = SelectSeeds(A)$
2: $LockB = \{\}$
3: **for** 每一个 $Seeds$ 中的源节点集合 S_0 **do**
4: $\quad LockB = LockB \cup Scoop(S_0, m_{min}, n_{min}, \hat{d})$
5: **end for**
6: **输出:** 密集块集合 $LockB$。

LockInfer 可以从任意种子节点集合出发，甚至随机选取的节点。然而认真选取种子节点能加快检测速度。如下线索指出如何找到合适的种子：

- 选择出度在尖峰处的粉丝，但出度在尖峰处的节点大多数实际上正常。
- 用之前所给出的规则集合来选取粉丝，后续会证实这是非常有效的。

当然，如果采用额外信息，比如 IP 地址、个人信息等内容，可以用这些额外信息来选择可疑节点，例如，大量的可疑粉丝会设置自己的生日为同一年的第一天，都是男性，都来自同样的城市。然而，如果没有额外信息，LockInfer 还是能够从规则中有效选取种子。图5.9给出找到种子的步骤方法。种子选取的算法有三个步骤，如下：

首先生成一系列的特征子空间图，计算最大的 k 个左奇异向量 $u_1, \ldots, u_k$，并画出所有粉丝节点在每一对奇异向量张成的子空间中的分布，如图5.9(a) 所示。在



正常情况下特征子空间的散点图会给出正常的 θ 和 r 频度分布

“镭射线” 会在 θ 的频度图上形成两个尖峰，出现在 0^o 和 90^o

旋转的 “镭射线” 会在 θ 的频度图上形成两个尖峰

“珍珠状” 在 r 的频度图上形成一个离原点很远的尖峰

图 5.9　如何找到镭射线和珍珠状代表的源节点集合 (a) 特征子空间图; (b) 极坐标转换 (距离 r 和旋转角度 θ); (c) 距离 r 频度的柱状图; (d) 角度 θ 频度的柱状图.

Algorithm 8 $SelectSeeds(A)$

Require: 邻接矩阵 A，奇异值数量 k，距离频度和角度频度的柱数 K_r 和 K_θ

1: $Seeds = \{\}$
2: 计算 A 的 k 维奇异值向量 U 和 V
3: **for** 每一特征对 $(i,j), 1 \le i < j \le k$ **do**
4: 　第 1 步：画出 U_i 和 U_j 的特征子空间图
5: 　第 2 步：转换为极坐标系（距离 r 和角度 θ）：
6: 　对每一个用户 $u_x, x \le N, r_x = \sqrt{U_{i,x}^2 + U_{j,x}^2}, \theta_x = \arctan \frac{U_{j,x}}{U_{i,x}}$
7: 　第 3.1 步：画出距离分布的柱状图（r 和频度），用中位数过滤法检测尖峰，r 的频度为 $freq(r) = |\{x|r_x = r\}|$，$\theta$ 的频度为 $freq(\theta) = |\{x|\theta_x = \theta\}|$
8: 　第 3.2 步：画出角度分布的柱状图（θ 和频度）并检测尖峰，用中位数过滤法找到图5.9(d) 的红色柱子
9: 　第 3.3 步：将尖峰处的节点放入 $Seeds$ 集合中
10: **end for**
11: **输出:** 初始源节点（种子）集合 $Seeds$。

高维度的情况下，例如 U_{19} vs U_{20}，常见的特征子空间图中原点附近有一大簇云状点集。然而，从 U_1 vs U_3 中可以看到构成直角的镭射线，从 U_1 vs U_2 中可以看到旋转的镭射线，从 U_2 vs U_8 中可以看到珍珠状分布，这些都是非常奇怪的。

第二，用极坐标变换把所有的点画成 (r,θ)，其中 r 是点离原点的距离，θ 是旋转的角度。如图5.9(b) 所示，镭射线会形成在 $\theta = 0^o$ 和 $\theta = 90^o$ 处的两个团簇，珍珠状会形成在较大的 r 处的一些微小的点簇。

第三，可以把距离 (r) 和角度 (θ) 的轴分割为若干个部分，并且把 r 和 θ 的频度画在柱状图中。镭射线构成的角度分布图会在 0^o 和 90^o 形成尖峰，但是在其他部位并没有尖峰；珍珠状构成的距离分布图会形成单个尖峰，而其他图中频度会随着距离增大而慢慢减小。用中位数过滤法[278] 可以检测尖峰，并且把尖峰处的节点放入种子集合中。

要注意的是，如果不存在密集行为，邻接矩阵中没有密集块，特征子空间图会在原点周围形成云状节点。角度 θ 的频度会几乎是一个常数，而 r 的节点频度会随着 r 的增大而减小。

设定参数时，密集块的最小规模是 $m_{min} \times n_{min}$ $(m_{min} = 100, n_{min} = 10)$，最小的密度是 $\hat{d}$。同时可以设置 $k = 20$ 来权衡特征子空间图的数量和 SVD 的运算时间。通过阅读极坐标图，画出距离和角度的频度分布图，切割距离的轴为 $K_r = 20$ 个柱子，切割角度的轴为 $K_\theta = 2K_r = 40$ 个柱子，因为 θ 可能为正也可能为负。

下面是进行 “密集值” 的传递来找到所有有密集行为的粉丝和偶像节点。定义粉丝节点的密集值为其对应的有密集行为的偶像节点的百分比，定义偶像节点的密集值为对应的有密集行为的粉丝节点百分比，由此每一次用一个密度阈值来选择新的具有密集行为的粉丝节点和偶像节点集合。Scoop 函数如信任传递算法，迭代地从粉丝到偶像，从偶像到粉丝传递密集值。下面来解释其中每一个步骤。

Algorithm 9 $Scoop(S_0, m, n, d)$

Require: 种子集合 S_0，密集块必须大于 $m \times n$ 并且至少 d 稠密

1: $T_0 = \{\}; i = 0;$
2: **while** $S_i == S_{i-1}$ **do**
3: $\quad T_i = S2T(S_i, d);$
4: $\quad$ if $(\mid T_i \mid < n)$ return $(\{\},\{\})$
5: $\quad S_{i+1} = T2S(T_i, d)$
6: $\quad$ if $\mid S_{i+1} \mid < m$ return $(\{\},\{\})$
7: $\quad i = i + 1$
8: **end while**
9: return (S_i, T_{i-1})

Algorithm 10 $S2T(S, d)$

Require: 源节点集合 S 和密度 d

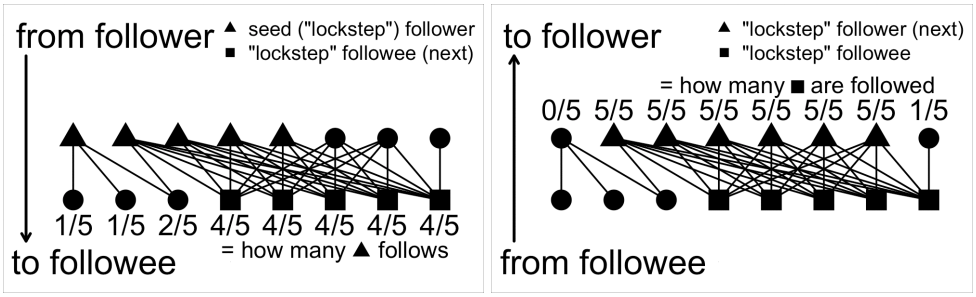
1: return $T = \{j : \sum_{i \in S} A_{i,j} > d \mid S \mid\}$

Algorithm 11 $T2S(T, d)$

Require: 目标节点集合 T 和密度 d

1: return $S = \{i : \sum_{j \in T} A_{i,j} > d \mid T \mid\}$

- 从源节点（粉丝）到目标节点（偶像）： 图5.10(a) 的有向图中上面是粉丝集合，下面是偶像集合。如果从 5 个密集的粉丝出发，对每一个偶像，计算他们粉丝在种子集合中的比例。选取有比例高的偶像作为有密集行为的偶像。函数 $S2T$ 给出如何从源节点传递密集值到目标节点。
- 从目标节点（偶像）到源节点（粉丝）： 接着是对每一个粉丝，计算他有多少比例的偶像是有密集行为的。图5.10(b) 给出了如何选取新的密集行为粉丝，并去除无辜的不关注或者只关注 1 个的偶像。函数 $T2S$ 给出了如何从目标节点传递密集值到源节点。



(a) $S2T$: 用粉丝选取有密集行为的偶像 (b) $T2S$: 用偶像选取有密集行为的粉丝

图 5.10　迭代地在源节点（粉丝）和目标节点（偶像）之间传递密集值：选择那些有太多密集行为的偶像（粉丝）的粉丝集合（偶像集合）。

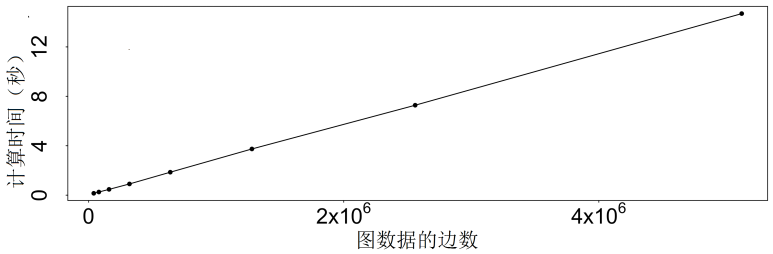


图 5.11　LockInfer 是有可扩展性的，复杂度与边数呈线性关系。

- 重复到收敛： 当收敛的时候如果密集块并不为空，报告有密集行为的粉丝和偶像集合。

LockInfer 的时间复杂度为 $O(E)$，算法是随着边数呈线性关系。在仿真的呈对角块状的邻接矩阵中运用该算法，每一个块都有 10,000 个节点和 40,000 个随机边，重复块结构得到含有 1,300,000 个节点和 5,100,000 个边的仿真图。图5.11展示了随着仿真图数据的大小（边数）变化的运行时间。注意到的是时间曲线与社交图的规模呈线性关系，LockInfer 算法是可扩展的，可被用于真实应用中。

5.3　跨维度行为可疑程度的通用评价指标

本节介绍跨维度行为可疑程度的通用评价指标。内容包括引言、相关工作、评价行为可疑程度的指标须满足的公理、行为可疑程度概率测度的通用评价指标以及基于评价指标的局域搜索算法。

5.3.1　本节引言

假设你在 Twitter 的工作是检测操纵热门话题、流行微博的欺诈者，因为时间很有限，下面哪一样更值得你去深入分析：一个是 2,000 个 Twitter 用户，一起转发同样的 20 个微博，每人转发 4 到 6 次；另一个是 225 个 Twitter 用户，一起转

发同一个微博，每人转发 10 到 15 次？现在如果知道前者发生在 10 小时内，而后者发生在 3 个小时内，哪一个更可疑？如果又知道后者的 225 个用户是只在 2 个 IP 地址上操作的呢？

图5.12给出了中国最大的微博平台之一腾讯微博的行为模式。本工作中所提出的方法 CrossSpot 检测到了一个 225 个用户在 2 个 IP 地址（蓝色的圆圈和红色的叉）、200 个分钟里转发微博 27,000 余次。进一步的分析会发现这些用户往往每隔 5 分钟发生一次行为。这类同步行为可能由自动脚本引起，会在图5.12中产生密集的块状。这些块是展开为若干个维度（用户 ID、时间戳、话题 hashtag 等）。虽然这里的任务是在类 Twitter 的网络中检测欺诈行为，所提出的方法也可以用在一些其他的设定，比如分布式服务攻击（DDoS，全称 Distributed Denial of Service），欺诈性的交互链接，欺诈性的用户点击，甚至是保险的欺诈行为等。本节工作的核心问题是如何比较两个二维或是高维密集块（张量数据）的可疑程度，定义为

问题 5.1 (可疑程度衡量)： 给定一个 K 维的整数值张量 $\mathcal{D}$，两个子张量数据 $\mathcal{Y}_1$ 和 $\mathcal{Y}_2$，哪一个子张量数据更为可疑，值得进一步分析？

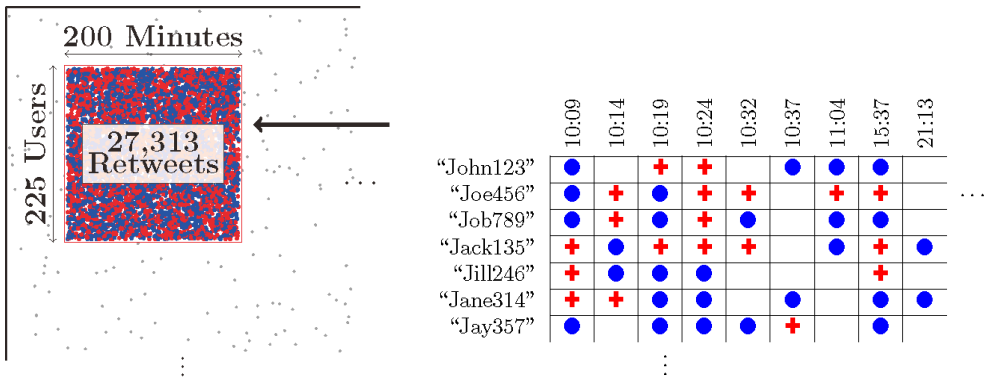


图 5.12　多维度数据中的密集块是可疑的。左边是腾讯微博上 225 个用户在 2 个 IP 地址、超过 200 分钟的时间里转发同一个微博 27,313 次；右边是这个密集块的放大后能看到的细节。● 和✚ 表示了这 2 个 IP 地址。由此可以看到用户行为是如何在多个维度（IP 地址、用户 ID 和时间戳）上同步。

为什么需要用高维度的张量数据：图结构数据和社交网络已经吸引了相当多的研究者，他们把数据集建模为 K=2 个维度，也就是矩阵形式：在 K=2 维的情况下，研究者们已经可以建模 Twitter 的 "谁关注谁" 的网络[195]，Facebook 的 "谁和谁是好友" 网络和 "谁喜欢什么" 的网络[190]，eBay 的 "谁购买了谁的东西" 的图[176]，"谁交易了哪个股票" 的金融行为数据以及 "谁引用了谁" 的论文关系。但是更有影响力的数据集往往会维度更高：如果考虑到时间因素或者是在 eBay 或是 Amazon 上的产品评价词汇时，会需要用到 K=3 维的数据表示。如果在研究网络

攻击的时候，会需要考虑源 IP、目标 IP、目标端口和时间戳的 K=4 维数据[107,194]。另一个例子是欺诈健康保险数据（含有病人 ID、医生 ID、处方 ID 和时间戳），其中会存在医生开出假冒的处方、很贵的药物由此来欺负他们的病人[186,273]。

为什么密集块更值得怀疑、更需要分析？在上述例子中，密集块会更令人惊奇[①]。过去的工作已经发现了在张量数据中的密集块往往对应着可疑的、同步的行为，比如 Facebook 上可购买的页面喜欢也就是一部分人会同时喜欢同样的一些页面[190]。垃圾传播者会重复去给饭馆或者旅馆写同样的、或高或低的、对自己有利的评价，或是用同样的用户、甚至是同样的文字[187,189]。僵尸粉会去关注他们的顾客，使得粉丝数量能够很大，看上去非常知名。这种高密度的产生是归于同一个原因：垃圾传播者会受限于资源（用户、IP 地址、时间戳等），但还是想尽可能在图或者张量数据中加更多的边来最大化金钱利益。直觉上讲，数据中的行为越同步，维度越高，这个数据就越值得进一步调查。

	方法和指标	给密集块评分	公理				
			密度公理	尺寸公理	浓度公理	对比公理	高维度公理
			1	2	3	4	5
评价指标	**Suspiciousness**	✔	✔	✔	✔	✔	✔
	数量	✓	✓	✗	✗	✗	✓
	密度	✓	✓	✗	✓	✗	✗
	平均度数[216]	✓	✓	✗	✗	✗	N/A
	奇异值[227]	✓	✓	✓	✓	✗	✗
检测方法	**CrossSpot**	✔	✔	✔	✔	✔	✔
	Subgraph[218,227,228]	✓	✓	✓	✓	✗	N/A
	CopyCatch[190]	✓	✓	✓	✓	✗	N/A
	EigenSpokes[239]	✗	N/A				
	TrustRank[172,185]	✗	N/A				
	BP[176,189]	✗	N/A				

表 5.8　与先进的评价指标和基线方法作比较，这些方法对于特定应用都是成功的，但是并不能够满足通用目标。

本工作的新视角：已经有很多工作来寻找密集的子图[218,227–229]，块状

① 极端稀疏的块也是非常可疑的，但是欺诈者往往不会让自己不做操作，所以稀疏块并不是工作的研究重点。

和社区[199,220,222]，包括矩阵代数方法（SVD[93,280]，张量分解[99,103] 和 PageRank/TrustRank[212,282]）；另有一些工作在异常检测和欺诈检测中用这些方法[107,172,239]。这些方法确实能有效地找到可疑的行为，近乎总是和密集的子图相关联。然而，其中并没有任何一个方法能够回答前面提出的问题（问题 5.1）。下面的特质能够把本工作和过去工作分割开来（如表5.8所示）：

- **给块状数据评分：** 本文关注于找到和衡量块状数据的重要评价指标。或是不评分的方法（如 SVD 和 PARAFAC/Tucker 的张量分解等）或是只给节点评分（如 PageRank、TrustRank 和 Belief Propagation 等），但并不是整个组。只有对组评分、对组检测，才能有效抓住可疑行为模式。
- **跨维度：** 检测所有 K 维度、或是维度的任意子集的可疑高密度块。对比起来，SVD 和密集子图挖掘的方法只能在 K=2 维的情况下使用，而 PARAFAC 和相关的张量分解方法只能返回所有维度下的高维度块。

本工作的贡献和创新点如下：

- **新的指标要求：** 提出了一系列满足从稀疏的高维数据中检测密集的块状子数据的基本公理（比如如果两个块是同样的大小，那么更密的就更令人惊讶）。虽然这些公理很简单，但是满足所有公理并不是简单容易的。
- **新的评价指标：** 介绍了一种新颖的名为 Suspiciousness 的评价标准来评价子向量、子矩阵和子张量数据在高维度数据中是否可疑。这里给出的评价标准是从基本的概率论得出，并且满足所有特定的要求。
- **新的 CrossSpot 算法：** 设计了可扩展搜索算法找到张量数据中的可疑块。
- **有效性证明：** 充分实验来证明在检测操纵热门话题、转发微博推广的可疑行为的任务中，直接优化评价指标的 CrossSpot 算法比起如 SVD 等分解方法的效果要好很多。

5.3.2　相关工作

本节调研了相关工作，包括可疑行为检测、分解方法和密集子图挖掘算法。在表5.8中比较了本工作和基线方法，并指出了本工作的独特性。

可疑行为检测：一些研究方法已经从多维度的关系数据中找到可疑的行为。这种欺诈模式在 eBay 评论[176]、在垃圾观点的传播[178,179,191]、在虚假用户[185,195]中都有被检测出来。许多方法关注于标记个人，比如用信任传播[176,189]（Belief Propagation，简称 BP）或是类 PageRank 的 TrustRank 评分方法[172,185]。这些方法仿佛可以把密集的、可疑的组群标记出来，但并不可以返回真实存在的密集组群。接着，还有一些方法发现增加更多维度的信息能够有助于检测可疑行为。CopyCatch[190] 发现 Facebook 上喜欢页面这一行为在时间上的可疑模式是欺诈行

符号	定义
K	数据集中的维度数量
$\mathcal{D}$	K 维度的张量数据集
$\mathcal{Y}$	$\mathcal{D}$ 中的子张量
$\mathbf{N}$	$\mathcal{D}$ 中每一个维度的尺寸的 K 长度的向量
C	$\mathcal{D}$ 中的总计数 ($\mathcal{D}$ 中所有元素的总和)
$\mathbf{n}$	$\mathcal{Y}$ 中每一个维度的尺寸的 K 长度的向量
c	$\mathcal{Y}$ 中的总计数 ($\mathcal{Y}$ 中所有元素的总和)
p	$\mathcal{D}$ 的密度，即 $C/\prod_k N_k$
ρ	$\mathcal{Y}$ 的密度，即 $c/\prod_k n_k$
f	Suspiciousness 函数：用计数做参数的可疑程度指标
$\hat{f}$	Suspiciousness 函数：用密度做参数的可疑程度指标
$D_{KL}(\rho\|\|p)$	泊松分布 (p) 和泊松分布 (ρ) 的直接 KL 区分度，即 $p-\rho+\rho\log\frac{\rho}{p}$

表 5.9　本工作中的符号和定义。

为的征兆。Jindal 等分析了 Amazon 的产品评价、评价用户、评分、日期、评论标题、内容和反馈来检测垃圾的、可疑的评价行为[178]。上述许多方法能够给单个用户、单个 IP 地址标记为是否可疑，但是并不能够对密集块给一个分数，这个任务甚至对于人工标注来说都是很困难的，但这个任务非常重要。（试问如何能够根据一个 IP 上的一个个人行为是可疑的?）最后，因为这些方法都有不同的形式化问题，所以并不能够在这个领域相互比较方法的有效性。然而，这些方法中没有一个能够给出 k 维度下的密集子数据的可疑程度分数。相应地，本文中尝试从本质上研究并量化可疑行为模式。

基于 SVD 的方法：分解方法已经被广泛用在子空间聚类[99]、社区挖掘[227,239]、和模式发现[100,101,103,104] 中。SVD 关注于矩阵中的密集块：Prakash 等提出了 EigenSpoke 来理解每一对奇异值向量的散点图并发现模式和社区[239]。Chen 等用谱聚类的架构来抽取密集子图[227]。对于高维数据来说，张量分解方法也已经用于很多实际系统中[101,103]。高维奇异值能够反映聚类的重要程度[99]。后续小节中会给出 SVD 在评价跨维度密集块的局限性。

密集子图挖掘方法：很多有意义的工作是在寻找有很高的平均度数的密集子图[215,220,222,229]。Charikar 给出简单的贪心近似算法来找到连接程度很高的、有很高平均度数的子图[216]。近似闭环和 K-密集子图的形式是以密度为基础的测量密集部分的检测办法[218,227,228]。然而，如后续要证明的，平均读书或是密度并不能够在衡量多维度数据集时得到实用效果。

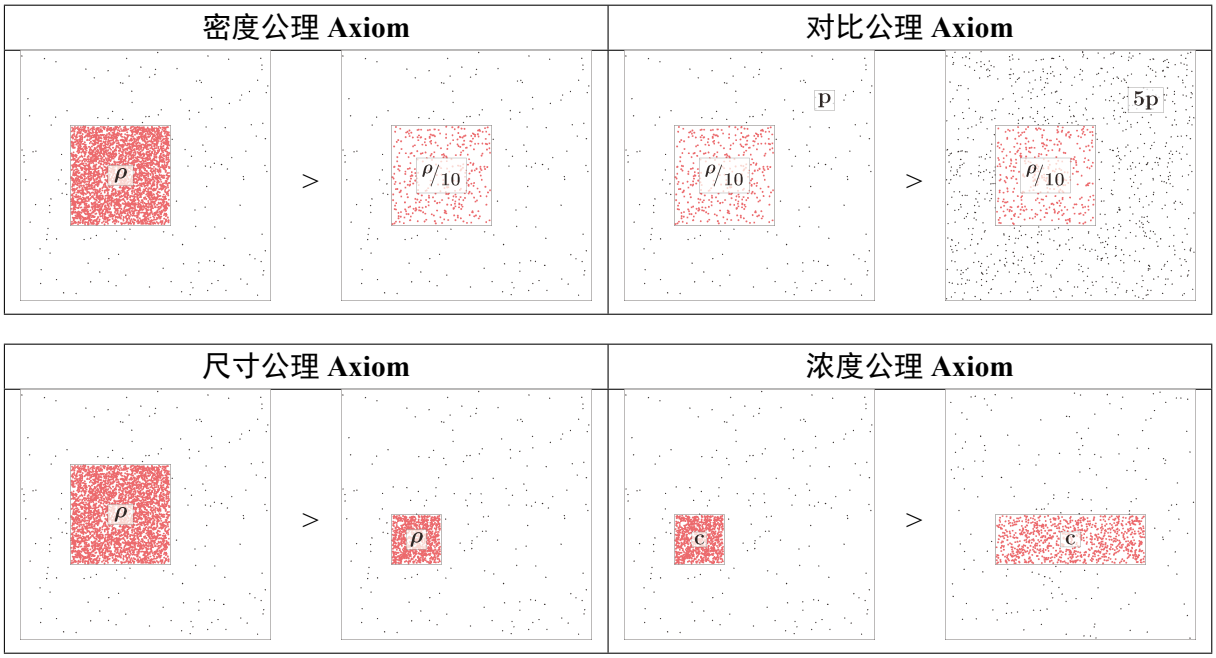


表 5.10　若干个合理的评价指标必须满足的公理。左侧的密集块比起右侧的都更为可疑。($\rho = 0.1, c = 1000, p = 0.0008$)

