计算机研究与发展
Journal of Computer Research and Development
DOI:10.7544/issn1000-1239.2020.20190275
57(12):2556-2570,2020

基于超图的 EBSN 个性化推荐及优化算法

于亚新　张文超　李振国　李　莹
(东北大学计算机科学与工程学院　沈阳　110169)
(医学影像智能计算教育部重点实验室(东北大学)　沈阳　110169)
(yuyx@mail.neu.edu.cn)

Hypergraph-Based Personalized Recommendation & Optimization Algorithm in EBSN

Yu Yaxin, Zhang Wenchao, Li Zhenguo, and Li Ying
(*School of Computer Science and Engineering, Northeastern University, Shenyang* 110169)
(*Key Laboratory of Intelligent Computing in Medical Image (Northeastern University), Ministry of Education, Shenyang* 110169)

Abstract　The service of personalized recommendations in event-based social networks (EBSN) is a very significant and valuable issue. Most of existing research work are mainly based on the ordinary graph to model relationships in EBSN. However, EBSN is a heterogeneous and complex network with many different types of entities. Because of that, modeling EBSN with ordinary graphs has the problem of high-dimensional information loss, resulting in reduced recommendation quality. Based on this background, in this paper, we first propose a hypergraph-based personalized recommendation (PRH) algorithm in EBSN. The basic idea is to make use of the characteristics of hypergraphs without losing high-dimensional data information to model high-dimensional complex social relationship data in EBSN more accurately, and to use regularized calculation of manifold ordering to obtain preliminary recommendation results. Next, this paper proposes an optimized PRH (oPRH) algorithm from the perspective of improving the query vector setting method and applying diverse weights to all sorts of different types of super edges to further optimize the recommendation results obtained by the PRH algorithm, so as to achieve accurate recommendation. The extended experiments show that the hypergraph-based personalized recommendation algorithm in EBSN and its optimization algorithm have higher accuracy than the previous ordinary graph-based recommendation algorithms.

Key words　event-based social networks (EBSN); hypergraph; manifold ranking; regularization computation; accurate personalized recommendation; optimization

摘　要　基于事件的社交网(event-based social networks, EBSN)中的个性化推荐服务是一个十分重要且颇具应用价值的问题,现有研究工作主要基于普通图来对 EBSN 中的关系进行建模,但由于 EBSN 是一种异构型复杂社交网络,具有多种不同类型实体,因而用普通图建模 EBSN 会存在高维信息丢失问题,导致推荐质量降低.基于此,首先提出一种基于超图模型的 EBSN 个性化推荐(hypergraph-based personalized recommendation in EBSN, PRH)算法,其基本思想在于利用超图具有不丢失高维数据信息

收稿日期:2019-05-10;**修回日期**:2020-01-10
基金项目:国家自然科学基金项目(61871106,61973059);国家重点研发计划项目(2016YFC0101500)
This work was supported by the National Natural Science Foundation of China (61871106, 61973059) and the National Key Research and Development Program of China (2016YFC0101500).

之特点来更准确地对 EBSN 中复杂社交关系数据进行高维建模,并利用流形排序正则化计算获取初步推荐结果.其次,又分别从查询向量设置方式改进和对不同类超边施以不同权重等角度,提出了优化的 PRH(optimized PRH, oPRH)算法以进一步优化 PRH 算法所获推荐结果,从而实现精准推荐.扩展实验表明,基于超图的 EBSN 个性化推荐及其优化算法,推荐结果相比于以前基于普通图的推荐算法具有更高准确性.

关键词 基于事件的社交网;超图;流形排序;正则化运算;精准个性化推荐;优化

中图法分类号 TP311

近年来,随着 Web2.0 时代和 O2O(online to offline)销售模式的不断发展,出现了一种新型社交网络,即基于事件的社交网(event-based social networks, EBSN)[1],借助该应用平台,用户可以创建、发布和组织社交事件,比如组织学术会议、举行正式聚会、募集抗灾基金以及分发商品优惠券等.由于 EBSN 不仅包含传统社交网的线上交互(online interactions)操作,而且还包含颇具价值的线下交互(offline interactions),因此使得虚拟与物理双重的社交交互变得易于融合.EBSN 这一特色备受社交用户青睐,从而令 EBSN 变得越来越流行.

目前,典型的 EBSN 应用代表有:Meetup(meetup.com),Plancast(www.plancast.com),Groupon(www.groupon.com)和豆瓣(www.douban.com)等.在 EBSN 丰富异构信息中,包含着多种实体,比如用户(user)、事件(event)、地点(venues)、组(group)、标签(tag)和交互关系(interaction relationship)等,基于此,如何有效利用这些信息进行高质量个性化推荐是目前学术界和工业界共同关注的热点研究问题.