5.3.3　评价行为可疑程度的指标须满足的公理

给出所研究问题直觉含义、并和与过去工作比较后，给出问题正式定义：

问题 5.2 (可疑程度衡量)：**给定：** 一个 K 维度的大小为 $\mathbf{N} = [N_k]_{k=1}^K$ 的张量数据 $\mathcal{D}$，其中包含 C 个事件（也就是张量中的元素总和），**定义：**一个函数 $f(\mathbf{n}, c, \mathbf{N}, C)$ 能够衡量大小为 $\mathbf{n} = [n_k]_{k=1}^K$ 的子张量数据 $\mathcal{Y}$ 是否可疑。

另一种参数化方法是采用密度，而非计数。用 ρ 表示子张量 $\mathcal{Y}$ 的密度，用 p 表示数据集 $\mathcal{D}$ 的密度，那么

$$\hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) = f\left(\mathbf{n}, \rho \prod_{k=1}^{K} n_k, \mathbf{N}, p \prod_{k=1}^{K} N_k\right) \tag{5-29}$$

由于通常情况下维度数量都是很明确的 K，所以上述函数实际上是指代函数 f_K 和 $\hat{f}_K$。表5.9中给出了所有符号的定义。这里限制函数 f 只关注于密度符合 $\rho > p$ 的密集块，也就是说块中的密度比起数据（整个张量）的密度要大很多。虽然极端稀疏的部分也非常特别，但这不是本文所关注的重点。

下面列出了五个合理的可疑程度衡量指标 f 函数必须满足的基本定理。表5.10中给出可疑程度指标符合公理的可视化表示。

公理 1. 密度公理： 如果两个密集块尺寸一样、维度一样，计数大的密集块要比计数小的要更可疑。正式写下来是

$$c_1 > c_2 \iff f(\mathbf{n}, c_1, \mathbf{N}, C) > f(\mathbf{n}, c_2, \mathbf{N}, C) \tag{5-30}$$

公理 2. 尺寸公理： 如果两个密集块密度一样、维度一样，尺寸大的密集块比起尺寸小的要更可疑，也就是

$$n_j > n'_j \wedge n_k \geq n'_k \ \forall k \implies \hat{f}(\mathbf{n}, \rho, \mathbf{N}, p) > \hat{f}(\mathbf{n}', \rho, \mathbf{N}, p) \tag{5-31}$$

公理 3. 浓度公理： 如果两个密集块计数一样多、维度一样，尺寸小的密集块比起尺寸大的要更可疑，即

$$n_j < n'_j \wedge n_k \leq n'_k \ \forall k \implies f(\mathbf{n}, c, \mathbf{N}, C) > f(\mathbf{n}', c, \mathbf{N}, C) \tag{5-32}$$

公理 4. 对比公理： 如果张量数据中的两个密集块完全一样，但是其中一个张量数据更稀疏，那么稀疏的数据中的密集块更为可疑，即

$$p_1 < p_2 \iff \hat{f}(\mathbf{n}, \rho, \mathbf{N}, p_1) > \hat{f}(\mathbf{n}, \rho, \mathbf{N}, p_2) \tag{5-33}$$

公理 5. 高维度公理： 如果一个密集块中包含某个维度所有的值，那么可疑程度和把这个维度忽略没有差别，即

$$f_{K-1}\left([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C\right) = f_K\left(([n_k]_{k=1}^{K-1}, N_K), c, [N_k]_{k=1}^{K}, C\right) \tag{5-34}$$

引理 5.2： **跨维度的可疑程度** 每增多一个维度，会让密集块变得更为可疑，即

$$f_{K-1}\left([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C\right) \leq f_K\left([n_k]_{k=1}^{K}, c, [N_k]_{k=1}^{K}, C\right) \tag{5-35}$$

证明：

$$\begin{aligned} f_{K-1}\left([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C\right) &= f_K\left(([n_k]_{k=1}^{K-1}, N_K), c, [N_k]_{k=1}^{K}, C\right) \\ &\leq f_K\left(([n_k]_{k=1}^{K-1}, n_K), c, [N_k]_{k=1}^{K}, C\right) \end{aligned} \tag{5-36}$$

上述证明中第一个等号由公理 5 得来，第二个等号由公理 3 得来。 ■

上述公理简单且符合直觉，但并不是很容易就能做到。表5.8中给出了过去的评价指标会不能满足一部分的公理。

总计数： 一个可能的指标是密集块的总计数，即 $f(\mathbf{n}, c, \mathbf{N}, C) = c$。同样数量的事件在更小的空间（块）中发生，并不会改变可疑程度函数值，所以不符合公理 3；同样这也没有考虑到背景数据的密度 p，也就不符合公理 4。

密度： 另一个常用的指标是密度，即 $\hat{f}(\mathbf{n},\rho,\mathbf{N},p)=\rho$。然而，这并没有考虑密集块的大小，所以会不满足公理 2。这个指标也并没有考虑背景数据密度，所以不能满足公理 4。因为通常情况下维度越多，密度也就自然越小，所以会不满足公理 5。

平均度数： 很多寻找密集子图的研究方法都着重于密集子图的平均度数[215,220]，也就是 $f(\mathbf{n},c,\mathbf{N},C)=c/n_1$。这个指标不满足公理 2 和公理 3，因为没有考虑 n_2，也会因为不考虑 C 和 $\mathbf{N}$ 而破坏公理 4。在 $K>2$ 情况下定义的平均度数并不能用于高维度数据。

SVD: 矩阵 $\mathbf{A}$ 的 SVD 是形式为 $\mathbf{A}=\mathbf{U}\Sigma\mathbf{V}^{\top}$ 的分解。标记 $\mathbf{A}$ 的奇异值为 $\Sigma_{r,r}$，奇异向量为 $\mathbf{U}$ 和 $\mathbf{V}$。在高维度数据中前几个奇异值和奇异向量反映了大的密集块和数据簇，所以能够找到可疑行为[99,239]。比如[195] 方法，一个大小为 $n_1\times n_2$、计数为 c 的独立密集块的奇异值为 $\sigma=\frac{c}{\sqrt{n_1 n_2}}=\sqrt{\rho c}$。因为 SVD 能够给数据集的各部分子数据大小不等的奇异值，所以可以被看作可与可疑程度竞争的评价标准。虽然这个指标能够满足定理 1 到定理 3，但很难全部满足。首先，很显然它并没有考虑背景数据的性质，所以不满足定理 4。

那么应该如何拓展这个评价指标到高维度数据中？高维 SVD（HOSVD）并没有和 SVD 一样可证明的保障，也就难以找到最大、最密集的块。对于高维度的密集块来说，每多一个维度，块的体积就会变得更大，密度就会降低一些。这就不符合公理 5，会造成算法总是在尝试把数据映射到单一维度上，而不是考虑在所有 K 个维度之间的关系。在后续实验部分会给出高维度下 SVD 方法的缺点。

由此可以看到基于平均度数的方法和 SVD 方法能够满足其中一大部分的公理，但还是会不满足其中一些公理，造成这种方法在拓展到高维度数据中寻找可以行为时存在局限性。继而给出研究方法并证明这在实际应用中的有效性。

5.3.4　概率测度行为可疑程度的评价指标

本工作所提出的评价指标是基于随机分布的数据模型，也就是数据中的 C 个事件随机分布在数据 $\mathcal{X}$。对于二值数据来说，数据模型是符合二项分布的多维度 Erdös-Rényi 模型。因为通常情况下张量数据中的每一个元素都可以超过 1，所以这里用泊松分布（Poisson distribution）来取代二项分布，由此可以得到 Erdös-Rényi-Poisson 模型：

定义 5.3 (Erdös-Rényi-Poisson (ERP) 模型)： 张量数据 $\mathcal{X}$ 是用 ERP 模型生成的，那么张量中的每一个元素都是从参数为 p 的泊松分布中生成的，即

$$\mathcal{X}_{\mathbf{i}}\sim Poisson(p) \tag{5-37}$$

通常情况下，可以用 p 来表示整个张量数据的密度。用上述模型可以定义可疑程度的评价指标如下：

定义 5.4 (Suspiciousness 评价指标)：　一个高维度密集块的 Suspiciousness 值是其总计数在 Erdös-Rényi-Poisson 模型下存在概率的负对数似然估计。

下面给出一维向量中的 Suspiciousness 评价指标定义。给定 N 长度的、用 EPR 模型生成的一维向量 $\mathbf{X}$，可以假设它是描述每个 IP 地址上的微博数量。如果总共有 C 个微博，密度为 $p = \frac{C}{N}$，每一个元素 X_i 都符合泊松分布，即

$$Pr(X_i|p) = \frac{p^{X_i}}{X_i!}e^{-p} \tag{5-38}$$

于是从中搜索一个长度为 n 的子向量 $X_{i_1},\ldots,X_{i_n}$，该子向量不太可能存在，也就有很高的 Suspiciousness 值：

引理 5.3：　在长度为 N 的向量数据 $[X_1,\ldots,X_N]$ 中，一个长度为 n 的子向量 $[X_{i_1},\ldots,X_{i_n}]$ 的 Suspiciousness 函数为

$$f(n,c,N,C) = c\left(\log\frac{c}{C}-1\right)+C\frac{n}{N}-c\log\frac{n}{N} \tag{5-39}$$

$$\hat{f}(n,\rho,N,p) = n\left(p-\rho+\rho\log\frac{\rho}{p}\right)=nD_{KL}(\rho\|p) \tag{5-40}$$

其中 $c = \sum_{j=1}^{n} X_{i_j}$，$D_{KL}(\rho\|p)$ 是从 ρ 的泊松分布 $Poisson(\rho)$ 到 p 的泊松分布 $Poisson(p)$ 之间的 Kullback-Leibler(KL) 区分度。

证明　这里定义 n 个变量的和为 $Y_n = \sum_{j=1}^{n} X_{i_j}$，根据泊松特性可知，$Y_n \sim Poisson(pn)$。$Y_n$ 值等于微博数量为 c 的概率是

$$Pr(Y_n = c) = \frac{(pn)^c e^{-pn}}{c!} = \frac{C^c}{c!}\left(\frac{n}{N}\right)^c e^{-\frac{Cn}{N}} \tag{5-41}$$

由于阶乘的近似公式，即 Stirling 公式，那么

$$\log(c!) = c\log c - c + O(\log c), \tag{5-42}$$

由此可知 Suspiciousness 的函数为

$$\begin{aligned} f(n,c,N,C) &= -\log[Pr(Y_n=c)] = -\log\left[\frac{C^c}{c!}\left(\frac{n}{N}\right)^c e^{-\frac{Cn}{N}}\right] \\ &\approx c\left(\log\frac{c}{C}-1\right)+C\frac{n}{N}-c\log\frac{n}{N}. \end{aligned} \tag{5-43}$$

□

可以继续把 Suspiciousness 指标扩展为二维矩阵：

引理 5.4：　在大小为 $N_1 \times N_2$、总计数为 C 的数据中，一个大小为 $n_1 \times n_2$、总计数为 c 的二维子矩阵（子数据）的 Suspiciousness 函数为

$$f([n_1,n_2],c,[N_1,N_2],C) = c\left(\log\frac{c}{C}-1\right)+C\frac{n_1n_2}{N_1N_2}-c\log\frac{n_1n_2}{N_1N_2} \tag{5-44}$$

$$\hat{f}([n_1,n_2],\rho,[N_1,N_2],p) = n_1n_2D_{KL}(\rho\|p) \tag{5-45}$$

继而可以扩展到 K 维度的张量数据中：

引理 5.5：　在大小为 $N_1 \times \cdots \times N_K$、总计数为 C 的数据中，一个大小为 $n_1 \times \cdots \times n_K$、总计数为 c 的子张量（子数据）的 Suspiciousness 函数为

$$\boxed{f(\mathbf{n},c,\mathbf{N},C) = c(\log\frac{c}{C}-1)+C\prod_{i=1}^{K}\frac{n_i}{N_i}-c\sum_{i=1}^{K}\log\frac{n_i}{N_i}} \tag{5-46}$$

如果用 ρ 表示密集块的密度，用 p 表示数据集的密度，可以得到更简单的表示

$$\boxed{\hat{f}(\mathbf{n},\rho,\mathbf{N},p) = \left(\prod_{i=1}^{K}n_i\right)D_{KL}(\rho\|p)} \tag{5-47}$$

考虑到 KL 区分度的非负性，可知 $f = \hat{f} \geq 0$。

根据上述的 Suspiciousness 的定义，这里进一步证明该指标函数符合所有在之前给出的公理。

公理 1. 密度公理

证明　采用等式 (5-46)，Suspiciousness 函数关于密集块的总计数 c 的导数为 ①

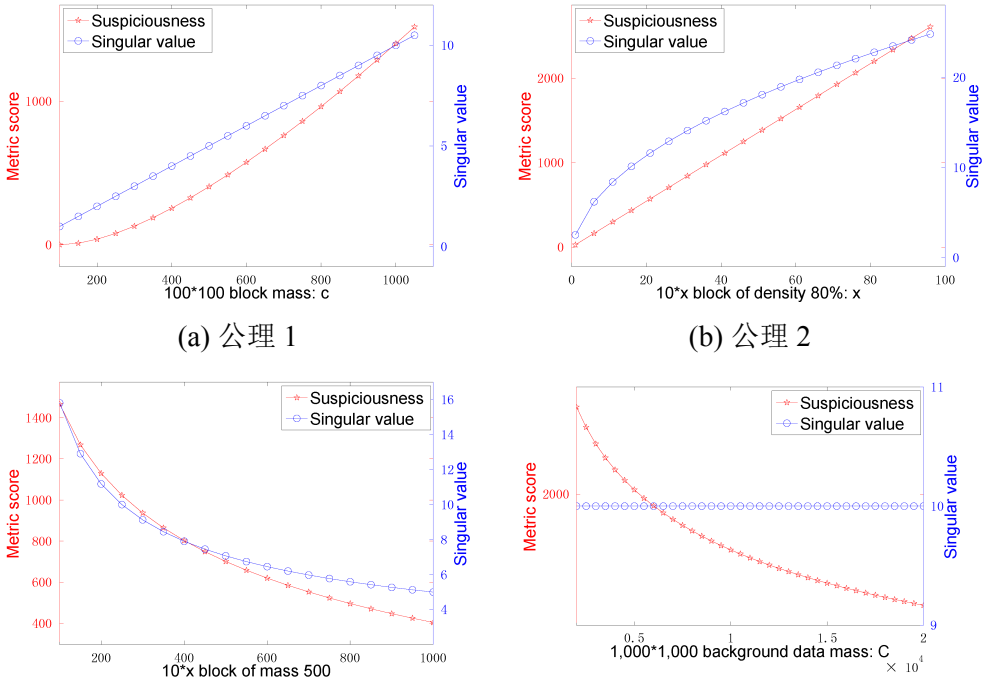
$$\frac{d\hat{f}}{dc} = \log\frac{c}{C}+\log\left(\prod_{i=1}^{K}\frac{N_i}{n_i}\right) = \log\frac{\rho}{p} \tag{5-48}$$

其中 $p = \frac{C}{\prod_{i=1}^{K}N_i}$ 并且 $\rho = \frac{c}{\prod_{i=1}^{K}n_i}$。由于只考虑比数据密度要高的密集块，也就是 $\rho > p$，所以 $\frac{d\hat{f}}{dc} > 0$，所以 Suspiciousness 是随着密度增加要增高的。　□

案例 1. 给定大小为 1,000×1,000、总计数为 10,000 的数据集，可以比较一个大小为 100×100、总计数为 c 的密集块在 c 取不同值时的可疑程度。图5.13(a) 中画出了密集块的 Suspiciousness 和奇异值曲线。可以观察到

- 两者都随着 c 的增大而增大，符合密度更高的密集块会更可疑的假设。当密集块小到几乎为空的时候，两者都接近 0。

① 由于 c 是离散变量，但公式 (5-46) 可以将 c 拓展为连续实数，并通过改变 c 来观察单调性。无论 c 是不是一个整数，实数设定下的公式 (5-46) 是可以满足整数的要求，来证明公式 (5-46) 的单调性。



(a) 公理 1　　(b) 公理 2

(c) 公理 3　　(d) 公理 4

图 5.13　公理 1-4 的仿真表示：(a) 密度越高的密集块就越可疑；(b) 同样密度下，越大的密集块就越可疑；(c) 同样总计数的密集块，越小越可疑；(d) 如果数据集更稀疏了，密集块就更加可疑。

- Suspiciousness 具有凸函数形状的增长，随着 c 的增大，增长的速度会增快，而奇异值是与 c 呈线性关系。

公理 2. 尺寸公理

证明　运用公式 (5-47)，$k \neq j$ 保持 n_k 不变，Suspiciousness 函数对 n_j 的导数为

$$\frac{d\hat{f}}{dn_j} = \left(\prod_{k\neq j} n_k\right) D_{KL}(\rho\|p) > 0 \tag{5-49}$$

那么，当密度 ρ 不变的时候，如果增大密集块任意一个维度的规模，剩余的维度保持不变，那么 Suspiciousness 的数值增大。 □

案例 2. 给定大小为 1,000×1,000、总计数为 10,000 的数据集中，比较同样密度为 80% 时的大小为 10×x 的密集块的可疑程度。图5.13(b) 画出了密集块的 Suspiciousness 值和奇异值。可以观察到

- 两者都随着密集块的规模增大而增大，符合同等密度、更大的密集块会更可疑的假设。当密集块为空的时候，两者都为 0。
- Suspiciousness 值与块的大小呈线性关系，而奇异值是凹函数的增长趋势，也就是随着体积增大，可疑程度会增长得越来越慢。

公理 3. 浓度公理

证明　运用公式 (5-46)，$k \neq j$ 保持 n_k 不变，Suspiciousness 对 n_j 的导数为

$$\frac{d\hat{f}}{dn_j} = \frac{C}{n_j}\prod_{i=1}^{K}\frac{n_i}{N_i} - \frac{c}{n_j} = \frac{c}{n_j}\left(\frac{p}{\rho} - 1\right) \tag{5-50}$$

由于 $\rho > p$，最后一个表达式为负数，那么对于同样的 c，越大的密集块就越不够可疑。 □

案例 3. 给定大小为 1,000×1,000、总计数为 10,000 的数据集中，比较同样总计数为 500、大小为 10×x 的密集块的可疑程度。图5.13(c) 给出密集块的 Suspiciousnesss 和奇异值曲线。可疑观察到两者都随着块的大小增大而减小，因为密集块的总计数不变。两者都有凸函数状单调减小，即块的大小越大，可疑程度减小越慢。

公理 4. 对比公理

证明　运用公式 (5-47)，关于数据密度 p 的 Suspiciousness 函数的导数是

$$\frac{d\hat{f}}{dp} = \left(\prod_{k=1}^{K} n_i\right)\left(1 - \frac{\rho}{p}\right) \tag{5-51}$$

因为 $\rho > p$，$\frac{d\hat{f}}{dp} < 0$，所以随着数据集变得更稠密，密集块就变得更不可疑。 □

案例 4. 给定大小为 1,000×1,000、总计数为 C 的数据集，比较大小为 100×100、计数为 1,000 的密集块的可疑程度。图5.13(d) 给出密集块的 Suspiciousnesss 和奇异值曲线。可以观察到

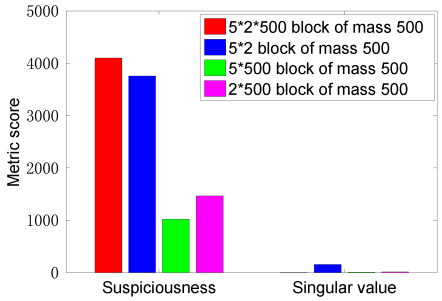
- Suspiciousness 值随着数据密度 p 的增大而减小，随着总计数 c 的增大而减小。该函数以凸函数状减小，随着 p 和 C 的增大，函数值降低速度降低。
- 奇异值只考虑了前景的密集块属性，所以，无论背景数据如何变化，奇异值都保持不变，这与前面所提出的直觉不同。

公理 5. 高维度公理

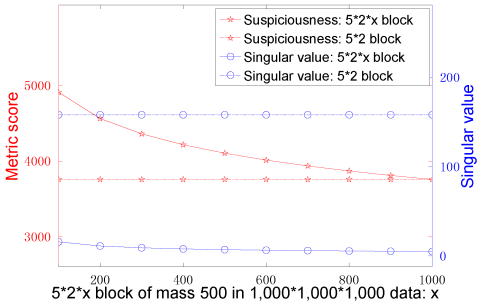
证明　运用公式 (5-46)，可知

$$\begin{aligned} f_K\left(([n_k]_{k=1}^{K-1}, N_K), c, [N_k]_{k=1}^{K}, C\right) &= c\left(\log\frac{c}{C} - 1\right) + C\frac{N_K}{N_K}\prod_{i=1}^{K-1}\frac{n_i}{N_i} - c\left(\log\frac{N_K}{N_K} + \sum_{i=1}^{K-1}\log\frac{n_i}{N_i}\right) \\ &= c\left(\log\frac{c}{C} - 1\right) + C\prod_{i=1}^{K-1}\frac{n_i}{N_i} - c\sum_{i=1}^{K-1}\log\frac{n_i}{N_i} \\ &= f_{K-1}\left([n_k]_{k=1}^{K-1}, c, [N_k]_{k=1}^{K-1}, C\right) \end{aligned} \tag{5-52}$$

□



(a) 高维度密集块比起低维度密集块要更加可疑



(b) 同样总计数时，高维度块比起低纬度块要更加可疑

图 5.14　高维度下的密集块比起低维度下的密集块要更可疑（公理 5）。

给定大小为 $N_1\times\cdots\times N_K$、总计数为 C 的数据集，比较 k 维的大小为 $n_1\times\cdots\times n_k$、总计数为 c 的密集块，以及，增加一个维度，也就是一个大小为 $n_1\times\cdots\times n_k\times n_{k+1}$、总计数为 c 的密集块。低维度密集块和高维度密集块的 Suspiciousness 值分别为：

$$\begin{aligned} & f_k(n_1,\ldots,n_k,c;N_1,\ldots,N_k,C) \\ = \; & c(\log\frac{c}{C}-1)+C\prod_{i=1}^{k}\frac{n_i}{N_i}-c\sum_{i=1}^{k}\log\frac{n_i}{N_i} \end{aligned} \tag{5-53}$$

$$\begin{aligned} & f_{k+1}(n_1,\ldots,n_k,n_{k+1},c;N_1,\ldots,N_k,N_{k+1},C) \\ = \; & c(\log\frac{c}{C}-1)+C\prod_{i=1}^{k+1}\frac{n_i}{N_i}-c\sum_{i=1}^{k+1}\log\frac{n_i}{N_i} \end{aligned} \tag{5-54}$$

定义这两个值的差异如下：

$$g(n_{k+1},N_{k+1})=f_{k+1}-f_k=C\prod_{i=1}^{k}\frac{n_i}{N_i}\cdot\left(\frac{n_{k+1}}{N_{k+1}}-1\right)-c\log\frac{n_{k+1}}{N_{k+1}} \tag{5-55}$$

定义 $x=\frac{n_{k+1}}{N_{k+1}}$，那么

$$g(x)=C\prod_{i=1}^{k}\frac{n_i}{N_i}\cdot(x-1)-c\log x \tag{5-56}$$

$$\frac{dg}{dx}=C\prod_{i=1}^{k}\frac{n_i}{N_i}-\frac{c}{x}=c\left(\frac{p}{\rho}-\frac{1}{x}\right) \tag{5-57}$$

其中 $\rho=\frac{c}{\prod_{i=1}^{k}n_i}$，$p=\frac{C}{\prod_{i=1}^{k}N_i}$。所以，当 $\rho>p$ 时，对于任意的 $0<x\le 1$，有 $\frac{dg}{dx}<0$。那么对于任意的 $1\le n_{k+1}\le N_{k+1}$，可知 $g(0)=+\infty$，$g(1)=0$，而且当且仅当 $n_{k+1}=N_{k+1}$ 时，有 $f_{k+1}\ge f_k$。

案例 5.1. 给定大小为 1,000×1,000×1,000、总计数为 10,000 的数据集，比较 (1) 一个大小为 ×2×500、总计数为 500 的密集块，和其三个在大小为 1,000×1,000、

总计数为 10,000 的数据中的映射密集块，包括 (2) 一个大小为 5×2、总计数为 500 的密集块，(3) 一个大小为 5×500、总计数为 500 的密集块，和 (4) 一个大小为 2×500、总计数为 500 的密集块。图5.14(a) 画出了这些不同维度的密集块的 Suspiciousness 值和奇异值。可以观察到

- 高维度的密集块比起低维度的密集块有更高的可疑程度值。大小为 5×2×500 的密集块比起大小为 5×2 的映射块的可疑程度要高很多。
- 奇异值指标只考虑局部密度和大小，奇异值所给出的大小为 5×2 的二维密集块比起大小为 5×2×500 的三维密集块要高很多。

案例 5.2. 给定大小为 1,000×1,000×1,000 的数据，可以比较 (1) 一个大小为 5×2×x、总计数为 500 的密集块 (1≤x≤1000) 和 (2) 一个大小为 5×2 的、计数相同的映射密集块。它们的 Suspiciousness 值分别为

$$\begin{aligned} f_2(5,2,500;1000,1000,10000) &= 3759 \\ f_3(5,2,x,500;1000,1000,1000,10000) &= 0.0001x - 500\log x + 7212 \end{aligned}$$

图5.14(b) 画出了该二维密集块和三维密集块的 Suspiciousness 值和奇异值曲线。可以观察到

- 同样总计数的高维度密集块比起低维度密集块的 Suspiciousness 值要高。只有当高维度块在第三个维度中用到了所有的值的时候，三维密集块和映射的二维密集块的 Suspiciousness 值是相等的。（见图中两条红色的五角星曲线）
- 同样总计数的低维度密集块比起高维度密集块的奇异值要高，因为低维度密集块的密度要更大。奇异值并不能够合理地比较不同维度的密集块。（见图中两条蓝色圆圈的曲线）只有当高维度块在第三个维度中用到了所有的值的时候，三维密集块和映射的二维密集块的 Suspiciousness 值是相等的（见两条红色的五角星曲线）。

小结： 和奇异值相比，所提出的 Suspiciousness 指标在衡量可疑行为的优势，也就是能够找到高维度数据集中又大又密度高的跨维度密集块。

- 正确的变化趋势：Suspiciousness 指标若干种单调性特征符合所有用于在不同的前景背景和大小密度下衡量可疑行为的公理（公理 1 到公理 5）。奇异值并不能够符合重要的公理 4 和公理 5.
- 正确的拐点值：Suspiciousness 指标在遇到 (1) 密集块尺寸为 0，(2) 密集块计数为空，和 (3) 密集块密度与数据密度相同的情况下，值为 0。另外，高维度的密集块如果在多出来的维度中含有所有的元素，那么和低维度密集块有同样的 Suspicious 值。然而奇异值却并不考虑背景数据的密度，所以对于和数据密度相同的块，奇异值会认为块越大则值越大，不符合直觉。

5.3.5　基于评价指标的局域搜索算法

在定义了密集块的 Suspiciousness 指标后，可以定义跨维度可疑块检测的问题，并基于所提出的指标，给出检测可疑块的可扩展算法。问题定义如下：

问题 5.3 (可疑块检测)：**给定：** 大小为 $N_1 \times \cdots \times N_K$、总计数为 C 的数据集 $\mathcal{D}$，**找到：** 数据 $\mathcal{D}$ 中任意一个维度组合下一系列 Suspiciousness 值高的密集块，基于公式 (5-46) 和公式 (5-58) 呈从高到低的逆序。

这里依旧用 K 维度张量数据 $\mathcal{D}$ 中 k 维度的子张量 $\mathcal{Y}$ 来表示可疑块。子张量在维度 j 上有 N_j 个可能的值：$\mathcal{P}_j = \{p_1^{(j)}, \ldots, p_{N_j}^{(j)}\}$。子张量 $\mathcal{Y}$ 覆盖了每一个维度的一组值：$\tilde{\mathcal{P}}_j \subseteq \mathcal{P}_j, \forall j$。定义 $\tilde{\mathcal{P}} = \{\tilde{\mathcal{P}}_j\}_{j=1}^K$。用 $c(\tilde{\mathcal{P}})$ 表示子张量 $\tilde{\mathcal{P}}$ 中的事件次数。

密集块 $\mathbf{n}$ 的某维度长度为 $n_j = |\tilde{\mathcal{P}}_j|$。如果维度 j 并没有被考虑，那么可以根据公理 5 和映射性质来考虑 $\tilde{\mathcal{P}}_j = \mathcal{P}_j$。由此定义参数化的 Suspiciousness 函数如下

$$\tilde{f}(\tilde{\mathcal{P}}, \mathcal{D}) = f([|\tilde{\mathcal{P}}_j|]_{j=1}^K, c(\tilde{\mathcal{P}}), [|\mathcal{P}_j|]_{j=1}^K, |\mathcal{D}|) \tag{5-58}$$

由此可以设计在数据集中局域搜索可疑密集块的算法 CrossSpot。从一个种子可疑块出发，迭代优化，每一步最优化在维度 j 中的一组值，同时保持其他维度的值。如此更新可疑块知道收敛。完整的算法（算法12）如下所示：

Algorithm 12 局域搜索可疑块算法 CrossSpot

Require: 数据集 $\mathcal{D}$，种子可疑块 $\mathcal{Y}$，即 $\tilde{\mathcal{P}} = \{\tilde{\mathcal{P}}_j\}_{j=1}^K$

```
while 还未收敛 do
    for j = 1...K do
        P̃_j ← 调整维度(j)
    end for
end while
return P̃
```

调整维度： 在 “调整维度” 的每一次迭代中，保持其他维度（对于 $j' \neq j$ 的 $\tilde{\mathcal{P}}_{j'}$）不变，优化 $\mathcal{P}_j$ 中的值集合，也就是说保持不变。定义 $\Delta c_{p_i^{(j)}}$ 为在维度 j 的第 i 行中的事件次数，那么可以用 $\Delta c_{p_i^{(j)}}$ 表示 $p_i^{(j)}$ 的增益。在 “调整维度” 函数13中用增益程度对 $\mathcal{P}_j$ 中的所有值由大到小排序，并定义排序后的列表为 $\mathbf{P}_j$。

引理 5.6：对于所有的 $j' \neq j$，保持 $\tilde{\mathcal{P}}_{j'}$ 不变，$\tilde{\mathcal{P}}_j \subseteq \mathcal{P}_j$ 中的值的优化选择是选取最靠前的 n_j 值。

Algorithm 13 调整维度 (j)

1: $\tilde{\mathcal{P}}'_j \leftarrow \{\}$;
2: $\mathbf{P}_j \leftarrow \{p_i^{(j)}\}_{i=1}^{N_j}$ 按照 $\Delta c_{p_i^{(j)}}$ 逆序排列
3: **for** $p_i^{(j)} \in \mathbf{P}_j$ **do**
4: $\quad \tilde{\mathcal{P}}'_j \leftarrow \tilde{\mathcal{P}}'_j \cup p_i^{(j)}$
5: $\quad \tilde{\mathcal{P}}' \leftarrow \{\tilde{\mathcal{P}}_{j'}\}_{j' \neq j} \cup \tilde{\mathcal{P}}'_j$
6: $\quad$ **if** $\tilde{f}(\tilde{\mathcal{P}}, \mathcal{D}) \leq \tilde{f}(\tilde{\mathcal{P}}', \mathcal{D})$ **then**
7: $\quad\quad \tilde{\mathcal{P}}_j \leftarrow \tilde{\mathcal{P}}'_j$
8: $\quad$ **end if**
9: **end for**
10: **return** $\tilde{\mathcal{P}}_j$

证明 用反证法证明。假设 $\tilde{\mathcal{P}}_j \subseteq \mathcal{P}_j$ 已经是在维度 j 上最优的值集合，但是 $\tilde{\mathcal{P}}_j$ 并不是 $\mathbf{P}_j$ 最好的 $|\tilde{\mathcal{P}}_j|$ 值。所以，一定会存在一对值 $p_i^{(j)}, p_{i'}^{(j)}$，其中 $p_i^{(j)} \in \tilde{\mathcal{P}}_j$ and $p_{i'}^{(j)} \notin \tilde{\mathcal{P}}_j$ but $\Delta c_{p_{i'}^{(j)}} > \Delta c_{p_i^{(j)}}$。从公理 1 中知道，移除值 $p_i^{(j)}$，而增加一个值 $p_{i'}^{(j)}$ 到 $\tilde{\mathcal{P}}_j$ 中后所得到的密集块比起之前的、假设是最优的密集块会有更高的 Suspiciousness 值。由此反正得知，$\tilde{\mathcal{P}}_j$ 中最优的值选择是 $\mathbf{P}_j$ 中靠前值。 □

定理 5.2： 对于所有 $j' \neq j$，保持 $\tilde{\mathcal{P}}_{j'}$ 不变，算法12中 “调整维度 (j)” 的操作能够在 $\tilde{\mathcal{P}}_j$ 中最大化 $f(\mathbf{n}, c, \mathbf{N}, C)$。

证明 因为 “调整维度” 能够对 $\mathcal{P}_j$ 排序，检测维度 j 中每一个可能的 n_j 值，公式5.6表明每一步 “调整维度” 都能够优化该维度值的选择。 □

种子密集块选取： 算法12是从种子子张量（块）$\mathcal{Y}$ 出发的，最简单的情况下是从随机选取的种子集合，或是张量数据中的某一个单独的元素，或者是随机选取的较大的密集块。如后续实验显示，随机选取的种子都甚至能达到很好的效果。算法 CrossSpot 在选取种子时是非常灵活的，能够从之前的数据挖掘工作或是辅助信息中找到启发。例如，可以从奇异值分解（SVD）所返回的排序中选择密集块。当有多个种子密集块的时候可以并行化地搜索可疑块，这样能够充分调用计算资源，来用更多的随机种子，找到最合适的结果。

计算复杂度： 算法12（CrossSpot）的计算复杂度是 $O(T \times K \times (E + N \log N))$，其中 T 是迭代次数，K 是维度数量，E 是数据中的非零元素的数量，$N = \max_j N_j$ 是任意维度中最大的长度。因为 T 和 K 通常被设置为常数，所以计算复杂度是与非零元素的数量呈线性的。CrossSpot 在真实的检测可疑行为的应用中是可扩展的。

收敛保证： 通过定理5.2知，CrossSpot 算法能够收敛到局域最优。每一次调整维度操作后，$\tilde{f}(\tilde{\mathcal{P}}, \mathcal{D})$ 的值或是不变或是增加。那么因为子张量的实际数量是有限的，目标选择也是有限的，所以算法终将收敛。

5.4　性能评测

本节从以下三个方面评测本章所提出工作的性能：一是具有同步行为的可疑用户检测效果，二是具有密集行为的可疑用户检测效果，三是信息操纵行为检测效果。

5.4.1　具有同步行为的可疑用户检测性能

本小节给出 CatchSync 的实验结果，证实了算法在检测可疑行为的有效性。许多异常检测算法都把这个问题作为标记性数据上的任务；在真实数据中，异常检测是机器学习、人工验证和新攻击类型分析的融合。这里把问题当作可疑行为检测下的分类问题来证明算法有效，同时给出可疑行为模式的样例分析。

- 检测有效性： 证实了算法标记可疑行为的能力，并且通过下面三种任务来移除可疑行为。(1) 检测攻击行为：通过在仿真图中注入组攻击，给出准确率、召回率，与时下最先进的算法作比较证明算法更加有效。(2) 标记任务：在真实数据上人工标记随机账户是否可疑，然后用算法检测。(3) 还原正常的模式：通常图的度数分布是幂律分布，移除可疑节点后能够还原图中的分布。同时还能够还原特征空间的分布，去除异常。
- 算法属性： 测试 CatchSync 的算法属性包括关于参数 α 的鲁棒性，运行速度和可扩展性。
- 异常发现： 用 CatchSync 作为工具来研究真实网络，从 Twitter 和腾讯微博上找到了非常特别的异常账户，并汇报他们之间的行为模式。

下面仿真数据和真实数据上测试 CatchSync 算法的效果。仿真数据集在表5.11中给出，真实数据集在表5.12中给出。仿真数据集的描述如下。依照 Chung-Lu 模型[260] 生成随机幂律图。在此模型里先给点 u 和 v 假设出度 $d_o(u)$ 和入度 $d_i(v)$，然后按照正比于概率 $d_o(v)d_i(u)$ 生成边 (u, v)，其中幂律指数为 -1.5。因为真实网络中就是这样的取值[258]。接着注入源节点 -目标节点的组攻击。

为了证明算法的有效性，对仿真图数据的特性讨论如下：

- 图的大小： 随机产生幂律图，图中包含大约 100 万、200 万和 300 万个节点，称为 Synth-1M, Synth-2M 和 Synth-3M。

	点数	注入 源节点	注入 目标节点	伪装 类型	伪装 比例
Synth-1M	1,034,100	31,000	1,000	-	-
Synth-2M	2,034,100	31,000	1,000	-	-
Synth-3M	3,034,100	31,000	1,000	-	-
Synth-Rand1	3,034,100	31,000	1,000	随机伪装	+10%
Synth-Rand5	3,034,100	31,000	1,000	随机伪装	+50%
Synth-Pop1	3,034,100	31,000	1,000	流行伪装	+10%
Synth-Pop5	3,034,100	31,000	1,000	流行伪装	+50%

表 5.11　仿真数据：在从 100 万到 300 万节点的仿真随机幂律图中，注入 5 组不同大小的攻击，或是有伪装，或是没有伪装。

	点数	边数	人工标注的可疑源节点
TwitterSG	41,652,230	1,468,365,182	173 / 1,000
WeiboJanSG	117,288,075	3,134,074,580	237 / 1,000
WeiboNovSG	353,509,867	12,168,482,951	

表 5.12　Twitter 和腾讯微博上的真实数据集：真实社交网络是包含百万级的用户和亿级关系的社交关系图。手动地随机标记了一小部分用户作为真实值。

- 注入组的大小和数量： 注入不同大小的 5 组源节点 -目标节点的组攻击。最小的是 1,000 个新的源节点，随机从 100 个新的目标节点中选取 20 个连接。因为真实情况下，售粉公司的规模往往至少有 1,000 个假冒粉丝账户。注入的组的大小逐步翻倍，从 2,000，4,000 到 8,000，而最大的组是 16,000 个源节点和 1,600 个目标节点。于是注入的总源节点数目是 31,000。
- 伪装类型和大小: 注入的源节点是可以用伪装来绕开检测算法的，比如说，假冒粉丝是可以关注像美国总统奥巴马一样知名的名人，或者是关注随机用户，虽然他们已经连接了几十或者几百个顾客。受到启发的是，在 Synth-3M 上尝试这两种不同的伪装方法。对于一个被注入的源节点，它可以连接到或是随机的、普通目标节点，或是前 100 个高入度的目标节点。允许改变伪装的权重 d_{camou}：或是 $d_{camou} = 10\%$，也就是 18 个注入的目标用户以及 2 个伪装目标用户；或是 $d_{camou} = 50\%$，也就是 10 个注入的目标用户以及 10 个伪装目标用户。特别的，给有 10% (50%) “随机” 伪装的仿真数据命名为 Synth-Rand1(Synth-Rand5)；给有 10% (50%) “流行” 伪装的仿真数据命名为 Synth-Pop1(Synth-Pop5)。

由上面不同的设置可得 5 个仿真数据集。在仿真数据集上有正例也有负例，于是可以用常用的评价指标，包括精确度（accuracy），准确率（precision）和召回率（recall）[117]。这些值越高，检测效果越好。

除了仿真数据，还有三个真实数据集，包括 TwitterSG，WeiboJanSG 和 WeiboNovSG。这些图都是有亿级边的流行在线社交媒体中的完整图。CatchSync 可以在 TwitterSG 的数据集 ① 上得到重现。如同网页所说，因为 Twitter 的新协议条目，学术研究者很难接触到如微博内容等附加信息。幸好在多数应用中只需要有向图，比如谁关注谁的数据，所以开发出无需附加信息的方法 CatchSync。WeiboJanSG 是在 2011 年 1 月份从中国最大的微博平台之一腾讯微博爬取的，WeiboNovSG 是 2011 年 11 月份从同一平台得到的。对于每一个数据集，CatchSync 都只需要使用图结构信息。腾讯微博的数据中有用户 ID、昵称以及他们的一些个人信息来证明用户的可疑与否。从 TwitterSG、WeiboJanSG 和 WeiboNovSG 中采用出 1,000 个节点，并进行手工标注，标注为可疑的还是正常的用户。有一半的节点是随机从集合 U_{sync} 中产生，另外一半则是从剩余集合得来。虽然采样中的整体可疑比例比起整个数据集都高，实验中每一个算法都是公平的。5 名志愿者都是 20 到 25 岁的大学学生，他们都至少有 3 年的社交媒体使用经验。实验中提供给他们这 1,000 个用户的 Twitter 或者是腾讯微博的主页链接。他们阅读微博信息和个人介绍，依据下面线索判断用户是否可疑：

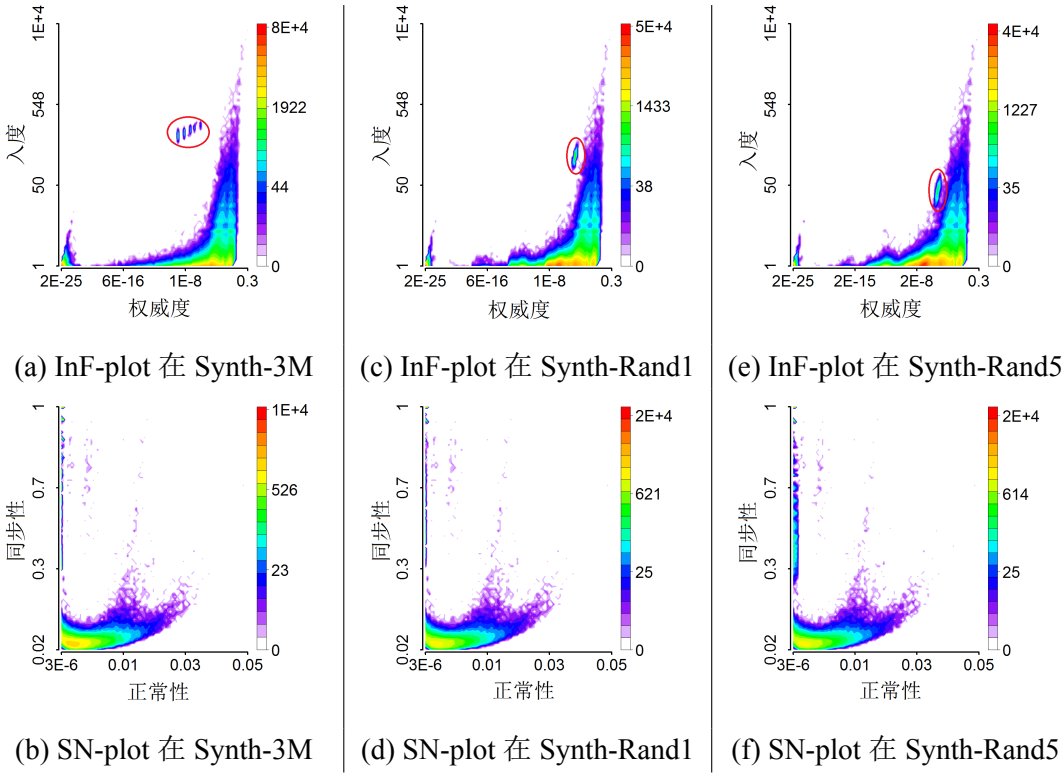
- 失效账户： 是说已经被微博万展禁止的账户。比如说在腾讯微博上的用户@marra_xiao_bai 在 2011 年有 9 个粉丝和 36 个关注的人，Twitter 上的用户@wYWvk0310 曾经在 2010 年有 666 个粉丝和 926 个关注的人，但是他们的账号都被禁止使用了。
- 可疑的用户昵称： 用户自己使用的名字往往也是非常可疑的，尤其是他们是机器人设置成的形式：Twitter 上的 @“Buy_XX##” (@Buy_AB22, @Buy_BT47)，或者是腾讯微博上的 “a#####” (@a58444, @a70054)。
- 许多账号有很多来关注的人却没有发微博： 有很多关注的人，却从没有发过一条微博，比如说 Twitter 用户@P8igBg801 在 2010 年有 923 个关注的人，以及@AjaurNYj2 有 869 个关注的人，但是他们都没有发过微博。
- 可疑的微博内容： 有些账号重复发一些内容和链接，是为了获得经济利益。例如 Twitter 用户@Buy_BT66 只发布了 3 条消息，但全都是关于房间出租的，腾讯微博用户@aa52011 发了上百条关于在线游戏的消息。

① http://an.kaist.ac.kr/traces/WWW2010.html

如果有超过一半，也就是 3 个志愿者认为账号可疑，就认为这个账号可疑。任务是从关注关系中检测出这些可疑的账户。和在仿真数据中相似的是，用精确度、准确率和召回率来衡量有效性。一个好的算法有更高的精确度、准确率和召回率。

工作中仔细地实现了下面的先进算法作为基线方法：(1) OddBall[205]，是从图中寻找接近完全图和星状的子图；(2) OutRank[204]，用随机漫步算法通过节点相似性得到节点的可疑程度；(3) SpokEn[239]，用特征向量对来找到紧密连接起来的社区子图。利用被标记好的真实数据，借助基于内容的垃圾传播者算法 SPOT[183]，来学习他们发布微博的文本和可疑链接。CatchSync 和如 SPOT 的基于文本方法非常不同，于是给出了混合方法 CatchSync+SPOT，既怀疑 CatchSync 给出的可疑节点，也怀疑 SPOT 给出的可疑节点。这个混合方法能够从图结构特征和文本特征中找到可疑微博账户。所有的算法都用 Java 实现，实验在有 2.40GHz CPU 和 32GB 内存的单一机器上运行。

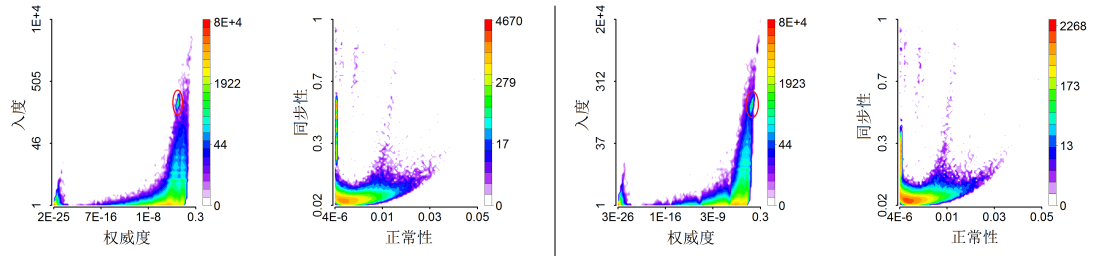
首先给出在仿真数据上的实验结果：在特征空间和同步性 -正常性的图中检测注入的组攻击，给出准确率和召回率的结果以及还原幂律出度分布的效果。



(a) InF-plot 在 Synth-3M　(c) InF-plot 在 Synth-Rand1　(e) InF-plot 在 Synth-Rand5

(b) SN-plot 在 Synth-3M　(d) SN-plot 在 Synth-Rand1　(f) SN-plot 在 Synth-Rand5

图 5.15　虽然有随机伪装，CatchSync 也能够检测出这些可疑节点：(a) 中的数据集没有伪装，(b) 中的 SN-plot 很容易找到这些注入的节点。伪装是能够把注入的节点藏起来，靠近 (c) 和 (e) 中的主体，但是 SN-plots 还是能够在 (d) 和 (f) 中抓到可疑节点。



(a) InF-plot 在 Synth-Pop1　(b) SN-plot 在 Synth-Pop1　(c) InF-plot 在 Synth-Pop5　(d) SN-plot 在 Synth-Pop5

图 5.16　虽然有流行伪装，CatchSync 能够检测出可疑节点：与随机伪装不同的是，流行伪装能够使得被注入的节点与正常节点太像。所以在特征空间中很难找到它们：在 (a) 和 (c) 的主体中已经融入了可疑节点。而在 (b) 和 (d) 中 SN-plots 是能够抓到这些节点。

图5.15(a) 给出仿真图 Synth-3M 的特征空间。相比于大多数点来说，注入的点形成了异常的组群。在图5.15(b) 中的 SN-plot 展现出 CatchSync 算法成功地把这些注入节点放在同步性的轴旁边，也就是有近乎 0 的正常性和非常大的同步性值。接着在5.15和图5.16中，给出 SN-plots 是如何在随机伪装和流行伪装中找到这些注入节点的。对于随机伪装来说，当伪装的权重增大的时候，从 Synth-Rand1 中的 d_{camou}=10% 到 Synth-Rand5 的 d_{camou}=50%，注入的目标节点越来越靠近 InF-plot 中的主体部分，看图5.15(c) 和图5.15(e)。那么可以看到图5.15(d) 和图5.15(f) 能够把可疑的注入节点放到高的同步性和极端小的正常性的位置。对于流行伪装来说，当伪装的权重增大的时候，从 Synth-Pop1 中的 d_{camou}=10% 到 Synth-Pop5 的 d_{camou}=50%，注入的目标节点越来越靠近 InF-plot 中的主体部分，看图5.16(a) 和图5.15(c)。那么可以看到图5.16(b) 和图5.16(d) 能够把可疑的注入节点放到高的同步性和极端小的正常性的位置。总而言之，CatchSync 算法能够从 SN-plots 中准确的找到注入的可疑节点，虽然节点们有着机智策略来把它们隐藏在 InF-plots 中。

仿真数据	Synth-1M	Synth-2M	Synth-3M
CatchSync	**0.998**	**0.987**	**0.956**
OddBall	0.827	0.796	0.755
OutRank	0.805	0.777	0.725
SpokEn	0.695	0.682	0.677

表 5.13　CatchSync 一直比基线算法好：能够达到近乎 100% 的准确率，无论是对 100 万、200 万还是 300 万的仿真数据。

表5.13中给出在有 100 万到 300 万节点的 3 个仿真数据上的准确率。如果没有伪装，CatchSync 是能够有超过 95% 的准确率的。表5.14中给出了如果对图中加入随机伪装或者流行伪装，CatchSync 比起其他基线算法逗号。CatchSync 算法在

仿真图	Synth-Rand1	Synth-Rand5	Synth-Pop1	Synth-Pop5
伪装 (d_{camou})	10%	50%	10%	50%
CatchSync	**0.910**	**0.764**	**0.885**	**0.792**
OddBall	0.702	0.525	0.657	0.433
OutRank	0.678	0.516	0.694	0.392
SpokEn	0.586	0.470	0.553	0.351

表 5.14　CatchSync 一直都更好，即使有伪装：虽然仿真图被随机伪装和流行伪装注入了 10% 到 50% 的可疑节点。CatchSync 能够很准确地找到这些可疑节点。

Synth-Rand1 上能有 29.6% 的准确率，在 Synth-Pop5 上能有 27.5% 的准确率。

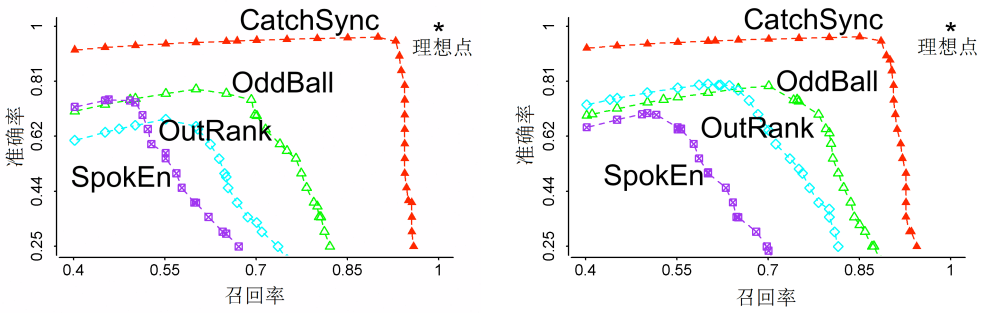
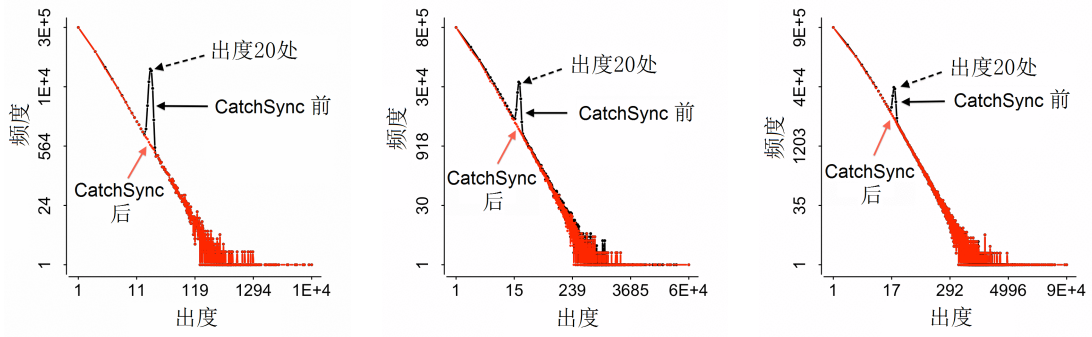


图 5.17　CatchSync 能够给出更高的准确率和召回率：在有 10% 的随机伪装和 50% 的流行伪装的仿真图上。

表5.17中画出在检测可疑节点时的准确率 -召回率曲线。CatchSync 方法（红色实三角）能够既有很高的准确率，又有很高的召回率。



Synth-1M　　Synth-2M　　Synth-3M

图 5.18　CatchSync 方法能够恢复幂律的出度分布：通过移除可以节点，CatchSync 能够将正常的出度分布恢复。

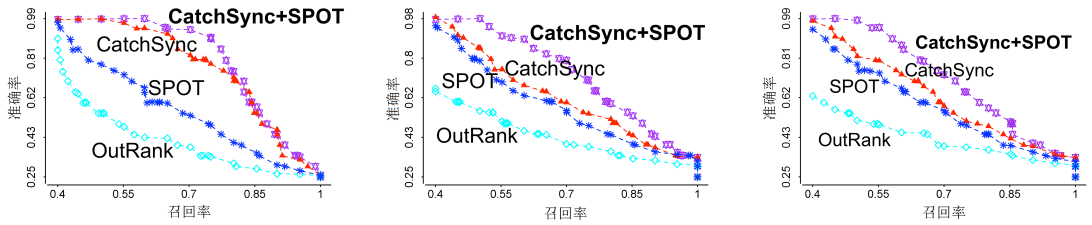
所注入的异常节点是与接近 20 个目标用户相连的，所以根据这种行为模式，在出度分布上出度为 20 的地方会有尖峰。图5.18中给出了使用 CatchSync 算法移除可疑节点的前后，所得到的出度分布。随着图的规模越来越大，因为注入的节

点数量不变，所以尖峰会越来越小。无论尖峰的大小是多少，CatchSync 都能够检测到注入的节点。可以看到通过移除这些节点和它们所形成的边，幂律形状的出度分布是可以在数据中被恢复的。

接下来给出 CatchSync 在真实数据上的实验效果。首先给出准确率和召回率等数值，接着检查是否能够还原幂律分布，最后展现特征空间的变化。

	TwitterSG	**WeiboJanSG**	**WeiboNovSG**
CatchSync	0.751	0.694	0.654
OutRank	0.412	0.377	0.336
SPOT	0.597	0.653	0.611
CatchSync+SPOT	**0.813**	**0.785**	**0.709**

表 5.15　混合方法 CatchSync+SPOT 比起其他任何一种方法都好：CatchSync 在从图结构信息中学习行为模式的效果上比起 OutRank 要好，而 SPOT 则是在学习基于文本的特征。两者相结合会能做到最好。



TwitterSG　　WeiboJanSG　　WeiboNovSG

图 5.19　混合方法能够在检测可疑节点方面做到既有最好的准确率又有最好的召回率。

表5.15展现了从三个真实社交媒体中找到标记为可疑节点的准确率。图5.19画出了准确率 -召回率曲线来比较 CatchSync，OutRank，SPOT 和混合算法 CatchSync+SPOT。从实验结果中观察到了这样的现象并给出解释：

- CatchSync 比起 OutRank 效果要好： 同样是基于图的特征，OutRank 使用图中两点相似度的方法利用随机漫步模型来找到可疑节点，而 CatchSync 通过学习同步性行为的特征找到可疑节点，所以 CatchSync 更为直接有效。
- CatchSync 比起 SPOT 要更好： CatchSync 从结构化信息中学习基于图的特征，而 SPOT 则是从用户所发出的微博中学习基于文本的特征。从可疑用户形成的组攻击主要特征中知道，CatchSync 有更好的准确率和召回率。

实际上，CatchSync 是与 SPOT 互补的：通过整合这两个方法所发现的可疑节点，能够得到更好的实验效果（图5.19中的紫色的线）。混合算法能够同时用上述两个方法找到不同类型的攻击者。CatchSync+SPOT 比起其他的基线算法都有更好的效果：在 TwitterSG 上能够提升 36% 的准确率，在 WeiboJanSG 上能够提升 20%，在

WeiboNovSG 上能够提升 16%。这里分析 CatchSync 算法没有预测准确的项，以期以后更好地检测社交媒体的可疑账户：

- CatchSync 预测不出的可疑用户（False Negatives）通常会被 SPOT 找到：因为存在大量可疑用户并没有帮助顾客得到很多粉丝，但是他们大肆散播垃圾信息，这些信息是同样文本或者是站外链接，所以基于文本的方法会找到它们。后续可以将文本特征融入 CatchSync 的特征空间，以期更好的效果。
- CatchSync 所找到的某些可疑账户（False Positives）很难被人工标记出。因为就如之前所说，这些账户单个拿出来，其个人信息和发布内容都并不可疑，但是放到一起就能看出他们是成组地，一致性的去关注同一批用户。这证实了组攻击是很难被肉眼发现的，但 CatchSync 算法能攻克这一难题。

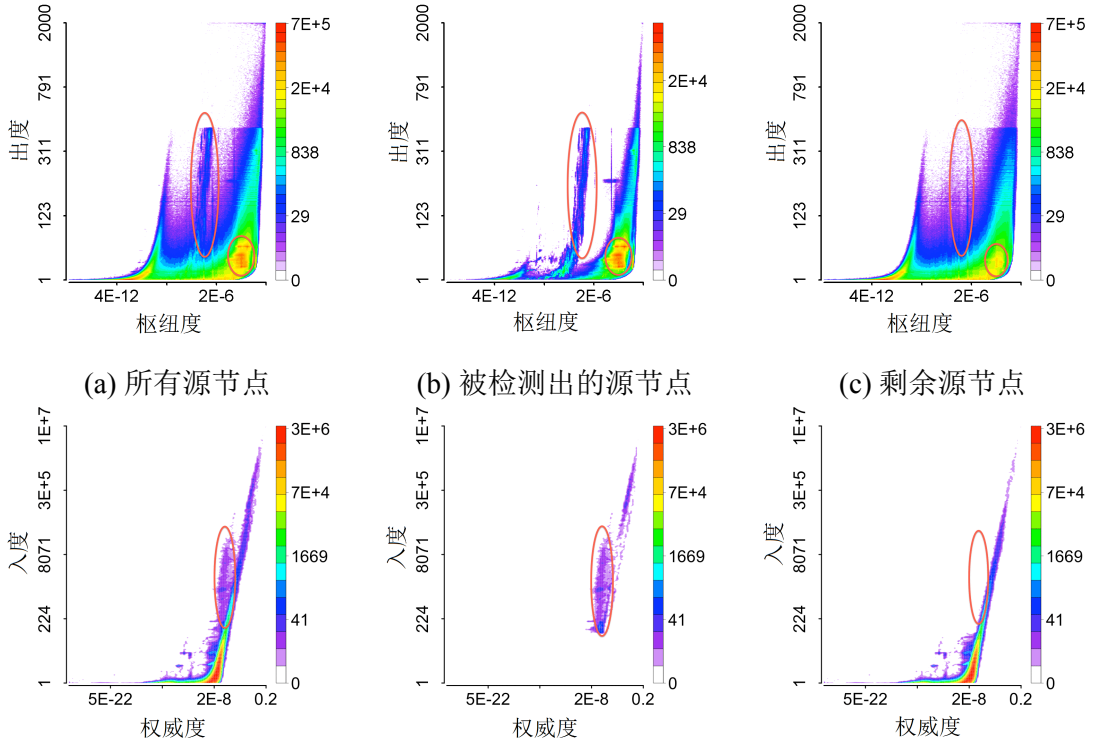
所以建议社交媒体网站能够在他们的服务中使用 CatchSync 算法，并与想 SPOT 一样的基于文本的方法同时使用，这样能够从内容和行为两个方面找到可疑账号。

CatchSync 在仿真数据集上可以还原被扭曲了的出度分布，在真实数据集上一样能够做到这一点：图5.1(b-d) 分别是 Twitter、WeiboJanSG 和 WeiboNovSG 上删除可疑节点前后的出度分布变化。通过移除可疑节点，图中剩余部分形成顺滑的幂律分布。因为幂律的出度分布被看作社交媒体的标志性特征，能够还原本来面目证明其在大规模数据上有效抓住可疑部分。

CatchSync 在 WeiboJanSG 上特征空间的变化中存在很有趣的现象。图5.20(a)，图5.20(b) 和图5.20(c) 在 OutF-plots 上组成了等式：所有节点减去检测出的可疑节点（带同步性行为）等于正常节点。图5.20(b) 显示出在 OutF-plot 中可疑节点看上去同步并且异常。他们往往都聚集在与大部分节点不同的地方，或是红色的一小撮，或是蓝色的条状。而这些异常在图5.20(c) 中消失了。图5.20(d)，图5.20(e) 和图5.20(f) 在 InF-plots 中组成了相似的等式。图5.20(e) 把异常节点放在了紫色的一簇上，而在图5.20(f) 中可疑节点被移除后这个簇都消失了。上述观察能够给很好的证据说明可疑节点往往有同步性行为。CatchSync 方法能够消除特征空间奇怪的模式。

接下来讨论 CatchSync 算法特性。首先是参数 α 的鲁棒性；接着是格子数量 G 的鲁棒性；最后证实算法有很快的运行速度，也是随着数据规模变化可扩展。

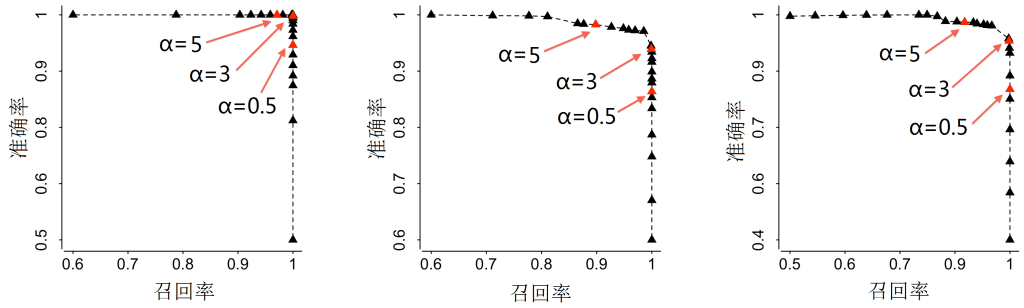
通过仿真数据的实验来测试 α 的鲁棒性，也就是说根据多少倍的标准差区标记可疑节点。可以发现 $\alpha = 3.0$ 能够给出最好的结果，或者是接近最好结果。细节上来说，在三个不同大小的仿真数据上，给出准确率 -召回率关于 α 的变化。图5.21给出了准确率 -召回率曲线，最理想的位置是 (1.0,1.0)；把 α 从 0.5 变化到 5.0，准确率和召回率总是超过 0.8 的。CatchSync 算法的效果对于 α 来说是稳定的。于是对于所有实验中，都设置 $\alpha = 3.0$ 为默认参数。当标记节点时用 3 倍标准



(a) 所有源节点　(b) 被检测出的源节点　(c) 剩余源节点

(d) 所有目标节点　(e) 被检测出的目标节点　(f) 剩余目标节点

图 5.20　被 CatchSync 找到的源节点和目标节点往往是异常：(a,b,c) 和 (d,e,f) 分别组成了特征空间变化的两个等式。(a) 减去 (b) 等于 (c); (d) 减去 (e) 等于 (f), 其中 (a,d) 表示所有节点, (b,e) 表示被检测出的节点, 而 (c,f) 表示正常节点。

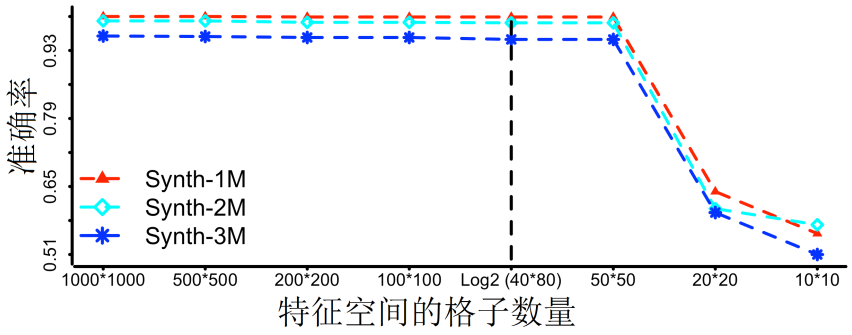


100 万节点图　200 万节点图　300 万节点图

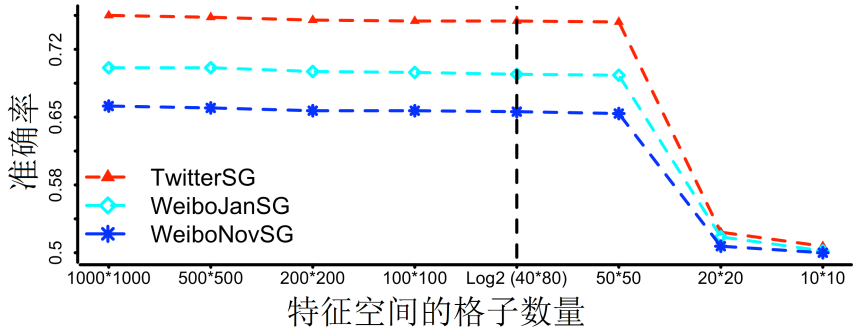
图 5.21　接近完美的鲁棒性：CatchSync 对于 α 并不敏感，设置 $\alpha = 3.0$ 作为默认。

差（也就是 99.7% 的正态分布范围）的时候，可疑节点只取 0.3% 范围的部分，但对于整个百万节点数的大规模图来说也是很多的节点。

为了保证计算同步性和正常性数值的可扩展性，采用近似方法加速算法运行。随着特征空间中的格子数量 G 从 1000^2 减小到 500^2, 200^2 ..., 测试检测效果，也就是检测可疑节点的准确率，在数据集包括 Synth-1M, Synth-2M 和 Synth-3M 上，即图5.22(a) 中画出格子数量与有效性的关系。在图5.22(b) 中画出真实数据包括



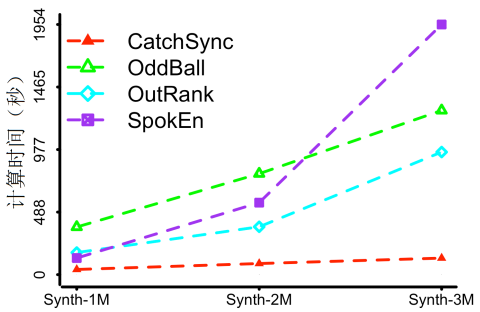
(a) 仿真数据



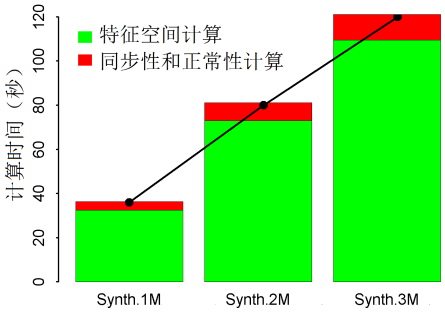
(b) 真实数据

图 5.22　关于格子数量 G 的鲁棒性：注意到将格子数量从 1000^2 减低到 $500^2, 200^2, \ldots$，检测结果一直能够给出很高的准确度。$b = 2$ 是使用算法时一个很好的设置，然而，当 G 小到 $20 \times 20, 10 \times 10$ 时，CatchSync 是不能找到可疑节点的。

TwitterSG, WeiboJanSG 和 WeiboNovSG 的结果。当 G 比 50^2 要更大的时候，算法是具有很好的鲁棒性的。但是 G 比起 20^2 要小的时候，算法效果很差。这里选择 2 作为 b 的默认值；格子的默认数量为 40×80，效果和 $G = 1000^2$ 时是一样好的。



(a) 运行时间



(b) 不同阶段所耗时间

图 5.23　CatchSync 算法很快而且可扩展：(a) 运行时间随着图规模变化的结果；(b) 在计算特征空间和同步性 -正常性图的运行时间。

在有 100 万节点到 300 万节点的仿真图中测试运行时间。图5.23(a) 中画出了图规模和运行时间的关系，CatchSync 的效率随着图大小呈线性关系（如红色实三

角所示），比起其他算法都快。图5.23(b) 中画出在不同阶段的运行时间。同步性和正常性的计算非常快，只需要用计算特征（度数和 HITS 值）的 10% 的时间。如果这些特征能够很早地选择并计算，那么算法是可以快到在线计算大规模图的同步性和正常性的。

如同之前介绍的，检测可疑节点不只是标记问题。在真实生活中存在各种新的攻击方法会把微博服务扭曲。本工作证实了 CatchSync 能检测最常见的垃圾传播行为，也能够找到普通标记算法会错失的同步行为。从网上实际内容看，Twitter 上的欺诈行为比起个体用户发出可疑微博要复杂得多。可以看到僵尸粉能够从顾客得到粉丝中获得收益 ① ②。这会导致购买僵尸粉市场的出现，于是给政治家或者是其他名人以虚假的知名度 ③，同时会使 “Tweeter” 的价值升高，存在能从售卖僵尸粉获益的公司 ④。

图 5.24　CatchSync 的真实运行结果：只是用图结构信息，找到的最大的组中是含有 91,035 个粉丝和 667 个关注的人的。这里给出 3 个粉丝在左边，4 个被关注的人在右边。附件信息能够说明他们的可疑性：这 3 个粉丝几乎没有微博，而他们的社交关系的数量是非常相似的。4 个关注的人所给出的站外链接已经被 Twitter 标记为不安全。

市场是非常复杂的，僵尸粉很难被手工标记出来。CatchSync 所找到的 Twitter 账户是有相近的用户名的，并且通过附加信息来说明这些找到的行为是可疑的。图5.24 从 CatchSync 找到的大规模的可疑组（包括 91,035 个粉丝和 667 个关注的人）中给出了一个小集合（3 个粉丝和 4 个被关注的人）。这里给出 3 个粉丝在左边，4 个被关注的人在右边。附件信息能够说明他们的可疑性：这 3 个粉丝几乎没有微博，而他们的社交关系的数量是非常相似的。4 个关注的人所给出的站外链接已经被 Twitter 标记为不安全。单独给出其中一个账户，不会觉得他们有多可疑。然而把这些账户同时给出时，他们是非常可疑的。首先给出的三个粉丝是@AjaQwX1Z3，@AjaurNYj2和@mastertwitlist。这三个账户有非常相似的名字，

① PaidPerTweet - Get Paid For Tweets. http://paidpertweet.com
② Twitter Advertising: Sponsored Tweets. http://sponsoredtweets.com
③ Newt Gingrich’s Twitter follower count under scrutiny - Faster Forward - The Washington Post. http://www.washingtonpost.com/blogs/faster-forward/post/newt-gingrichs-twitter-follower-count-under-scrutiny/2011/08/02/gIQAVQ33pI_blog.html
④ Buy, Sell, and Trade Twitter accounts. http://socialsellouts.com

或者没发过、或者是发过很少的微博。但是这些粉丝都关注大约 700 个账户。也被大约 400 个账户关注。每个账户看上去很平常，因为没有任何证据能够明确说明他们的可疑性。然而作为一组粉丝，这些账户很明显是可疑的了，因为他们有太多特征是非常相似的。在图5.24右侧，看到四个被关注的人，他们大多是很明显的垃圾信息传播者，这些账户需要很多粉丝来关注，显得非常出名。这些被关注的人包括 “SEO 专家” 或者是一些不知名的却有很多粉丝的小商家。第一个用户是@AaronMartirano 所发出来的一些碎语微博，非常可疑，内容中包括 “auto follower” 和 “follback”，并链接到被 Twitter 标记为不安全的内容。还可以看到微博用户@aaronseal，他的个人属性给出了堪萨斯州的关于 Bell Credit Union 的 GPS 信息以及微博来要求用户点击 “喜欢”。相似地，可以看到@biz2day，这是给自己描述为 “广告和 SEO 商务的网站拥有者”，并没有给出不安全的内容，但是连接到非常可疑的网页。对于@aaronseal 和@biz2day来说，这些账户很明显的购买了微博和粉丝来增加他们的价格。最后来看看账号@HousingReporter，这是一个拥有 164,700 个粉丝的租房公司拥有者，甚至超过了马萨诸塞州的议员。这个租房公司拥有想要显得更加知名和有信誉度，所以买了很多粉丝。

从上述的例子中知道不可能从粉丝自己的微博和社交关系知道单个粉丝到底是否可疑。然而用 CatchSync 是可以只利用图结构信息，就找到非常可疑的用户。实验中给出了附加信息和上下文信息来证明这些被抓住的粉丝和顾客都是非常可疑的。从而知道 CatchSync 能够非常有效地检测潜藏的可疑行为。

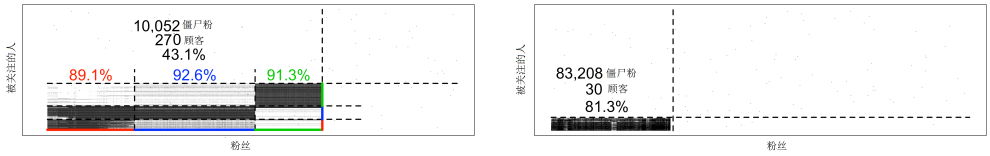
5.4.2　具有密集行为的可疑用户检测性能

本节中给出实验评测结果，首先是两大真实数据集，包括社交数据和影视数据，接着给出在仿真数据中的效果。首先在腾讯微博的上亿用户数据集上测试算法有效性。在实际应用中很难运行完全 SVD，所以使用低秩 SVD 能够更节省运行时间。在微博数据集上设置 $k = 20$。原因是在图5.9(a) 中的 U_{19} 和 U_{20} 可以看到这样的高维度子空间往往是随机分布的。本工作建议按照实际应用来选取合适的 k。由于 $k \ll N$ （N 是图中的节点数量），算法会比起完全 SVD 要快太多。表5.16给出了社交数据中所找到的密集行为：

- “密集块” 和 “阶梯状块”: 根据所提出的规则和算法，可以根据镭射线模式找到密集块，可以根据珍珠状模式找到阶梯状的三个重合的块。图5.25给出了邻接矩阵和其中的粉丝集合 F_0 以及 F_1-F_3。
- 高密度，少量 “伪装” 和少量 “伪知名”: 每一个块的密度都超过 80%，但是阶梯状块的整体密度只有 43%。这证明了阶梯状块是由部分重合的块组成的。“伪装”，也就是有密集行为的粉丝和没有密集行为的用户之间的连接模式，

	“镭射线状” F_0	“珍珠状” F_1	“珍珠状” F_2	“珍珠状” F_3	“珍珠状” 全体
种子数量	100	1,239	107	990	—
块大小	$83,208 \times 30$	$3,188 \times 135$	$7,210 \times 79$	$2,457 \times 148$	$10,052 \times 270$
密度	81.3%	91.3%	92.6%	89.1%	43.1%
伪装	0.14%	0.06%	0.10%	0.05%	0.07%
伪知名	0.05%	1.93%	1.94%	1.72%	1.73%
出度	231±109	310±7	312±7	304±5	310±7
入度	2.0±1.4	9±6	10±6	17±13	12±9

表 5.16　密集行为所构成的连接模式的统计数值：块的密度都大于 80%，另外会存在较少量的伪装和伪知名现象。



(a) 镭射线状，密集块　　(b) 珍珠状，阶梯状块

图 5.25　LockInfer 所找到的密集块的特性：(a) 83,000 余粉丝密集关注同样的 30 个偶像，形成邻接矩阵中的密集块；(b) 10,000 余粉丝关注 270 个偶像，形成三个重合的阶梯状块。

是只有 0.2% 的小密度的。伪知名也不到 2%。

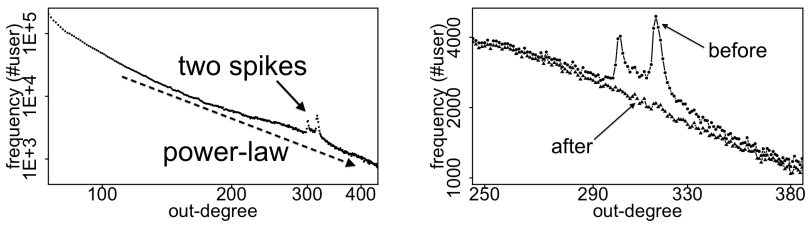
上述数字证明了不重合和部分重合的密集行为的存在性，也证实了所提出方法的有效性。另外，这里还能给出更多的证据来说明这些有密集行为的粉丝是可疑的。

- 可疑的个人信息： 检测到的 10,787 个粉丝用户的昵称过于相似，形式都是 “a#####”（# 表示数字，比如 “a27217”）。他们自定义的出生日期也非常相似，都是 1 月 1 日。他们和正常用户不同，可能是用同样的脚本生成。表5.17中给出了具体的细节和实例。
- 太小的入度，几乎没有粉丝： 检测到的密集块中的粉丝用户的平均入度只有 2.0，阶梯状块中的粉丝用户的平均入度小于 20。有秘籍行为的用户有很多关注的人，但是他们却并没有多少名声来获得粉丝。
- 过于相似的出度，过于相同数量的关注的人： 阶梯状块中的有密集行为的粉丝用户的出度大多在 300 周围。如图5.26所示，用对数 -对数画出社交图的出度分布，其中在出度为 300 的位置存在异常的出度频度。移除有密集行为的粉丝用户后，会发现尖峰消失了，出度分布变得顺滑。

大多数出度分布会形成顺滑形状，展现出常见的长尾分布特征，包括幂律分布[258,269]，指数分布等。Broder 等人发现了观察到偏离顺滑分布的现象：在英特网中会存在链接注入现象形成的尖峰[269]。那么本工作尝试移除有密集行为的用

注册昵称	出生日期	入度	出度	注册昵称	出生日期	入度	出度
a15681	1986:01:01	1	301	a27290	1980:01:01	5	107
a21154	1975:06:04	2	301	a38887	1982:01:01	3	310
a27217	1982:01:01	3	304	catty	1972:01:01	2	316

表 5.17　所找到的有密集行为的用户具有可疑的个人信息：LockInfer 所找到的具有密集行为的粉丝在注册昵称、自设定的出生日期和粉丝数量上都非常可疑，因为往往是“a#####”形式的昵称，1 月 1 日的出生日期和很小的入度。



(a) 出度分布中存在两个尖峰　　(b) LockInfer 能将分布变得顺滑

图 5.26　出度分布能够恢复成正常的模式（幂律分布）：通过移除有密集行为的粉丝用户能够移除出度分布中的尖峰。这些粉丝有相似的出度值，相近数量的关注的人。

出度	用户数量	出度	用户数量	出度	用户数量	异常现象
270	3,436	280	3,048	290	2,773	
300	3,944	**301**	**4,043**	302	3,418	尖峰
311	2,852	312	2,679	323	2,836	
315	4,373	**316**	**4,918**	317	4,414	尖峰
320	2,821	330	1,976	340	1,650	

表 5.18　出度分布中的异常现象：出度分布在 301 和 316 的出度处存在两个尖峰。本工作旨在观察异常现象，并用连接模式解释，恢复出度分布。

	“镭射线” I	“镭射线” II	“镭射线” III	“镭射线” IV
种子数量	74	57	30	103
密集块	414×30	178×44	139×15	373×12
密度	61.4%	58.2%	66.7%	57.4%
伪装	0.65%	1.74%	1.87%	0.69%
伪知名	6.06%	6.59%	12.0%	12.8%

表 5.19　IMDb 上的密集行为：LockInfer 能从 IMDb 中通过分析 4 个特征子空间的镭射线找到密集块。

户，并观察是否将出度分布变得顺滑。这也证实了 LockInfer 能找到异常。

在表5.18和图5.26(a) 中展示了正常的出度分布会符合幂律分布，而在出度为 301 和 316 的位置，存在两个尖峰。一种直接的解释就是这种粉丝节点关注同样数量用户的异常行为。图5.26(b) 给出了通过移除 LockInfer 所检测到的有密集行为的用户，可以把出度分布还原成常见的幂律分布形状。那么可知，所以移除的异常用户是具有相似的出度值，换句话说，关注同样数量的用户。

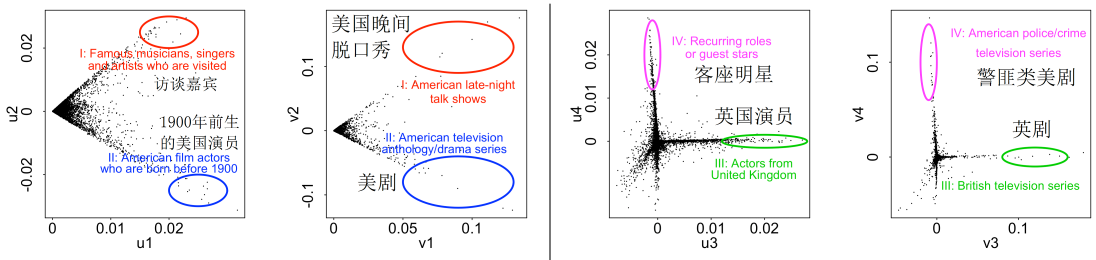
工作中还在 IMDb 的影视数据上做实验。这个数据是一个从演员到电影/电视剧的大规模二部图。在图5.27中能观察到特征子空间图里 U_1 和 U_2，以及 U_3 和 U_4 构成的镭射线。表5.19给出了 IMDb 中的连接模式。从这些镭射线找到种子节点，并继而搜寻出密集块。块的详细信息如下：

- 块 I 中包括美国谈话节目。参加的访谈嘉宾包括音乐家、歌手和电影演员。
- 块 II 中包括诸多出生在 1900 年以前的著名电影演员。他们的电影大多创作于 1950 年以前。
- 块 III 包括英国演员和他们所参与的 BBC 出品的电影。
- 块 IV 包括参与了美国侦探剧和犯罪类电影的客座明星，这些电视剧包括《CSI》,《NYPD Blue》和《Cold Case》。

另外，继续从表5.19中可以观察到：

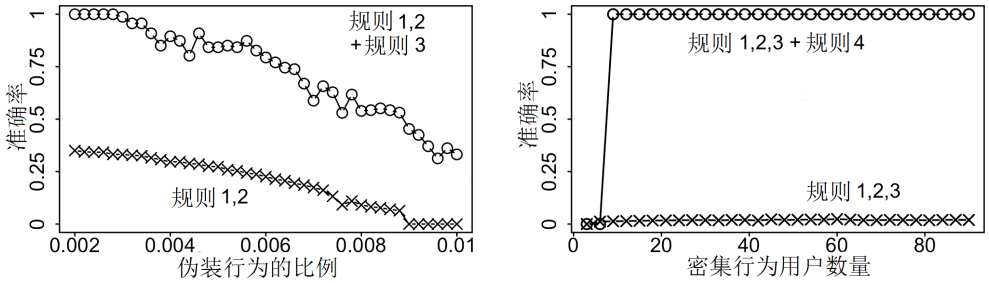
- 大规模的密集行为：从上述 4 个密集块的统计可以看出，演员的数量从 100 到 500，电影的数量从 10 到 50，数量在影视数据里比较起来很大。
- 密度高：密度从 50% 到 70% 不等，说明有超过一半的演员会加入几乎每一部电影，超过一半的电影会需要几乎所有演员参与。
- 少量 “伪装”：这些演员大多比较普通，所以加入其他电影的可能性非常小。
- 大量 “伪知名”：这些电影还是会邀请其他的演员加入来扩充队伍，所以会有大约 10% 的伪知名值。

接下来是仿真实验的结果。首先来证明规则 3（旋转 “镭射线”）和规则 4（“珍珠状”）的有效性。在 100 万节点的随机幂律图中注入一组粉丝和被关注的人。目



(a.1) U_1 和 U_2　　(a.2) V_1 和 V_2　　(b.1) U_3 和 U_4　　(b.2) V_3 和 V_4

图 5.27　IMDb 数据形成的特征子空间中的镭射线：可以看到旋转的镭射线，它们代表了在演员和电影之间密集的连接模式。



(a) 规则 3 (旋转 “镭射线”) 起作用，如果注入粉丝有 “伪装”

(b) 规则 4 (“珍珠状”) 起作用，如果注入粉丝形成 “阶梯状”

图 5.28　规则 3 和 4 的效果：准确率高意味着效果好。

标是预测被注入的是哪些节点。这里用准确率来评价算法效果。通过给粉丝添加伪装，也就是增加这些注入粉丝与非注入被关注人的关系，将密度从 0 增加到 1%。在图5.28(a) 比较了采用和不采用规则 3 来检测注入粉丝的方法，从结果中可以看到随着伪装增加时，准确率降低，很明显地，采用规则 3 会效果好很多。

注入的密集行为如果存在重合，也就是在邻接矩阵中注入阶梯状的块。通过改变阶梯状块的大小，也就是注入粉丝的数量，比较采用和不采用规则 4 的算法。规则 4 是说如果存在阶梯状块，那么特征子空间图中存在珍珠形状。图5.28(b) 中给出了考虑所有规则的算法的效果，当有密集行为的粉丝数量超过 7 的时候，密集行为丰富到足够在特征子空间中留下踪迹，算法能够检测出超过 95% 的注入粉丝，而不采用规则 4 的方法会失败。

那么传统算法在检测伪装时效果如何呢？在仿真数据上比较了 Autopart, Outrank 和 Oddball。这三种方法的准确率和召回率、以及采用与不采用规则 3 的 LockInfer 的效果在表5.20中提出。当数据中只有伪装的时候，只有采用规则 3 的 LockInfer 能够检测出这些注入粉丝。所有对比算法以及不采用规则 3 的算法都无法有效检测（准确率和召回率非常低）。LockInfer 是否能在阶梯状块的检测上能比别的方法都强？表5.21给出仿真实验结果。可以看到所提出的 LockInfer 大幅度

伪装程度	0.5%		1.0%	
	准确率	召回率	准确率	召回率
含规则 3 的 LockInfer	**0.80**	**0.78**	**0.64**	**0.47**
不含规则 3 的 LockInfer	0.34	0.32	0.10	0.08
Autopart	0.45	0.33	0.21	0.20
Outrank	0.20	0.25	0.01	0.02
Oddball	0.02	0.02	0.03	0.04

表 5.20　LockInfer 在仿真数据中存在 “伪装” 时达到最好的效果：“伪装” 并不能够逃脱规则 3（旋转镭射线）的检测。

阶梯状块的大小 (用户数量)	50		100	
	准确率	召回率	准确率	召回率
采用规则 4 的 LockInfer	**0.99**	**1.00**	**1.00**	**1.00**
不采用规则 4 的 LockInfer	0.00	0.00	0.02	0.03
Autopart	0.15	0.16	0.14	0.20
Outrank	0.24	0.19	0.32	0.32
Oddball	0.07	0.05	0.22	0.14

表 5.21　采用规则 4（珍珠状）LockInfer 能在存在阶梯状块的仿真数据上效果更好。

地提升了密集行为检测效果，尤其是和不采用规则 4 的方法相比，LockInfer 的检测效果几乎完美（99%-100%）。

5.4.3　信息操纵行为检测性能

本小节通过实验来回答下面几个问题：(1）所提出的 CrossSpot 算法是否能有效地找到可疑块？(2) CrossSpot 算法能否在真实数据集中找到可疑的行为模式？(3) CrossSpot 算法的效率如何？实验结果表明 CrossSpot 比起基线算法能够更准确、同时在计算上更快速地找到可疑块。同样 CrossSpot 在也能在操纵转发量、操纵话题热度和操纵网络包的数据集上找到很大很密集的块，并用辅助信息来证明这些密集块确实是可疑行为造成的。

实验中不仅使用了仿真数据集，还有两个大规模的、新的社交网络数据集和一个公共网络包数据集。表5.22中给出了所有数据集的汇总信息。

仿真数据： 采用 Erdös-Rényi-Poisson 模型生成高维度数据，生成的数据可以是大小为 $N_1 \times \cdots \times N_K$、事件数为 C 的 K 维度张量数据，并在张量中注入 b 个密集块。每一个块都定义为一个大小为 $n_1 \times \cdots \times n_K$、事件数为 c 的密集块。如果一个被注入的密集块只在某一个维度集合 $\mathcal{I}$ 中，可以设置对于 $i \notin \mathcal{I}$，有 $n_i = N_i$。

数据集	维度 #1	维度 #2	维度 #3	维度 # 4	事件数
操纵转发量	用户 ID	微博 ID	IP 地址	分钟	转发行为
	29,468,040	19,755,875	27,817,611	56,943	221,719,535
操纵热门话题	用户 ID	话题	IP 地址	分钟	微博行为
	81,186,369	1,580,042	47,717,882	56,943	276,944,456
操纵网络包	源 IP	目标 IP	端口号码	秒	发包行为
	2,345	2,355	6,055	3,610	230,836

表 5.22　数据统计：社交网络和网络包的多维度数据集。

- k：数据中的维度数量；
- $N_1 \times \cdots \times N_k$ 和 C：数据的大小和事件数；
- b：被注入的密集块的数量；
- $n_1 \times \cdots \times n_k$ 和 c：密集块的大小和事件数；
- k' 和 $\mathcal{I}$：密集块的维度数量和维度集合，对于 $i \notin \mathcal{I}$ 有 $n_i = N_i$。

操纵微博转发量数据集： 从中国最大的社交网络之一的腾讯微博上收集该数据集，其中包括用户 ID，微博 ID，IP 地址，时间（2011 年从 11 月 9 日到 12 月 20 日），以及转发时添加的评论内容。在微博上经常可见操纵微博转发的行为，有大量的转发微博是通过购买得来使得微博看上去流行。这会扭曲用户体验。表5.22中给出了这个数据集以及其他一些真实数据集的统计结果。

操纵话题热度的数据集： 和操纵微博转发是一样的，这里用腾讯微博中含有话题的原创微博。这个数据集中含有用户 ID，话题，IP 地址，时间和微博内容。这个数据集中因为存在恶意话题和推广话题而很有趣，有一些用户通过购买微博多次推广他们的内容使得这个话题很火爆。通过搜索密集的高维度行为，希望能够找到可疑的操纵话题的行为模式。

操纵网络包的数据集： 这是一个为了研究因特网中网络包的公开数据[272]，其中包括上千的由劳伦斯伯克利国家实验室（Lawrence Berkeley National Lab，简称 LBNL）从多个服务器上收集到的网络包。每一个网络包种都会包括源 IP，目标 IP，端口号和以秒计的时间戳。希望能在数据中找到密集结构。

基线算法： 本工作将所提出的 CrossSpot 算法和以下的基线算法作比较。所有基线算法都会分析结构化的行为信息，但分析方式不同。

- SVD 和 HOSVD (高维 SVD)[239,280] 通过张量数据的多维度构建正交空间，并用自动调整的方法选择阈值来切分分解向量[239]。
- MAF (MultiAspectForensics)[107] 能用特征值的柱状图分布来找到意味着高维度子数据的尖峰，这往往代表密集的二部子图模式。

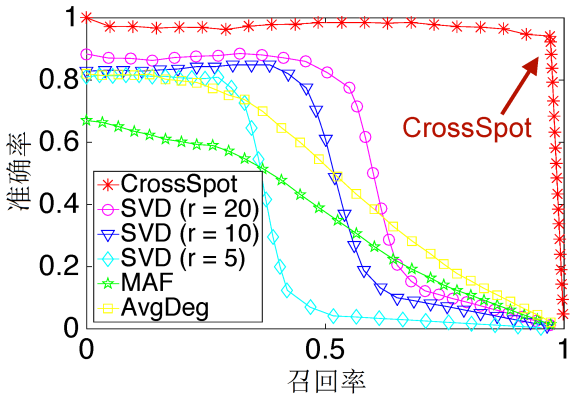


图 5.29　检测密集子图的效果：CrossSpot 在检测二维密集块的效果几近完美。

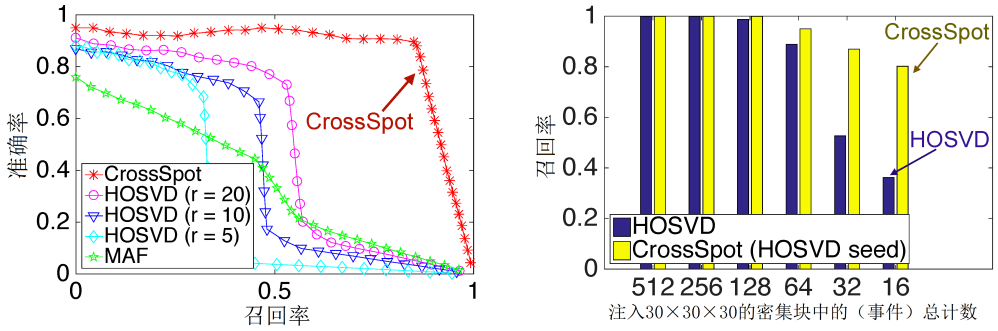
- AvgDeg (Dense Graph Components)[216] 定义用平均度数（average degree）作为衡量密集子图的指标，并且采用贪心近似算法找到图中密集的组成部分。

参数设置： 实验中为每一个方法找到最佳的表现。在运行 CrossSpot 的时候，生成 1,000 个随机种子密集块来寻找最终的密集块。其中随机决定种子模块的维度和每个维度的值集合。这里用 Python 来实现 CrossSpot。对于 SVD 和 HOSVD 的算法来说，设置不同的分解特征数量，如 5、10 和 20，并做比较。通过从 0 到 1 对所有奇异向量修改阈值，对每一个维度将高于阈值的值放入该维度的集合中，由此得到密集块。对于其他基线算法，工作中采用它们标准的实现方法。实验中在 2.40GHz×8 Intel Xeon CPU、64GB 内存、运行 Windows Server 2008-64 位系统的机器上实验。

评价方法： 采用标准的信息检索评价方法（如准确率、召回率和 F1 值等）衡量对正常行为和可疑行为做分类的检测方法有效程度[115]。准确率是检测为可疑且确实可疑的行为占检测为可疑行为的百分比，召回率是检测为可疑且确实可疑的行为占确实可疑行为的百分比。F1 值是准确率和召回率的几何平均数。

仿真实验中测试了 CrossSpot 算法，总而言之，CrossSpot 是有效的：能够从二维数据中检测密集子图，从 k 维的张量数据中检测密集 k 维密集块，和从 k 维张量数据中检测 k' 维的密集块（$k' < k$）。都能取得非常高的准确率和召回率。CrossSpot 方法是高效的：比起复杂的传统方法来说，执行时间要更短，运行更快。实验中测试了在仿真数据中检测代表可疑行为的大的、高维度密集块的以上三个任务，以及随机种子的鲁棒性。

检测密集子图（二维密集块）： 基于 ERP 模型依照下述参数生成随机矩阵：(1) 维度数量为 k=2, (2) 数据大小为 $N_1 = 1000$ 和 $N_2 = 1000$, (3) 数据中的事件数量为 $C = 10{,}000$。在随机数据中注入 $b = 6$ 个维度为 $k' = 2$ 的密集块，所以 $\mathcal{I} = \{1, 2\}$。每



(a) 检测效果　　(b) 运用 HOSVD 方法作为种子的召回率

图 5.30　检测密集块的效果：CrossSpot 算法检测三维密集块的效果比起基线算法要高，能够直接提升最好的 HOSVD 的召回率。

一个密集块的大小都是 30×30，而且密集块的事件数为 $c \in \{16, 32, 64, 128, 256, 512\}$。任务是从整个数据中把事件分为可疑的（注入的）和正常的两类。图5.29给出了检测密集子图任务下所提出的 CrossSpot 和其他基线算法的分类效果。观察到

- CrossSpot 几乎达到完美的准确率：CrossSpot 只把会增加 Suspiciousness 值的事件检测出来，因为这些事件是属于密集块的。同时，这个方法可以给出几乎完美的召回率：局部搜索不会错过密集块中任何一个值。CrossSpot 最高的 F1 值是 0.967，而 SVD, MAF, AvgDeg 这些方法的 F1 值分别为 0.634，0.439 和 0.511。MAF 会检测大量相似的元素，所以能够找到比较大的密集块，忽略小的非常密集的块，而 AvgDeg 只会检测密度非常高的块，会忽略很大的，或是不太稠密的密集块。
- SVD 会在召回率很小的时候给出非常高的准确。然而，SVD 很难找到小的、不太稠密的密集块，比如大小为 30×30、事件数为 16 或是 32 的密集块。即使这些密集块都比数据的密度要高，但是依旧难以检测到。越高的分解特征数会带来越好的分类准确度。

从高维数据中找到高维密集块： 继续根据下列参数生成随机张量数据：(1) 维度数量为 k=3，(2) 数据大小为 N_1=1,000, N_2=1,000 和 N_3=1,000，以及 (3) 数据中的事件数为 C=10,000。在数据中注入 b=6 个维度数量为 k'=3 的密集块，那么 $\mathcal{I} = \{1, 2, 3\}$。每一个块的大小都是 30×30×30，并且其中的事件数为 $c \in \{16, 32, 64, 128, 256, 512\}$。任务依旧是对正常行为和可疑行为分类。表5.30(a)中给出了 CrossSpot 和基线算法的检测效果。可以从中观察到，为了检测所有的 6 个注入三维密集块，CrossSpot 比起基线算法能够取得更好的准确率和召回率。CrossSpot 取得的最佳 F1 值为 0.891，而 HOSVD 所能取得的最好的 F1 值为 0.610，CrossSpot 比 HOSVD 提升了 46.0%。如果以 HOSVD 的结果作为 CrossSpot 的种子

	召回率				整体评价效果		
	块 #1	块 #2	块 #3	块 #4	准确率	召回率	F1 值
HOSVD (r=20)	93.7%	29.5%	23.7%	21.3%	**0.983**	0.407	0.576
HOSVD (r=10)	91.3%	24.4%	18.5%	19.2%	0.972	0.317	0.478
HOSVD (r=5)	85.7%	10.0%	9.5%	11.4%	0.952	0.195	0.324
CrossSpot	**100%**	**99.9%**	**94.9%**	**95.4%**	0.978	**0.967**	**0.972**

表 5.23　CrossSpot 算法可以检测更多的低维度的密集块：CrossSpot 能够以非常高的准确度检测注入的 4 个密集块，包括 (1) 30×30×30, (2) 30×30×1,000, (3) 30×1,000×30, 和 (4) 1,000×30×30。而每一个的时间数都是 1,000。

用户 ID× 微博 ID×IP 地址 × 分钟	事件数 c	Suspiciousness
14×1×2×1,114	41,396	1,239,865
225×1×2×200	27,313	777,781
8×2×4×1,872	17,701	491,323

表 5.24　在转发微博的数据集中检测到的最可疑的密集块。

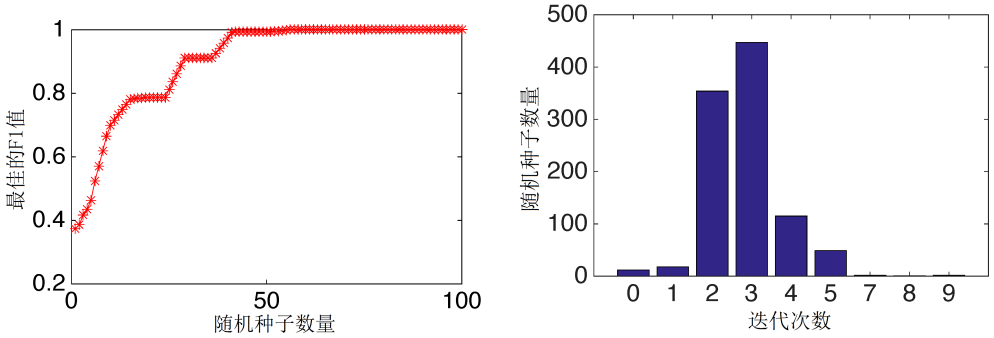
密集块，CrossSpot 所取得的最佳 F1 值为 0.979。图5.30(b) 给出了检测每一个注入的密集块的召回率。可以观察到 CrossSpot 能够比 HOSVD 提升召回率，尤其是对于较为稀疏、但还是比数据密度高的块。

从高维数据中检测低维度密集块： 用下述参数生成随机张量数据，包括 (1) 维度数量 k=3，(2) 数据大小 N_1=1,000, N_2=1,000 和 N_3=1,000，以及 (3) 数据中的事件数 C=10,000。在随机数据中注入 b=4 个密集块：

- 密集块 #1: k'_1=3，$\mathcal{I}_1$={1,2,3}。大小为 30×30×30，事件数为 c_1=512。
- 密集块 #2: k'_2=2，$\mathcal{I}_2$={1,2}. 大小为 30×30×1,000，事件数为 c_2=512。
- 密集块 #3: k'_3=2，$\mathcal{I}_3$={1,3}。大小为 30×1,000×30，事件数为 c_3=512。
- 密集块 #4: k'_4=2，$\mathcal{I}_4$={2,3}。大小为 1,000×30×30，事件数为 c_4=512。

注意到密集块 2 到密集块 4 很密集，但是只是在两个维度上，而在第三个维度上是随机分布的。表5.23给出了 CrossSpot 和基线算法的分类效果，即检测每一个注入密集块时的准确率、召回率和 F1 值。可以看到 CrossSpot 在检测三维密集块 #1 能有 100% 的召回率，而基线算法只有 85-95%。CrossSpot 能成功检测到二维密集块，而 HOSVD 很难做到，召回率很低。在 F1 值的评价标准上，CrossSpot 能够达到 0.972，得到 68.8% 的提升。

随机种子数量的鲁棒性： 这里测试如果使用更多的随机种子，CrossSpot 在检测低维度密集块效果是否有提升。图5.31(a) 给出了在不同的随机种子数量下最佳



(a) 对随机种子数量的鲁棒性　　(b) 快速收敛

图 5.31　CrossSpot 的性质：(a) 对随机种子数量的鲁棒性，即在检测 4 个注入的低维度密集块时，采用 41 个种子的时候，最佳 F1 值就能达到 1,000 个种子的效果；(b) CrossSpot 算法能够快速收敛：迭代次数的平均数为 2.87。

的 F1 值。当使用 41 个种子的时候，F1 值能够接近采用 1,000 个随机种子的结果。所以，当采用一定量的随机种子后，CrossSpot 对对随机种子数量具有鲁棒性。

运行效率分析：CrossSpot 能够在多台机器上采用不同集合的随机种子并行检测密集块。每一迭代所消耗的时间是随着高维度数据中非负元素数量呈线性关系的。图5.31(b) 给出了采用 1,000 个随机种子时的迭代次数。可以观察到 CrossSpot 通常情况下需要用 2 到 3 次迭代实现局部搜索。每一个迭代需要 5.6 秒。而如 HOSVD 和 MAF 中采用的 PARAFAC 需要更多的时间。在同一个机器上，采用分解特征数为 r=5, 10 和 20 的 HOSVD 分别能够消耗 280, 1750, 34,510 秒。表5.23和图5.31(b) 都是并没有并行化的结果，CrossSpot 需要用 230 秒达到 0.972 的 F1 值，而 HOSVD 在采用 r=5 时需要 280 秒才能达到相当小的 F1 值 0.324。

检测操纵微博转发量的评测结果：表5.24中给出了腾讯微博中转发行为里的大密集块。CrossSpot 能够找到数量很多、同时密度很高的块。可以观察到的是 (a) 14 个用户在 2 个 IP 地址、19 个小时里转发同一内容；(b) 225 个用户在 2 个 IP 地址、4 个小时里转发同一个微博大约 30,000 次。这些结果证明了有一些可疑用户的行为会造成某些微博看上去特别流行。

表5.25给出了操纵微博转发量中 CrossSpot 找到的大小为 225×1×2×200 的密集块的详细信息。一组用户（例如 A, B, C）在同样的 2 个 IP 地址上、每隔 5 分钟、同一个城市转发同样的微博 “Galaxy note 梦想计划: 开心快乐生活，走遍全世界”。可以看到的是他们的转发评论都是从一些文学、文艺书籍中摘取的。这些转发行为中的周期性和并没有道理的转发内容是证明 CrossSpot 确实有效找到可疑行为的强有力证据。

检测操纵话题热度的评测结果： 表5.26中给出了发布含话题微博中的大的密

用户 ID	时刻	IP 地址	转发评论
用户 -A	11-26 10:08:54	IP-1 (山东聊城)	不看天空的颜色只看脚下的影子
用户 -B	11-26 10:08:54	IP-1 (山东聊城)	你给我一个承诺，我给你一个结果
用户 -C	11-26 10:09:07	IP-2 (山东聊城)	想让圣洁的婚纱纪念是在同样圣洁...
用户 -A	11-26 10:13:55	IP-1 (山东聊城)	不看别人的颜色只看自己的本色
用户 -B	11-26 10:13:57	IP-2 (山东聊城)	下辈子要做一只考拉，一天睡 20 小时
用户 -C	11-26 10:14:03	IP-1 (山东聊城)	我们需要赖以生存的是有一个人来...
用户 -A	11-26 10:18:57	IP-1 (山东聊城)	记录我的点滴生活喜欢这种社交方式
用户 -C	11-26 10:19:18	IP-2 (山东聊城)	我的电脑蓝屏了
用户 -B	11-26 10:19:31	IP-1 (山东聊城)	最后我才相信，生活里并没有真完美
用户 -A	11-26 10:23:50	IP-1 (山东聊城)	兄弟不要伤心，一切都会过去
用户 -B	11-26 10:24:04	IP-2 (山东聊城)	生活就是一个个车站，我们都是过客
用户 -C	11-26 10:24:19	IP-1 (山东聊城)	我对我喜欢的人才会真的生气

表 5.25　操纵微博转发量：观察到密集地在同一组 IP 地址上（每隔 5 分钟）转发同一条微博 “Galaxy note 梦想计划: 开心快乐生活，走遍全世界”。这对应了在表5.24中给出的大小为 225×1×2×200 的密集块。

用户 ID× 话题 ×IP 地址 × 分钟	事件数 c	Suspiciousness
582×3×294×**56,940**	5,941,821	111,799,948
188×1×313×**56,943**	2,344,614	47,013,868
75×1×2×2,061	689,179	19,378,403

表 5.26　从操纵话题热度数据中找到的密集块。

用户 ID	时刻	IP 地址	带话题的微博内容
用户 -D	11-18 12:12:51	IP-1 (山东德阳)	# 雪之约定 # GALAXY SII QQ 定制服务...
用户 -E	11-18 12:12:53	IP-1 (山东德阳)	# 雪之约定 # GALAXY SII QQ 定制服务...
用户 -F	11-18 12:12:54	IP-2 (山东枣庄)	# 雪之约定 # GALAXY SII QQ 定制服务...
用户 -E	11-18 12:17:55	IP-1 (山东德阳)	# 李宁英雄装备 # 支持好活动!
用户 -F	11-18 12:17:56	IP-2 (山东枣庄)	# 李宁英雄装备 # 支持好活动!
用户 -D	11-18 12:18:40	IP-1 (山东德阳)	# 东芝明亮达人 # 颜色个性测试来看看...
用户 -E	11-18 17:00:31	IP-2 (山东枣庄)	# 雪之约定 # GALAXY SII QQ 定制服务...
用户 -D	11-18 17:00:49	IP-2 (山东枣庄)	# 东芝明亮达人 # 颜色个性测试来看看...
用户 -F	11-18 17:00:56	IP-2 (山东枣庄)	# 李宁英雄装备 # 支持好活动!

表 5.27　话题操纵行为：可以观察到一组用户在同样的一组 IP 地址上不停的发布带有某些话题的微博。这些行为对应表5.26中给出的大小为 582×3×294×56,940 的密集块。

源 IP 地址 × 目标 IP 地址 × 端口号 × 秒	事件数 c	Suspiciousness
411×9×6×**3,610**	47,449	552,465
533×6×1×**3,610**	30,476	400,391
5×5×2×**3,610**	18,881	317,529
11×7×7×**3,610**	20,382	295,869

表 5.28　在 LBNL 的网络访问数据中找到大规模的密集块。这些可疑块几乎都是占据了所有的时间，说明可疑的网络访问行为在这一小时里是持续发生的。

集块。CrossSpot 给出了事件数很多、密度很大的密集块。可以看到 (1) 持续性的攻击：582 个用户为 3 个话题、在 294 个 IP 地址上、在 43 天里的几乎每一分钟，总共接近 6,000,000 个微博；(2) 密集性的攻击：75 个用户为同一个话题、在 2 个 IP 上、35 个小时里，发布了接近 700,000 个微博。CrossSpot 所发现的前两个大的密集块几乎占据了时间维度的所有值，也就是说，不考虑任何一个时间值都会降低可疑程度。

表5.27给出了 CrossSpot 所找到的操纵话题行为形成的大小为 582×3×294×56,940 的密集块。一组用户 (例如 D 、E 和 F) 在多个 IP 地址、同一省份的两个城市发布广告性质话题的微博（例如 # 雪之约定 #、# 李宁英雄装备 # 和 # 东芝明亮达人 #）。这证明了 CrossSpot 能够找到虚假流行的广告性质话题。

检测操纵网络包的评测结果： 表5.28中给出了 LBNL 网络包数据集中的大的密集块模式。CrossSpot 能够找到大规模、密度高的密集块。可以看到 (1) 非常大和密集的块：411 个源 IP 地址向 9 个目标 IP 地址的 6 个端口上发送总共有 47,449 次访问，以及 533 个源 IP 地址向 6 个目标 IP 地址、同一个端口上发送 30,476 个网络包；(2) 小的、但是非常密集的块：5 个源 IP 地址向 5 个目标 IP 地址在 2 个端口上发送 18,881 个网络包，或是 11 个源 IP 地址向 7 个目标 IP、7 个不同的端口号上发送 20,382 个网络包。要注意的是这些最可疑的密集块占据了时间维度的每一个值，说明存在一组源 IP 地址不停地向多个目标机器、同一组端口上、一小时里的每分每秒发送大量的网络包。

5.5　本章小结

本章节提出一种新颖的可疑行为检测方法 CatchSync，依靠同步性和正常性两种行为特征，自动从大规模图中区分可疑节点。CatchSync 能够找到同步性行为，并且找到可疑的源节点 -目标节点组合。其算法复杂度与数据中的边数呈线性关系。同时 CatchSync 很容易实现，无须设定参数，无论是密度还是组的数量和

大小；更无需附加信息，只需要拓扑信息。CatchSync 算法不需要标注节点或是节点属性，但融入这些附加信息，可以提升算法效果。工作中在仿真和真实数据集上都给出实验效果，验证了 CatchSync 算法能够找到过去算法无法抓到的同步性行为模式。

第二，本章节提出了基于大规模图中连接模式检测密集行为的方法 LockInfer。工作中给出了对特征子空间图的新认识：镭射线状和珍珠状模式是由不同类型（不重合和重合）的密集行为形成的。本文给出了检测密集行为模式的快速算法，并证实了算法在真实数据集和仿真实验中的有效性。

第三，在本节率先提出了在任意的维度数量上衡量密集块的可疑程度的评价指标。主要的动机是检测欺诈行为，尝试回答可疑行为检测中的一个根本性问题，即给定两个或多个密集块，哪一个或者哪一些更值得人注意的问题。贡献点包括：(1) 提出了一系列能够衡量可疑行为形成的密集块的要求（公理）；(2) 提出了基于基本的、概率模型得到的可疑程度评价指标，并且证明该指标符合所有公理；(3) 提出了高维数据中寻找可疑的密集块的可扩展算法；(4) 证实了我们算法既有在仿真实验，也有在高达 3 亿事件的真实数据上的有效性。CrossSpot 能够稳定提升 F1 值，最多达到 68.8% 的增幅。

第 6 章　总结与展望

本文研究了社交媒体中复杂用户行为的分析与建模。在这一章，对全文进行总结，并对未来的研究工作进行展望。

6.1　研究工作总结

在社交媒体迅猛发展与融合的背景下，社交媒体用户行为预测与检测的目标在于提供高质量的推荐系统、个性化搜索、市场营销、反欺诈等服务。而行为的分析与建模是预测和检测技术的基础。本文结合社交媒体用户行为上下文关联性、跨域跨平台性和真伪性，解决高稀疏度、多元异构性和意图复杂性等问题。

首先，本文提出了基于社交上下文和时空上下文的采纳信息行为分析模型：为解决采纳信息行为的高稀疏性问题，本文提出融合信息采纳、信息内容、社交关系和用户交互等社交上下文，提升行为预测效果的方案。结合社交媒体信息传播和采纳机制，本文挖掘出个人兴趣偏好和人与人之间社交影响力的两大社交上下文因素在采纳信息行为的并发作用，提出基于社交上下文的融合行为模型。实验表明，该上下文关联的融合模型显著好于基于单一因素的模型。另一方面，复杂的社交媒体环境营造出丰富的时空上下文，使得用户行为具有明显的多面性和动态性特征。本文提出用辅助信息作为灵活约束项、用高维张量刻画行为的空间多面性、用张量序列描述行为的时间维度动态性的进化分析方法。对大规模数据的实验表明，该模型方法能显著提升行为预测效果。并且本文提出了实现快速增量数据处理的近似算法。

第二，本文提出了社交媒体跨域行为和跨平台行为的迁移学习算法。社交媒体有着复杂需求的用户在多域和多平台中得到满足，例如在微博平台中，用户需要转发微博来表达自己的兴趣，需要编辑社交标签来描述自我特征，需要加入社交群组进行互动。这些用户也会注册兴趣类的评分网站，收集自己喜欢的电影、音乐和书籍。为解决单一域或单一平台的行为稀疏性以及冷启动用户（即新注册的用户）的问题，本文在单一平台利用社交域桥接单一平台内的多个内容域，重构社交媒体平台为围绕社交域的星状图，给出迁移学习思想的随机漫步算法。本文进一步地提出利用重合用户特征桥接多个社交媒体平台，迁移辅助平台的行为信息，大幅提升目标平台的预测效果。实验表明，该算法在跨域行为、跨平台行为和冷启动用户行为预测中的效果比既有算法有显著提高。

最后，本文提出基于同步性和密集性的可疑行为分析方法和评价指标。欺诈、

垃圾传播、僵尸粉关注等可疑行为严重威胁社交媒体安全、降低用户体验。本文抓住可疑行为的同步性和密集性特征，提出快速有效的分析方法，成功从多个社交媒体数据中检测出可疑行为（如僵尸粉和信息操纵行为等），并还原被扭曲的统计分布（如幂律的出度分布等）。该方法优于基于内容的传统方法，并能互为补充。本文进一步基于概率论提出量化跨维度异常行为可疑程度的新颖指标，并给出快速检测算法。实验表明，该算法能有效地检测高维真实社交媒体数据中的信息操纵行为。

6.2　研究工作展望

本文研究了社交媒体复杂用户行为的分析与建模方法。作者认为，未来需要进一步研究的内容包括：

1. 社交媒体用户行为复杂特性和潜在规律深度分析。本文阐述了用户行为的上下文关联性、跨域跨平台性和真伪性三大特性，分析了采纳信息行为和欺诈等可疑行为的潜在规律。然而，社交媒体种类多样，应用设置各异，环境复杂，所以用户行为会有更独特的特性，其产生的潜在规律会随着环境变化而变化。如何进一步挖掘用户行为的复杂特性，分析潜在的行为产生规律，并据此提升用户行为的预测和检测技术在实际应用中的效果，对于行为分析和建模问题具有深远意义。
2. 基于用户行为的社交应用解决方案的模型表示。本文着眼于实现个体行为的预测和检测，即便采用以群体表示个体的方法，也仍然面临个体行为随机性大、猜不准的本质问题。对于个体行为的充分了解和精准预测对于实际社交应用，尤其是涉及媒体的政府决策和社会法则来说，并没有直接的指导意义。实际的社交应用需要的往往是如何调整能够提升用户量、信息质量和网站流量。政府和社会所需要的往往是一项简明扼要、易于实施的解决方案。所以如何以行为数据作为输入，社交应用通用的解决方案作为输出是非常有挑战性的。这要求后续的工作做好基于繁杂行为数据的应用中存在问题对应解决方案的模型表示。
3. 基于行为数据的预测模型与基于专家知识的分析技术相融合。两者都能为实际应用提出解决方案，然而，基于行为数据的预测模型（“机脑”）往往只能反映问题和现象，缺乏解决问题的智能，目前的数据技术受限于模型表示，只能依据数据形式而关注个体，不是可落实的解决方案或政策；基于专家知识的分析技术（“人脑”）太过主观，缺乏大规模数据的支撑，难以了解事实真相。所以只有融合数据技术和专家知识得出的解决方案才会是合理有

效的。人脑更容易给出方案框架，机脑更擅长分析方案可行程度，人脑从可行程度分析结果明确如何调整解决方案，机脑能够进一步确认方案的可行程度。但是这个任务是非常有挑战性的：需要正确的理论表示方法、实用的数据技术和便捷的群智机制。如何突破当下“机脑与人脑竞争”的趋势，形成机脑与人脑互补的新体系是非常重要的。

参考文献

[1] Balabanović M, Shoham Y. Fab: content-based, collaborative recommendation. Communications of the ACM, 1997, 40(3):66–72.

[2] Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms. Proceedings of the 10th international conference on World Wide Web. ACM, 2001. 285–295.

[3] Karypis G. Evaluation of item-based top-n recommendation algorithms. Proceedings of the tenth international conference on Information and knowledge management. ACM, 2001. 247–254.

[4] Si L, Jin R. Flexible mixture model for collaborative filtering. ICML, volume 3, 2003. 704–711.

[5] Si L, Jin R. Unified filtering by combining collaborative filtering and content-based filtering via mixture model and exponential model. Proceedings of the thirteenth ACM international conference on Information and knowledge management. ACM, 2004. 156–157.

[6] Deshpande M, Karypis G. Item-based top-n recommendation algorithms. ACM Transactions on Information Systems (TOIS), 2004, 22(1):143–177.

[7] Herlocker J L, Konstan J A, Terveen L G, et al. Evaluating collaborative filtering recommender systems. ACM Transactions on Information Systems (TOIS), 2004, 22(1):5–53.

[8] Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. Knowledge and Data Engineering, IEEE Transactions on, 2005, 17(6):734–749.

[9] Gori M, Pucci A, Roma V, et al. Itemrank: A random-walk based scoring algorithm for recommender engines. IJCAI, volume 7, 2007. 2766–2771.

[10] Koren Y. Factorization meets the neighborhood: a multifaceted collaborative filtering model. Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008. 426–434.

[11] Koren Y, Bell R, Volinsky C. Matrix factorization techniques for recommender systems. Computer, 2009, (8):30–37.

[12] Harvey M, Carman M J, Ruthven I, et al. Bayesian latent variable models for collaborative item rating prediction. Proceedings of the 20th ACM international conference on Information and knowledge management. ACM, 2011. 699–708.

[13] Agarwal D, Gurevich M. Fast top-k retrieval for model based recommendation. Proceedings of the fifth ACM international conference on Web search and data mining. ACM, 2012. 483–492.

[14] Zhang Z, Zhao K, Zha H. Inducible regularization for low-rank matrix factorizations for collaborative filtering. Neurocomputing, 2012, 97:52–62.

[15] Fan C, Lan Y, Guo J, et al. Collaborative factorization for recommender systems. Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. ACM, 2013. 949–953.

[16] Shi Y, Larson M, Hanjalic A. Collaborative filtering beyond the user-item matrix: A survey of the state of the art and future challenges. ACM Computing Surveys (CSUR), 2014, 47(1):3.

[17] Ma H, Yang H, Lyu M R, et al. Sorec: social recommendation using probabilistic matrix factorization. Proceedings of the 17th ACM conference on Information and knowledge management. ACM, 2008. 931–940.

[18] Konstas I, Stathopoulos V, Jose J M. On social networks and collaborative recommendation. Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009. 195–202.

[19] Chen Y, Canny J F. Recommending ephemeral items at web scale. Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. ACM, 2011. 1013–1022.

[20] Ma H, King I, Lyu M R. Learning to recommend with explicit and implicit social relations. ACM Transactions on Intelligent Systems and Technology (TIST), 2011, 2(3):29.

[21] Ma H, Zhou D, Liu C, et al. Recommender systems with social regularization. Proceedings of the fourth ACM international conference on Web search and data mining. ACM, 2011. 287–296.

[22] Shi Y, Serdyukov P, Hanjalic A, et al. Personalized landmark recommendation based on geotags from photo sharing sites. 2011..

[23] Noel J, Sanner S, Tran K N, et al. New objective functions for social collaborative filtering. Proceedings of the 21st international conference on World Wide Web. ACM, 2012. 859–868.

[24] Shen Y, Jin R, Dou D, et al. Socialized gaussian process model for human behavior prediction in a health social network. ICDM, volume 12. Citeseer, 2012. 1110–1115.

[25] Zhu X, Guo J, Cheng X, et al. More than relevance: high utility query recommendation by mining users' search behaviors. Proceedings of the 21st ACM international conference on Information and knowledge management. ACM, 2012. 1814–1818.

[26] Liu X, Aberer K. Soco: a social network aided context-aware recommender system. Proceedings of the 22nd international conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2013. 781–802.

[27] Sedhain S, Sanner S, Xie L, et al. Social affinity filtering: Recommendation through fine-grained analysis of user interactions and activities. Proceedings of the first ACM conference on Online social networks. ACM, 2013. 51–62.

[28] Tang J, Hu X, Gao H, et al. Exploiting local and global social context for recommendation. Proceedings of the Twenty-Third international joint conference on Artificial Intelligence. AAAI Press, 2013. 2712–2718.

[29] Tang J, Hu X, Liu H. Social recommendation: a review. Social Network Analysis and Mining, 2013, 3(4):1113–1133.

[30] Qian X, Feng H, Zhao G, et al. Personalized recommendation combining user interest and social circle. Knowledge and Data Engineering, IEEE Transactions on, 2014, 26(7):1763–1777.

[31] Sedhain S, Sanner S, Braziunas D, et al. Social collaborative filtering for cold-start recommendations. Proceedings of the 8th ACM Conference on Recommender systems. ACM, 2014. 345–348.

[32] Blei D M, Ng A Y, Jordan M I. Latent dirichlet allocation. the Journal of machine Learning research, 2003, 3:993–1022.

[33] Liu N N, Zhao M, Yang Q. Probabilistic latent preference analysis for collaborative filtering. Proceedings of the 18th ACM conference on Information and knowledge management. ACM, 2009. 759–766.

[34] Phelan O, McCarthy K, Smyth B. Using twitter to recommend real-time topical news. Proceedings of the third ACM conference on Recommender systems. ACM, 2009. 385–388.

[35] Liu L, Zhu F, Jiang M, et al. Mining diversity on social media networks. Multimedia Tools and Applications, 2012, 56(1):179–205.

[36] Sanderson M, Paramita M L, Clough P, et al. Do user preferences and evaluation measures line up? Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2010. 555–562.

[37] Stefanidis K, Pitoura E, Vassiliadis P. Managing contextual preferences. Information Systems, 2011, 36(8):1158–1180.

[38] Liu Q, Chen E, Xiong H, et al. Enhancing collaborative filtering by user interest expansion via personalized ranking. Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on, 2012, 42(1):218–233.

[39] Zhu H, Chen E, Yu K, et al. Mining personal context-aware preferences for mobile users. ICDM, volume 12, 2012. 1212–1217.

[40] Narang K, Nagar S, Mehta S, et al. Discovery and analysis of evolving topical social discussions on unstructured microblogs. Advances in Information Retrieval. Springer, 2013: 545–556.

[41] Benjamin L S. Structural analysis of social behavior. Psychological review, 1974, 81(5):392.

[42] Bond R, Smith P B. Culture and conformity: A meta-analysis of studies using asch's (1952b, 1956) line judgment task. Psychological bulletin, 1996, 119(1):111.

[43] Bandura A, Bryant J. Social cognitive theory of mass communication. Media effects: Advances in theory and research, 2002, 2:121–153.

[44] Leskovec J, Singh A, Kleinberg J. Patterns of influence in a recommendation network. Advances in Knowledge Discovery and Data Mining. Springer, 2006: 380–389.

[45] Liu L, Tang J, Han J, et al. Mining topic-level influence in heterogeneous networks. Proceedings of the 19th ACM international conference on Information and knowledge management. ACM, 2010. 199–208.

[46] Goyal A, Bonchi F, Lakshmanan L V. Learning influence probabilities in social networks. Proceedings of the third ACM international conference on Web search and data mining. ACM, 2010. 241–250.

[47] Huang J, Cheng X, Guo J, et al. Social recommendation with interpersonal influence. ECAI, volume 10, 2010. 601–606.

[48] Cui P, Wang F, Yang S, et al. Item-level social influence prediction with probabilistic hybrid factor matrix factorization. Twenty-Fifth AAAI Conference on Artificial Intelligence, 2011.

[49] Cui P, Wang F, Liu S, et al. Who should share what?: item-level social influence prediction for users and posts ranking. Proceedings of the 34th international ACM SIGIR conference on Research and development in Information Retrieval. ACM, 2011. 185–194.

[50] Yang S H, Long B, Smola A, et al. Like like alike: joint friendship and interest propagation in social networks. Proceedings of the 20th international conference on World wide web. ACM, 2011. 537–546.

[51] Yang X, Steck H, Liu Y. Circle-based recommendation in online social networks. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012. 1267–1275.

[52] Huang J, Cheng X Q, Shen H W, et al. Exploring social influence via posterior effect of word-of-mouth recommendations. Proceedings of the fifth ACM international conference on Web search and data mining. ACM, 2012. 573–582.

[53] Chua F C T, Lauw H W, Lim E P. Generative models for item adoptions using social correlation. Knowledge and Data Engineering, IEEE Transactions on, 2013, 25(9):2036–2048.

[54] Cui P, Jin S, Yu L, et al. Cascading outbreak prediction in networks: a data-driven approach. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 901–909.

[55] Cheng S, Shen H, Huang J, et al. Imrank: influence maximization via finding self-consistent ranking. Proceedings of the 37th international ACM SIGIR conference on Research & development in information retrieval. ACM, 2014. 475–484.

[56] Massa P, Avesani P. Trust-aware recommender systems. Proceedings of the 2007 ACM conference on Recommender systems. ACM, 2007. 17–24.

[57] Jamali M, Ester M. Trustwalker: a random walk model for combining trust-based and item-based recommendation. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009. 397–406.

[58] Moghaddam S, Jamali M, Ester M, et al. Feedbacktrust: using feedback effects in trust-based recommendation systems. Proceedings of the third ACM conference on Recommender systems. ACM, 2009. 269–272.

[59] Carminati B, Ferrari E, Girardi J. Trust and share: Trusted information sharing in online social networks. Data Engineering (ICDE), 2012 IEEE 28th International Conference on. IEEE, 2012. 1281–1284.

[60] Rachlin H. The value of temporal patterns in behavior. Current Directions in Psychological Science, 1995. 188–192.

[61] Kleinberg J. Temporal dynamics of on-line information streams, 2006.

[62] Sun J, Tao D, Faloutsos C. Beyond streams and graphs: dynamic tensor analysis. Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2006. 374–383.

[63] Lin Y R, Chi Y, Zhu S, et al. Facetnet: a framework for analyzing communities and their evolutions in dynamic networks. Proceedings of the 17th international conference on World Wide Web. ACM, 2008. 685–694.

[64] Koren Y. Collaborative filtering with temporal dynamics. Communications of the ACM, 2010, 53(4):89–97.

[65] Kumar R, Novak J, Tomkins A. Structure and evolution of online social networks. Link mining: models, algorithms, and applications. Springer, 2010: 337–357.

[66] Lathia N, Hailes S, Capra L, et al. Temporal diversity in recommender systems. Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2010. 210–217.

[67] Xiang L, Yuan Q, Zhao S, et al. Temporal recommendation on graphs via long-and short-term preference fusion. Proceedings of the 16th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2010. 723–732.

[68] Dunlavy D M, Kolda T G, Acar E. Temporal link prediction using matrix and tensor factorizations. ACM Transactions on Knowledge Discovery from Data (TKDD), 2011, 5(2):10.

[69] Yang J, Leskovec J. Patterns of temporal variation in online media. Proceedings of the fourth ACM international conference on Web search and data mining. ACM, 2011. 177–186.

[70] Radinsky K, Svore K, Dumais S, et al. Modeling and predicting behavioral dynamics on the web. Proceedings of the 21st international conference on World Wide Web. ACM, 2012. 599–608.

[71] Rossi R, Neville J. Time-evolving relational classification and ensemble methods. Advances in Knowledge Discovery and Data Mining. Springer, 2012: 1–13.

[72] Yu K, Ding W, Wang H, et al. Bridging causal relevance and pattern discriminability: Mining emerging patterns from high-dimensional data. Knowledge and Data Engineering, IEEE Transactions on, 2013, 25(12):2721–2739.

[73] Chen W, Hsu W, Lee M L. Modeling user's receptiveness over time for recommendation. Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. ACM, 2013. 373–382.

[74] Radinsky K, Bennett P N. Predicting content change on the web. Proceedings of the sixth ACM international conference on Web search and data mining. ACM, 2013. 415–424.

[75] Radinsky K, Svore K M, Dumais S T, et al. Behavioral dynamics on the web: Learning, modeling, and prediction. ACM Transactions on Information Systems (TOIS), 2013, 31(3):16.

[76] Sun Y, Tang J, Han J, et al. Co-evolution of multi-typed objects in dynamic star networks. 2013..

[77] Wang X, Zhai C, Roth D. Understanding evolution of research themes: a probabilistic generative model for citations. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 1115–1123.

[78] Yuan Q, Cong G, Ma Z, et al. Who, where, when and what: discover spatio-temporal topics for twitter users. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 605–613.

[79] Yuan Q, Cong G, Ma Z, et al. Time-aware point-of-interest recommendation. Proceedings of the 36th international ACM SIGIR conference on Research and development in information retrieval. ACM, 2013. 363–372.

[80] Zheng X, Ding H, Mamitsuka H, et al. Collaborative matrix factorization with multiple similarities for predicting drug-target interactions. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 1025–1033.

[81] Zhong E, Fan W, Zhu Y, et al. Modeling the dynamics of composite social networks. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 937–945.

[82] Zhong E, Xiang E W, Fan W, et al. Friendship prediction in composite social networks. arXiv preprint arXiv:1402.4033, 2014..

[83] Zhong E, Fan W, Yang Q. User behavior learning and transfer in composite social networks. ACM Transactions on Knowledge Discovery from Data (TKDD), 2014, 8(1):6.

[84] Wu X, Zhu X, Wu G Q, et al. Data mining with big data. Knowledge and Data Engineering, IEEE Transactions on, 2014, 26(1):97–107.

[85] Tipping M E, Bishop C M. Probabilistic principal component analysis. Journal of the Royal Statistical Society: Series B (Statistical Methodology), 1999, 61(3):611–622.

[86] Mnih A, Salakhutdinov R. Probabilistic matrix factorization. Advances in neural information processing systems, 2007. 1257–1264.

[87] Stewart G W, Sun J g, Jovanovich H B. Matrix perturbation theory, volume 175. Academic press New York, 1990.

[88] Lee D D, Seung H S. Algorithms for non-negative matrix factorization. Advances in neural information processing systems, 2001. 556–562.

[89] Wang H, Banerjee A, Boley D. Common component analysis for multiple covariance matrices. Proceedings of the 17th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2011. 956–964.

[90] Wang F, Tong H, Lin C Y. Towards evolutionary nonnegative matrix factorization. AAAI, volume 11, 2011. 501–506.

[91] Kolda T G. Orthogonal tensor decompositions. SIAM Journal on Matrix Analysis and Applications, 2001, 23(1):243–255.

[92] Singh A P, Gordon G J. Relational learning via collective matrix factorization. Proceedings of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2008. 650–658.

[93] Cai J F, Candès E J, Shen Z. A singular value thresholding algorithm for matrix completion. SIAM Journal on Optimization, 2010, 20(4):1956–1982.

[94] Sun J T, Zeng H J, Liu H, et al. Cubesvd: a novel approach to personalized web search. Proceedings of the 14th international conference on World Wide Web. ACM, 2005. 382–390.

[95] Cichocki A, Zdunek R. Regularized alternating least squares algorithms for non-negative matrix/tensor factorization. Advances in Neural Networks–ISNN 2007. Springer, 2007: 793–802.

[96] Bader B W, Harshman R A, Kolda T G. Temporal analysis of semantic graphs using asalsan. Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on. IEEE, 2007. 33–42.

[97] Bader B W, Kolda T G. Efficient matlab computations with sparse and factored tensors. SIAM Journal on Scientific Computing, 2007, 30(1):205–231.

[98] Bader B W, Kolda T G. Matlab tensor toolbox version 2.2. Albuquerque, NM, USA: Sandia National Laboratories, 2007..

[99] Huang H, Ding C, Luo D, et al. Simultaneous tensor subspace selection and clustering: the equivalence of high order svd and k-means clustering. Proceedings of the 14th ACM SIGKDD international conference on Knowledge Discovery and Data mining. ACM, 2008. 327–335.

[100] Kolda T G, Sun J. Scalable tensor decompositions for multi-aspect data mining. Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on. IEEE, 2008. 363–372.

[101] Sun J, Tao D, Papadimitriou S, et al. Incremental tensor analysis: Theory and applications. ACM Transactions on Knowledge Discovery from Data (TKDD), 2008, 2(3):11.

[102] Symeonidis P, Nanopoulos A, Manolopoulos Y. Tag recommendations based on tensor dimensionality reduction. Proceedings of the 2008 ACM conference on Recommender systems. ACM, 2008. 43–50.

[103] Kolda T G, Bader B W. Tensor decompositions and applications. SIAM review, 2009, 51(3):455–500.

[104] Grasedyck L. Hierarchical singular value decomposition of tensors. SIAM Journal on Matrix Analysis and Applications, 2010, 31(4):2029–2054.

[105] Rendle S, Schmidt-Thieme L. Pairwise interaction tensor factorization for personalized tag recommendation. Proceedings of the third ACM international conference on Web search and data mining. ACM, 2010. 81–90.

[106] Acar E, Dunlavy D M, Kolda T G, et al. Scalable tensor factorizations for incomplete data. Chemometrics and Intelligent Laboratory Systems, 2011, 106(1):41–56.

[107] Maruhashi K, Guo F, Faloutsos C. Multiaspectforensics: Pattern mining on large-scale heterogeneous networks with tensor analysis. Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on. IEEE, 2011. 203–210.

[108] Baskaran M, Meister B, Vasilache N, et al. Efficient and scalable computations with sparse tensors. High Performance Extreme Computing (HPEC), 2012 IEEE Conference on. IEEE, 2012. 1–6.

[109] Kang U, Papalexakis E, Harpale A, et al. Gigatensor: scaling tensor analysis up by 100 times-algorithms and discoveries. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012. 316–324.

[110] Wang M, Li H, Tao D, et al. Multimodal graph-based reranking for web image search. Image Processing, IEEE Transactions on, 2012, 21(11):4649–4661.

[111] Burges C, Shaked T, Renshaw E, et al. Learning to rank using gradient descent. Proceedings of the 22nd international conference on Machine learning. ACM, 2005. 89–96.

[112] Ng A Y, Zheng A X, Jordan M I. Link analysis, eigenvectors and stability. International Joint Conference on Artificial Intelligence, volume 17. LAWRENCE ERLBAUM ASSOCIATES LTD, 2001. 903–910.

[113] Wang M, Yang K, Hua X S, et al. Towards a relevant and diverse search of social images. Multimedia, IEEE Transactions on, 2010, 12(8):829–842.

[114] Ou M, Cui P, Wang F, et al. Comparing apples to oranges: a scalable solution with heterogeneous hashing. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 230–238.

[115] Yang Y. An evaluation of statistical approaches to text categorization. Information retrieval, 1999, 1(1-2):69–90.

[116] Davis J, Goadrich M. The relationship between precision-recall and roc curves. Proceedings of the 23rd international conference on Machine learning. ACM, 2006. 233–240.

[117] Fawcett T. An introduction to roc analysis. Pattern recognition letters, 2006, 27(8):861–874.

[118] Salganik M J, Dodds P S, Watts D J. Experimental study of inequality and unpredictability in an artificial cultural market. science, 2006, 311(5762):854–856.

[119] Gretton A, Borgwardt K M, Rasch M J, et al. A kernel two-sample test. The Journal of Machine Learning Research, 2012, 13(1):723–773.

[120] Niu S, Guo J, Lan Y, et al. Top-k learning to rank: labeling, ranking and evaluation. Proceedings of the 35th international ACM SIGIR conference on Research and development in information retrieval. ACM, 2012. 751–760.

[121] Lan Y, Niu S, Guo J, et al. Is top-k sufficient for ranking? Proceedings of the 22nd ACM international conference on Conference on information & knowledge management. ACM, 2013. 1261–1270.

[122] Berkovsky S, Kuflik T, Ricci F. Cross-domain mediation in collaborative filtering. User Modeling 2007. Springer, 2007: 355–359.

[123] Raina R, Battle A, Lee H, et al. Self-taught learning: transfer learning from unlabeled data. Proceedings of the 24th international conference on Machine learning. ACM, 2007. 759–766.

[124] Winoto P, Tang T. If you like the devil wears prada the book, will you also enjoy the devil wears prada the movie? a study of cross-domain recommendations. New Generation Computing, 2008, 26(3):209–225.

[125] Li B, Yang Q, Xue X. Can movies and books collaborate? cross-domain collaborative filtering for sparsity reduction. IJCAI, volume 9, 2009. 2052–2057.

[126] Li B, Yang Q, Xue X. Transfer learning for collaborative filtering via a rating-matrix generative model. Proceedings of the 26th Annual International Conference on Machine Learning. ACM, 2009. 617–624.

[127] Yang Q, Chen Y, Xue G R, et al. Heterogeneous transfer learning for image clustering via the social web. Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP: Volume 1-Volume 1. Association for Computational Linguistics, 2009. 1–9.

[128] Adams R P, Dahl G E, Murray I. Incorporating side information in probabilistic matrix factorization with gaussian processes. arXiv preprint arXiv:1003.4944, 2010..

[129] Cao B, Liu N N, Yang Q. Transfer learning for collective link prediction in multiple heterogenous domains. Proceedings of the 27th International Conference on Machine Learning (ICML-10), 2010. 159–166.

[130] Pan S J, Yang Q. A survey on transfer learning. Knowledge and Data Engineering, IEEE Transactions on, 2010, 22(10):1345–1359.

[131] Pan W, Xiang E W, Liu N N, et al. Transfer learning in collaborative filtering for sparsity reduction. AAAI, volume 10, 2010. 230–235.

[132] Porteous I, Asuncion A U, Welling M. Bayesian matrix factorization with side information and dirichlet process mixtures. AAAI, 2010.

[133] Zhuang F, Luo P, Xiong H, et al. Cross-domain learning from multiple sources: a consensus regularization perspective. Knowledge and Data Engineering, IEEE Transactions on, 2010, 22(12):1664–1678.

[134] Li B. Cross-domain collaborative filtering: A brief survey. Tools with Artificial Intelligence (ICTAI), 2011 23rd IEEE International Conference on. IEEE, 2011. 1085–1086.

[135] Li B, Zhu X, Li R, et al. Cross-domain collaborative filtering over time. Proceedings of the Twenty-Second international joint conference on Artificial Intelligence-Volume Volume Three. AAAI Press, 2011. 2293–2298.

[136] Pan W, Liu N N, Xiang E W, et al. Transfer learning to predict missing ratings via heterogeneous user feedbacks. IJCAI Proceedings-International Joint Conference on Artificial Intelligence, volume 22, 2011. 2318.

[137] Shi Y, Larson M, Hanjalic A. Tags as bridges between domains: Improving recommendation with tag-induced cross-domain collaborative filtering. User Modeling, Adaption and Personalization. Springer, 2011: 305–316.

[138] Zhu Y, Chen Y, Lu Z, et al. Heterogeneous transfer learning for image classification. AAAI, 2011.

[139] Moreno O, Shapira B, Rokach L, et al. Talmud: transfer learning for multiple domains. Proceedings of the 21st ACM international conference on Information and knowledge management. ACM, 2012. 425–434.

[140] Tang J, Wu S, Sun J, et al. Cross-domain collaboration recommendation. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012. 1285–1293.

[141] Zhong E, Fan W, Wang J, et al. Comsoc: adaptive transfer of user behaviors over composite social network. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012. 696–704.

[142] Zhou T, Shan H, Banerjee A, et al. Kernelized probabilistic matrix factorization: Exploiting graphs and side information. SDM, volume 12. SIAM, 2012. 403–414.

[143] Abel F, Herder E, Houben G J, et al. Cross-system user modeling and personalization on the social web. User Modeling and User-Adapted Interaction, 2013, 23(2-3):169–209.

[144] Chen W, Hsu W, Lee M L. Making recommendations from multiple domains. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 892–900.

[145] Gao S, Luo H, Chen D, et al. Cross-domain recommendation via cluster-level latent factor model. Machine Learning and Knowledge Discovery in Databases. Springer, 2013: 161–176.

[146] Hu L, Cao J, Xu G, et al. Personalized recommendation via cross-domain triadic factorization. Proceedings of the 22nd international conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2013. 595–606.

[147] Lu Z, Pan W, Xiang E W, et al. Selective transfer learning for cross domain recommendation. SDM. SIAM, 2013. 641–649.

[148] Shapira B, Rokach L, Freilikhman S. Facebook single and cross domain data for recommendation systems. User Modeling and User-Adapted Interaction, 2013, 23(2-3):211–247.

[149] Shi Y, Larson M, Hanjalic A. Exploiting social tags for cross-domain collaborative filtering. arXiv preprint arXiv:1302.4888, 2013..

[150] Yang L, Hanneke S, Carbonell J. A theory of transfer learning with applications to active learning. Machine learning, 2013, 90(2):161–189.

[151] Zhao L, Pan S J, Xiang E W, et al. Active transfer learning for cross-system recommendation. AAAI, 2013.

[152] Al-Shedivat M, Wang J J Y, Alzahrani M, et al. Supervised transfer sparse coding. Twenty-eighth AAAI conference on artificial intelligence, 2014. 1665–1672.

[153] Jing H, Liang A C, Lin S D, et al. A transfer probabilistic collective factorization model to handle sparse data in collaborative filtering. ICDM, 2014. 250–259.

[154] Li C Y, Lin S D. Matching users and items across domains to improve the recommendation quality. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014. 801–810.

[155] Tan B, Zhong E, Ng M, et al. Mixed-transfer: transfer learning over mixed graphs. Proceedings of SIAM International Conference on Data Mining, 2014.

[156] Tan B, Zhong E, Xiang E W, et al. Multi-transfer: Transfer learning with multiple views and multiple sources. Statistical Analysis and Data Mining: The ASA Data Science Journal, 2014, 7(4):282–293.

[157] Wang X, Huang T K, Schneider J. Active transfer learning under model shift. Proceedings of the 31st International Conference on Machine Learning (ICML-14), 2014. 1305–1313.

[158] Gao B, Liu T Y, Zheng X, et al. Consistent bipartite graph co-partitioning for star-structured high-order heterogeneous data co-clustering. Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, 2005. 41–50.

[159] Gao B, Liu T Y, Ma W Y. Star-structured high-order heterogeneous data co-clustering based on consistent information theory. Data Mining, 2006. ICDM'06. Sixth International Conference on. IEEE, 2006. 880–884.

[160] Tong H, Faloutsos C, Pan J Y. Fast random walk with restart and its applications. 2006..

[161] Avin C, Krishnamachari B. The power of choice in random walks: an empirical study. Computer Networks, 2008, 52(1):44–60.

[162] Safro I, Hovland P D, Shin J, et al. Improving random walk performance. CSC, 2009. 108–112.

[163] Chen Y C, Lin Y S, Shen Y C, et al. A modified random walk framework for handling negative ratings and generating explanations. ACM Transactions on Intelligent Systems and Technology (TIST), 2013, 4(1):12.

[164] Liu Y, Jin R, Yang L. Semi-supervised multi-label learning by constrained non-negative matrix factorization. Proceedings of the national conference on artificial intelligence, volume 21. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006. 421.

[165] Li T, Ding C, Jordan M I. Solving consensus and semi-supervised clustering problems using nonnegative matrix factorization. Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on. IEEE, 2007. 577–582.

[166] Wang F, Li T, Zhang C. Semi-supervised clustering via matrix factorization. SDM. SIAM, 2008. 1–12.

[167] Lee H, Yoo J, Choi S. Semi-supervised nonnegative matrix factorization. Signal Processing Letters, IEEE, 2010, 17(1):4–7.

[168] Tax D M, Duin R P. Outlier detection using classifier instability. Advances in Pattern Recognition. Springer, 1998: 593–601.

[169] Shekhar S, Lu C T, Zhang P. Detecting graph-based spatial outliers: algorithms and applications (a summary of results). Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2001. 371–376.

[170] Noble C C, Cook D J. Graph-based anomaly detection. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2003. 631–636.

[171] Wong W K, Moore A, Cooper G, et al. Bayesian network anomaly pattern detection for disease outbreaks. ICML, 2003. 808–815.

[172] Gyöngyi Z, Garcia-Molina H, Pedersen J. Combating web spam with trustrank. Proceedings of the Thirtieth international conference on Very large data bases-Volume 30. VLDB Endowment, 2004. 576–587.

[173] Idé T, Kashima H. Eigenspace-based anomaly detection in computer systems. Proceedings of the tenth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2004. 440–449.

[174] Liu C, Yan X, Yu H, et al. Mining behavior graphs for" backtrace" of noncrashing bugs. SDM. SIAM, 2005. 286–297.

[175] Chau D H, Pandit S, Faloutsos C. Detecting fraudulent personalities in networks of online auctioneers. Knowledge Discovery in Databases: PKDD 2006. Springer, 2006: 103–114.

[176] Pandit S, Chau D H, Wang S, et al. Netprobe: a fast and scalable system for fraud detection in online auction networks. Proceedings of the 16th international conference on World Wide Web. ACM, 2007. 201–210.

[177] Willems C, Holz T, Freiling F. Toward automated dynamic malware analysis using cwsandbox. IEEE Security & Privacy, 2007, (2):32–39.

[178] Jindal N, Liu B. Opinion spam and analysis. Proceedings of the 2008 International Conference on Web Search and Data Mining. ACM, 2008. 219–230.

[179] Benevenuto F, Rodrigues T, Almeida V, et al. Detecting spammers and content promoters in online video social networks. Proceedings of the 32nd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2009. 620–627.

[180] Lee K, Caverlee J, Webb S. Uncovering social spammers: social honeypots+ machine learning. Proceedings of the 33rd international ACM SIGIR conference on Research and development in information retrieval. ACM, 2010. 435–442.

[181] Heard N A, Weston D J, Platanioti K, et al. Bayesian anomaly detection methods for social networks. The Annals of Applied Statistics, 2010, 4(2):645–662.

[182] Stringhini G, Kruegel C, Vigna G. Detecting spammers on social networks. Proceedings of the 26th Annual Computer Security Applications Conference. ACM, 2010. 1–9.

[183] Perez C, Lemercier M, Birregah B, et al. Spot 1.0: Scoring suspicious profiles on twitter. Advances in Social Networks Analysis and Mining (ASONAM), 2011 International Conference on. IEEE, 2011. 377–381.

[184] Xiong L, Póczos B, Schneider J G, et al. Hierarchical probabilistic models for group anomaly detection. International Conference on Artificial Intelligence and Statistics, 2011. 789–797.

[185] Cao Q, Sirivianos M, Yang X, et al. Aiding the detection of fake accounts in large scale social online services. Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation. USENIX Association, 2012. 15–15.

[186] Hu J, Wang F, Sun J, et al. A healthcare utilization analysis framework for hot spotting and contextual anomaly detection. AMIA Annual Symposium Proceedings, volume 2012. American Medical Informatics Association, 2012. 360.

[187] Mukherjee A, Liu B, Glance N. Spotting fake reviewer groups in consumer reviews. Proceedings of the 21st international conference on World Wide Web. ACM, 2012. 191–200.

[188] Song Y, Cao L, Wu X, et al. Coupled behavior analysis for capturing coupling relationships in group-based market manipulations. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012. 976–984.

[189] Akoglu L, Chandy R, Faloutsos C. Opinion fraud detection in online reviews by network effects. ICWSM, 2013.

[190] Beutel A, Xu W, Guruswami V, et al. Copycatch: stopping group attacks by spotting lockstep behavior in social networks. Proceedings of the 22nd international conference on World Wide Web. International World Wide Web Conferences Steering Committee, 2013. 119–130.

[191] Hu X, Tang J, Zhang Y, et al. Social spammer detection in microblogging. Proceedings of the Twenty-Third international joint conference on Artificial Intelligence. AAAI Press, 2013. 2633–2639.

[192] Aggarwal A, Kumaraguru P. Followers or phantoms? an anatomy of purchased twitter followers. arXiv preprint arXiv:1408.1534, 2014..

[193] De Cristofaro E, Friedman A, Jourjon G, et al. Paying for likes?: Understanding facebook like fraud using honeypots. Proceedings of the 2014 Conference on Internet Measurement Conference. ACM, 2014. 129–136.

[194] Mao H H, Wu C J, Papalexakis E E, et al. Malspot: Multi2 malicious network behavior patterns analysis. Advances in Knowledge Discovery and Data Mining. Springer, 2014: 1–14.

[195] Shah N, Beutel A, Gallagher B, et al. Spotting suspicious link behavior with fbox: An adversarial perspective. arXiv preprint arXiv:1410.3915, 2014..

[196] Yu R, He X, Liu Y. Glad: group anomaly detection in social media analysis. Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2014. 372–381.

[197] Karypis G, Kumar V. Metis-unstructured graph partitioning and sparse matrix ordering system, version 2.0. 1995..

[198] Karypis G, Kumar V. A fast and high quality multilevel scheme for partitioning irregular graphs. SIAM Journal on scientific Computing, 1998, 20(1):359–392.

[199] Ding C H, He X, Zha H. A spectral method to separate disconnected and nearly-disconnected web graph components. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2001. 275–280.

[200] Chakrabarti D. Autopart: Parameter-free graph partitioning and outlier detection. Knowledge Discovery in Databases: PKDD 2004. Springer, 2004: 112–124.

[201] Cook D J, Holder L B. Mining graph data. John Wiley & Sons, 2006.

[202] Dhillon I S, Guan Y, Kulis B. Weighted graph cuts without eigenvectors a multilevel approach. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2007, 29(11):1944–1957.

[203] Eberle W, Holder L. Discovering structural anomalies in graph-based data. Data Mining Workshops, 2007. ICDM Workshops 2007. Seventh IEEE International Conference on. IEEE, 2007. 393–398.

[204] Moonesinghe H, Tan P N. Outrank: a graph-based outlier detection framework using random walk. International Journal on Artificial Intelligence Tools, 2008, 17(01):19–36.

[205] Akoglu L, McGlohon M, Faloutsos C. Oddball: Spotting anomalies in weighted graphs. Advances in Knowledge Discovery and Data Mining. Springer, 2010: 410–421.

[206] Aggarwal C C, Wang H. Managing and mining graph data, volume 40. Springer, 2010.

[207] Feng J, He X, Hubig N, et al. Compression-based graph mining exploiting structure primitives. Data Mining (ICDM), 2013 IEEE 13th International Conference on. IEEE, 2013. 181–190.

[208] Akoglu L, Tong H, Koutra D. Graph based anomaly detection and description: a survey. Data Mining and Knowledge Discovery, 2014. 1–63.

[209] Koutra D, Kang U, Vreeken J, et al. Vog: Summarizing and understanding large graphs. arXiv preprint arXiv:1406.3411, 2014..

[210] Dhillon I S. Co-clustering documents and words using bipartite spectral graph partitioning. Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2001. 269–274.

[211] Zha H, He X, Ding C, et al. Bipartite graph partitioning and data clustering. Proceedings of the tenth international conference on Information and knowledge management. ACM, 2001. 25–32.

[212] Sun J, Qu H, Chakrabarti D, et al. Neighborhood formation and anomaly detection in bipartite graphs. Data Mining, Fifth IEEE International Conference on. IEEE, 2005. 8–pp.

[213] Feng J, He X, Konte B, et al. Summarization-based mining bipartite graphs. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012. 1249–1257.

[214] Fang Y, Wang R, Dai B, et al. Graph-based learning via auto-grouped sparse regularization and kernelized extension. 2015..

[215] Asahiro Y, Iwama K, Tamaki H, et al. Greedily finding a dense subgraph. Journal of Algorithms, 2000, 34(2):203–221.

[216] Charikar M. Greedy approximation algorithms for finding dense components in a graph. Approximation Algorithms for Combinatorial Optimization. Springer, 2000: 84–95.

[217] Yan X, Han J. Closegraph: mining closed frequent graph patterns. Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2003. 286–295.

[218] Pei J, Jiang D, Zhang A. On mining cross-graph quasi-cliques. Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining. ACM, 2005. 228–238.

[219] Jiang D, Pei J. Mining frequent cross-graph quasi-cliques. ACM Transactions on Knowledge Discovery from Data (TKDD), 2009, 2(4):16.

[220] Andersen R. A local algorithm for finding dense subgraphs. ACM Transactions on Algorithms (TALG), 2010, 6(4):60.

[221] Lahiri M, Berger-Wolf T Y. Periodic subgraph mining in dynamic networks. Knowledge and Information Systems, 2010, 24(3):467–497.

[222] Lee V E, Ruan N, Jin R, et al. A survey of algorithms for dense subgraph discovery. Managing and Mining Graph Data. Springer, 2010: 303–336.

[223] Miller B, Bliss N, Wolfe P J. Subgraph detection using eigenvector l1 norms. Advances in Neural Information Processing Systems, 2010. 1633–1641.

[224] Zou Z, Li J, Gao H, et al. Mining frequent subgraph patterns from uncertain graph data. Knowledge and Data Engineering, IEEE Transactions on, 2010, 22(9):1203–1218.

[225] Giatsidis C, Thilikos D M, Vazirgiannis M. D-cores: Measuring collaboration of directed graphs based on degeneracy. Data Mining (ICDM), 2011 IEEE 11th International Conference on. IEEE, 2011. 201–210.

[226] Bahmani B, Kumar R, Vassilvitskii S. Densest subgraph in streaming and mapreduce. Proceedings of the VLDB Endowment, 2012, 5(5):454–465.

[227] Chen J, Saad Y. Dense subgraph extraction with application to community detection. Knowledge and Data Engineering, IEEE Transactions on, 2012, 24(7):1216–1230.

[228] Tsourakakis C, Bonchi F, Gionis A, et al. Denser than the densest subgraph: Extracting optimal quasi-cliques with quality guarantees. Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2013. 104–112.

[229] Balalau O D, Bonchi F, Chan T, et al. Finding subgraphs with maximum total density and limited overlap. Proceedings of the Eighth ACM International Conference on Web Search and Data Mining. ACM, 2015. 379–388.

[230] Clauset A, Newman M E, Moore C. Finding community structure in very large networks. Physical review E, 2004, 70(6):066111.

[231] Newman M E. Finding community structure in networks using the eigenvectors of matrices. Physical review E, 2006, 74(3):036104.

[232] Brown J, Broderick A J, Lee N. Word of mouth communication within online communities: Conceptualizing the online social network. Journal of interactive marketing, 2007, 21(3):2–20.

[233] Chi Y, Zhu S, Song X, et al. Structural and temporal analysis of the blogosphere through community factorization. Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2007. 163–172.

[234] Rosvall M, Bergstrom C T. An information-theoretic framework for resolving community structure in complex networks. Proceedings of the National Academy of Sciences, 2007, 104(18):7327–7331.

[235] Wakita K, Tsurumi T. Finding community structure in mega-scale social networks:[extended abstract]. Proceedings of the 16th international conference on World Wide Web. ACM, 2007. 1275–1276.

[236] Leskovec J, Lang K J, Dasgupta A, et al. Statistical properties of community structure in large social and information networks. Proceedings of the 17th international conference on World Wide Web. ACM, 2008. 695–704.

[237] Satuluri V, Parthasarathy S. Scalable graph clustering using stochastic flows: applications to community discovery. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009. 737–746.

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

[239] Prakash B A, Sridharan A, Seshadri M, et al. Eigenspokes: Surprising patterns and scalable community chipping in large graphs. Advances in Knowledge Discovery and Data Mining. Springer, 2010: 435–448.

[240] Wu L, Ying X, Wu X, et al. Line orthogonality in adjacency eigenspace with application to community partition. IJCAI Proceedings-International Joint Conference on Artificial Intelligence, volume 22, 2011. 2349.

[241] Ng A Y, Jordan M I, Weiss Y, et al. On spectral clustering: Analysis and an algorithm. Advances in neural information processing systems, 2002, 2:849–856.

[242] Wang J, Zeng H, Chen Z, et al. Recom: reinforcement clustering of multi-type interrelated data objects. Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval. ACM, 2003. 274–281.

[243] Schaeffer S E. Graph clustering. Computer Science Review, 2007, 1(1):27–64.

[244] Zhou D, Orshanskiy S A, Zha H, et al. Co-ranking authors and documents in a heterogeneous network. Data Mining, 2007. ICDM 2007. Seventh IEEE International Conference on. IEEE, 2007. 739–744.

[245] Chen C, Yan X, Zhu F, et al. Graph olap: a multi-dimensional framework for graph data analysis. Knowledge and Information Systems, 2009, 21(1):41–63.

[246] Fu Q, Banerjee A. Bayesian overlapping subspace clustering. Data Mining, 2009. ICDM'09. Ninth IEEE International Conference on. IEEE, 2009. 776–781.

[247] Huang L, Yan D, Taft N, et al. Spectral clustering with perturbed data. Advances in Neural Information Processing Systems, 2009. 705–712.

[248] Kriegel H P, Kröger P, Zimek A. Clustering high-dimensional data: A survey on subspace clustering, pattern-based clustering, and correlation clustering. ACM Transactions on Knowledge Discovery from Data (TKDD), 2009, 3(1):1.

[249] Müller E, Günnemann S, Assent I, et al. Evaluating clustering in subspace projections of high dimensional data. Proceedings of the VLDB Endowment, 2009, 2(1):1270–1281.

[250] Yan D, Huang L, Jordan M I. Fast approximate spectral clustering. Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2009. 907–916.

[251] Gunnemann S, Farber I, Boden B, et al. Subspace clustering meets dense subgraph mining: A synthesis of two paradigms. Data Mining (ICDM), 2010 IEEE 10th International Conference on. IEEE, 2010. 845–850.

[252] Trappey C V, Trappey A J, Wu C Y. Clustering patents using non-exhaustive overlaps. Journal of Systems Science and Systems Engineering, 2010, 19(2):162–181.

[253] Wauthier F L, Jojic N, Jordan M I. Active spectral clustering via iterative uncertainty reduction. Proceedings of the 18th ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, 2012. 1339–1347.

[254] Shen H W, Cheng X Q. Spectral methods for the detection of network community structure: a comparative analysis. Journal of Statistical Mechanics: Theory and Experiment, 2010, 2010(10):P10020.

[255] Ren F X, Shen H W, Cheng X Q. Modeling the clustering in citation networks. Physica A: Statistical Mechanics and its Applications, 2012, 391(12):3533–3539.

[256] Günnemann S, Boden B, Färber I, et al. Efficient mining of combined subspace and subgraph clusters in graphs with feature vectors. Advances in Knowledge Discovery and Data Mining. Springer, 2013: 261–275.

[257] ERDdS P, R&WI A. On random graphs i. Publ. Math. Debrecen, 1959, 6:290–297.

[258] Faloutsos M, Faloutsos P, Faloutsos C. On power-law relationships of the internet topology. ACM SIGCOMM Computer Communication Review, volume 29. ACM, 1999. 251–262.

[259] Aiello W, Chung F, Lu L. A random graph model for power law graphs. Experimental Mathematics, 2001, 10(1):53–66.

[260] Chung F, Lu L. The average distances in random graphs with given expected degrees. Proceedings of the National Academy of Sciences, 2002, 99(25):15879–15882.

[261] Albert R, Barabási A L. Statistical mechanics of complex networks. Reviews of modern physics, 2002, 74(1):47.

[262] Li J, Li H, Wong L, et al. Minimum description length principle: Generators are preferable to closed patterns. PROCEEDINGS OF THE NATIONAL CONFERENCE ON ARTIFICIAL INTELLIGENCE, volume 21. Menlo Park, CA; Cambridge, MA; London; AAAI Press; MIT Press; 1999, 2006. 409.

[263] Robins G, Pattison P, Kalish Y, et al. An introduction to exponential random graph (p*) models for social networks. Social networks, 2007, 29(2):173–191.

[264] Ying X, Wu X. On randomness measures for social networks. SDM, volume 9. SIAM, 2009. 709–720.

[265] Leskovec J, Chakrabarti D, Kleinberg J, et al. Kronecker graphs: An approach to modeling networks. The Journal of Machine Learning Research, 2010, 11:985–1042.

[266] Roman S. A problem of zarankiewicz. Journal of Combinatorial Theory, Series A, 1975, 18(2):187–198.

[267] Füredi Z. An upper bound on zarankiewicz'problem. Combinatorics, Probability and Computing, 1996, 5(01):29–33.

[268] Babai L, Guiduli B. Spectral extrema for graphs: the zarankiewicz problem. the electronic journal of combinatorics, 2009, 16(1):R123.

[269] Broder A, Kumar R, Maghoul F, et al. Graph structure in the web. Computer networks, 2000, 33(1):309–320.

[270] Hall B H, Jaffe A B, Trajtenberg M. The nber patent citation data file: Lessons, insights and methodological tools. Technical report, National Bureau of Economic Research, 2001.

[271] Tax D M, Duin R P. Support vector data description. Machine learning, 2004, 54(1):45–66.

[272] Pang R, Allman M, Bennett M, et al. A first look at modern enterprise traffic. Proceedings of the 5th ACM SIGCOMM conference on Internet Measurement. USENIX Association, 2005. 2–2.

[273] Price M, Norris D M. Health care fraud: physicians as white collar criminals? Journal of the American Academy of Psychiatry and the Law Online, 2009, 37(3):286–289.

[274] Becker R A, Volinsky C, Wilks A R. Fraud detection in telecommunications: History and lessons learned. Technometrics, 2010, 52(1).

[275] Kwak H, Lee C, Park H, et al. What is twitter, a social network or a news media? Proceedings of the 19th international conference on World wide web. ACM, 2010. 591–600.

[276] Aggarwal C C. An introduction to social network data analytics. Springer, 2011.

[277] Roy S B, De Cock M, Mandava V, et al. The microsoft academic search dataset and kdd cup 2013. Proceedings of the 2013 KDD cup 2013 workshop. ACM, 2013. 1.

[278] Brownrigg D. The weighted median filter. Communications of the ACM, 1984, 27(8):807–818.

[279] Kalman D. A singularly valuable decomposition: the svd of a matrix. The college mathematics journal, 1996, 27(1):2–23.

[280] De Lathauwer L, De Moor B, Vandewalle J. A multilinear singular value decomposition. SIAM journal on Matrix Analysis and Applications, 2000, 21(4):1253–1278.

[281] Timmons C. My favorite application using eigenvalues. 2013..

[282] Page L, Brin S, Motwani R, et al. The pagerank citation ranking: Bringing order to the web. 1999..

[283] Kleinberg J M. Authoritative sources in a hyperlinked environment. Journal of the ACM (JACM), 1999, 46(5):604–632.

[284] Chakrabarti S. Mining the Web: Discovering knowledge from hypertext data. Morgan Kaufmann, 2003.

[285] Chapelle O, Metlzer D, Zhang Y, et al. Expected reciprocal rank for graded relevance. Proceedings of the 18th ACM conference on Information and knowledge management. ACM, 2009. 621–630.

[286] Rissanen J. Modeling by shortest data description. Automatica, 1978, 14(5):465–471.

[287] Cook D J, Holder L B. Substructure discovery using minimum description length and background knowledge. Journal of Artificial Intelligence Research, 1994. 231–255.

[288] Barron A, Rissanen J, Yu B. The minimum description length principle in coding and modeling. Information Theory, IEEE Transactions on, 1998, 44(6):2743–2760.

[289] Hido S, Tsuboi Y, Kashima H, et al. Inlier-based outlier detection via direct density ratio estimation. Data Mining, 2008. ICDM'08. Eighth IEEE International Conference on. IEEE, 2008. 223–232.

[290] Sugiyama M, Kawanabe M, Chui P L. Dimensionality reduction for density ratio estimation in high-dimensional spaces. Neural Networks, 2010, 23(1):44–59.

[291] Hido S, Tsuboi Y, Kashima H, et al. Statistical outlier detection using direct density ratio estimation. Knowledge and information systems, 2011, 26(2):309–336.

[292] Chandola V, Banerjee A, Kumar V. Anomaly detection: A survey. ACM Computing Surveys (CSUR), 2009, 41(3):15.

[293] Song L, Teo C H, Smola A J. Relative novelty detection. International Conference on Artificial Intelligence and Statistics, 2009. 536–543.

[294] Sotiris V A, Tse P W, Pecht M G. Anomaly detection through a bayesian support vector machine. Reliability, IEEE Transactions on, 2010, 59(2):277–286.

[295] Muandet K, Schölkopf B. One-class support measure machines for group anomaly detection. arXiv preprint arXiv:1303.0309, 2013..

[296] Kong D, Ding C, Huang H. Robust nonnegative matrix factorization using l21-norm. Proceedings of the 20th ACM international conference on Information and knowledge management. ACM, 2011. 673–682.

致 谢

衷心感谢导师杨士强教授对我博士期间的悉心指导。从思想到生活，从项目工作到个人发展，杨士强教授都始终给予我温暖的关怀和有力的支持。他乐观开朗的人生态度、平易近人的人格魅力和因材施教的育人方法使我在做人、做事和做学问方面受益匪浅。

感谢实验室崔鹏老师对我课题选择、论文工作的精心指导。从拓展学术视野到提升学术能力，从严谨治学、精益求精的治学态度到紧扣逻辑、形象生动的表达方法，他都给予我耐心且强有力的教导。五年荏苒，他亦师亦友，始终是我学习的榜样，也必将积极影响我今后的学习和工作。

感谢实验室朱文武教授、孙立峰教授在研究工作中给予的指导和帮助。感谢 Christos Faloutsos 教授在我访问美国卡内基梅隆大学时给予的指导。感谢康涅狄格大学的王飞副教授给予我指导和帮助。感谢刘璐在我科研初期的引路和提携。感谢张康、章俊、董謜、赵宇等同学的鼓励和陪伴！感谢实验室课题组同学的热情帮助和支持！

在此，还特别感谢我论文工作中的其他合作作者，他们是香港科技大学的杨强教授，美国卡内基梅隆大学的 Alexandar Beutel，美国华盛顿大学的刘睿，清华大学的许欣然，谢谢你们无数次的讨论和鼓励！

本研究工作受到国家 973 计划项目、国家自然科学基金项目、“清华 –腾讯”联合实验室的资助和数据支持，特此致谢!

感谢我的爱人陈燕然和亲人朋友在我博士阶段给予的支持、鼓励和关怀。

声　明

本人郑重声明：所呈交的学位论文，是本人在导师指导下，独立进行研究工作所取得的成果。尽我所知，除文中已经注明引用的内容外，本学位论文的研究成果不包含任何他人享有著作权的内容。对本论文所涉及的研究工作做出贡献的其他个人和集体，均已在文中以明确方式标明。

签　名：_____ 日　期：_____

个人简历、在学期间发表的学术论文与研究成果

个人简历

1989 年 3 月 9 日出生于江苏省连云港市。

2006 年 8 月考入清华大学计算机科学与技术系计算机科学与技术专业，2010 年 7 月本科毕业并获得工学学士学位。

2010 年 8 月免试进入清华大学计算机科学与技术系攻读博士学位至今。

攻读博士学位期间的获奖情况

- **最佳论文最终列表**，ACM SIGKDD，2014（前 9 名，大陆地区首次）
- **国家奖学金**，清华大学，2014（前 2%）
- **搜狐研发奖学金**，清华大学，2013

参与的科研项目

- 国家 973 计划：“网络可视媒体的有效搜索与服务”，2011CB302206
- 自然科学基金重大国际（地区）合作研究项目：“社会化多媒体计算理论与关键技术研究”，61210008
- 自然科学基金面上项目：“基于社交访问行为与传播特性的在线视频内容部署与传输方法研究”，61272231
- 国家科技重大专项：“大型网络应用及服务平台方案设计及示范”，2012ZX01039001-003
- 科技部国际科技合作项目：“基于社交网络中媒体内容的品牌监测合作研究”，2013DFG12870
- 自然科学基金面上项目：“跨域异构媒体信息的社会化推荐关键技术研究”，61370022
- 自然科学基金青年科学基金：“网络信息感知的视频语义分析与检索”，61303075

发表的学术论文

期刊论文

[1] **Meng Jiang**, Peng Cui, Xumin Chen, Fei Wang, Wenwu Zhu and Shiqiang Yang. "Social Recommendation with Cross-Domain Transferable Knowledge." In IEEE Transactions on Knowledge and Data Engineering (TKDE), 2015. （SCI 收录，影响因子 1.815，CCF-A 类，2015 年 5 月接收)

[2] **Meng Jiang**, Peng Cui, Alex Beutel, Christos Faloutsos and Shiqiang Yang. "Catching Synchronized Behaviors in Large Networks: A Graph Mining Approach." In ACM Transactions on Knowledge Discovery from Data (TKDD), 2015. （SCI 收录，影响因子 1.147，CCF-B 类，2015 年 3 月接收)

[3] **Meng Jiang**, Peng Cui, Fei Wang, Wenwu Zhu and Shiqiang Yang. "Scalable Recommendation with Social Contextual Information." In IEEE Transactions on Knowledge and Data Engineering (TKDE), vol. 26, no. 11, pp. 2789–2802, November 2014. （SCI 收录，影响因子 1.815，CCF-A 类，索引号：WOS:000343607500015)

[4] Lu Liu, Feida Zhu, **Meng Jiang**, Jiawei Han, Lifeng Sun, Shiqiang Yang. "Mining Diversity on Social Media Networks." In Multimedia Tools and Applications (MTA), vol. 56, no. 1, pp. 179–205, January 2012. （SCI 收录，影响因子 1.058，CCF-C 类，索引号：INSPEC:12510156)

会议论文

[5] **Meng Jiang**, Peng Cui, Alex Beutel, Christos Faloutsos and Shiqiang Yang. "CatchSync: Catching Synchronized Behavior in Large Directed Graphs." In Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD), pp. 941–950, New York City, NY, US, August 24–August 27, 2014.（**最佳论文最终列表**，EI 收录，CCF-A 类，索引号：20143818172976）

[6] **Meng Jiang**, Peng Cui, Fei Wang, Xinran Xu, Wenwu Zhu and Shiqiang Yang. "FEMA: Flexible Evolutionary Multi-faceted Analysis for Dynamic Behavioral Pattern Discovery." In Proceedings of ACM SIGKDD International Conference on Knowledge Discovery and Data Mining (SIGKDD), pp. 1186–1195, New York City, NY, US, August 24–August 27, 2014. （EI 收录，CCF-A 类，索引号：20143818173000）

[7] **Meng Jiang**, Peng Cui, Alex Beutel, Christos Faloutsos and Shiqiang Yang. "Inferring Strange Behavior from Connectivity Pattern in Social Networks." In Proceedings of Pacific-Asia Conference, Advances in Knowledge Discovery and Data Mining (PAKDD), pp. 126–138, Tainan, Taiwan, May 13–May 16, 2014. （EI 收录，CCF-C 类，索引号：20142217765735）

[8] **Meng Jiang**, Peng Cui, Alex Beutel, Christos Faloutsos and Shiqiang Yang. "Detecting Suspicious Following Behavior in Multimillion-Node Social Networks." In Proceedings of International Conference on World Wide Web Companion (WWW), pp. 305–306, Seoul, Korea, April 7–April 11, 2014. （CCF-B 类，poster）

[9] **Meng Jiang**, Peng Cui, Rui Liu, Qiang Yang, Fei Wang, Wenwu Zhu and Shiqiang Yang. "Social Contextual Recommendation." In Proceedings of ACM International Conference on Information and Knowledge Management (CIKM), pp. 45–54, Maui, HI, US, October 29–November 2, 2012. （EI 收录，CCF-B 类，索引号：20125115816682）

[10] **Meng Jiang**, Peng Cui, Fei Wang, Qiang Yang, Wenwu Zhu and Shiqiang Yang. "Social Recommendation across Multiple Relational Domains." In Proceedings of ACM International Conference on Information and Knowledge Management (CIKM), pp. 1422–1431, Maui, HI, US, October 29–November 2, 2012. （EI 收录，CCF-B 类，索引号：20125115816819）

[11] Lu Liu, Jie Tang, Jiawei Han, **Meng Jiang**, Shiqiang Yang. "Mining Topic-Level Influence in Heterogeneous Networks." In Proceedings of ACM International Conference on Information and Knowledge Management (CIKM), pp. 199–208, Toronto, Canada, October 26–October 30, 2010. （EI 收录，CCF-B 类，索引号：20110313581682）