本文首次提出了利用超图(hypergraph)来解决 EBSN 下的推荐问题.超图节点可代表各类实体,超边则可代表相同或不同实体间的多元关系,因此超图相比于普通图能更准确地刻画异构图中各实体间的关系,信息密度较高,更适合解决推荐问题.在构建超图模型后,考虑到相比于基于欧氏距离的一般排序算法,流形排序算法(ranking of manifold)能更好体现对象间的拓扑结构,因此本文运用流行排序算法对 EBSN 超图模型进行排序,并将其命名为基于超图模型的 EBSN 个性化推荐(hypergraph-based personalized recommendation in EBSN, PRH)算法,该算法通过对超图模型运行流形排序及正则化计算,可以得到较准确推荐结果.另外,本文又立足于对查询向量设置方式进行改进和对不同类超边施以不同权重等角度,提出了一种推荐更为准确的优化 PRH(optimized PRH, oPRH)算法.

1 EBSN 特点概述

通过对 EBSN 进行深入研究,可以归纳出 EBSN 具有 4 个特点,其中后 3 个特点来源于文献[1]的总结(详细说明可参考文献[1]):

1) EBSN 中的数据具有高维关系特征.

2) EBSN 中的用户活动相比较于基于位置的社交网(location-based social networks, LBSN)具有更强的地域性[1].

3) 在 EBSN 中,用户和活动的分布满足重尾(heavy-tail)分布[1]而不是指数分布.

4) 用户间线上和线下行为具有正相关性,即线上行为相似的用户其线下行为也更相似,反之亦然.

首先,EBSN 中的数据具有高维关系特征.在 EBSN 中一般存在着不同类别的实体,比如用户、事件、地点、用户组和标签等.各实体间也存在着各种联系,比如用户与事件间有关系、事件与地点有关系、事件和组有关系、用户和标签有关系等.实体是多元的,实体间的关系也是多元的,因此 EBSN 信息实体的异构性就有别于传统社交网,使得 EBSN 数据具有高维特质.目前,高维数据的处理仍是数据处理领域的一个挑战性难题,而降维技术则是处理高维数据的重要途径之一.研究表明:大部分的降维算法都可归结于图的构造及其嵌入方式,换句话说,就是可以用普通图来表示数据之间的关系,但缺陷在于,普通图只能表达 2 点之间的关系,却不能表达多点之间的关系.不同于简单图,超图则可以表达 1 对多和多对多关系,并且可以对多阶特征进行表示,因此运用超图模型来对 EBSN 中的实体和关系进行建模就再合适不过了.

其次,EBSN 中的用户活动相比较于 LBSN 具有更强的地域性.EBSN 中无论是用户参加的活动地点,还是该用户的好友所参加的活动地点,绝大多数都集中在该用户较近距离范围之内;而且,这种

地域性在 EBSN 中相比较于 LBSN 体现得更为明显,因此,在对用户进行服务推荐时应该考虑用户的物理位置信息,即尽量将距离用户较近的事件推荐给用户.

在 EBSN 中用户和活动的分布满足重尾(heavy-tail)分布而不是指数分布.大部分事件的参与者是比较少的,拥有较多参与者的大型事件数量很少,这表现出一种重尾分布;同样,每个组的成员数量也具有非常类似的重尾分布特点.据此,在数据处理过程中,可以将用户规模太大的事件或者组去掉,从而得到更具有代表性的数据.

最后,用户间线上和线下行为具有正相关性.该特点体现在具有线上行为的用户间更有可能具有线下关系,反之亦然,这说明 EBSN 中的用户相比 LBSN 联系更加紧密,也更强调集体行为,往往以组(group)反映出来,而 LBSN 则强调个体行为.基于此,在为用户进行个性化推荐时,组和事件的关系作为精准推荐考虑要素之一需要格外关注,因为这种关联会直接影响事件的推荐效果.

2 相关工作

协同过滤作为推荐问题中最常用的一种方法可被普遍应用于 EBSN 中的推荐问题.Li 等人[2]提出一种混合协同过滤模型来对 EBSN 下的社交影响进行预测.该算法的关注点集中在预测用户对 EBSN 中即将到来事件的影响,建立了一个基于用户和事件间社交影响的矩阵,矩阵中每个实体表示的是用户对事件的影响因子,基于此通过协同过滤来实现对用户及事件的推荐.Dong 等人[3]提出一种针对训练数据的基于图的协同过滤推荐方法,主要思想是通过在时间上进行迭代来获得推荐结果.Ding 等人[4]考虑了影响用户参加活动的主要因子,并在此基础上提出一种基于滑动窗口的机器学习模型来实现对用户的事件推荐.

Macedo 等人[5]为克服冷启动问题,提出在推荐事件中,不仅应考虑用户参加的事件对结果的影响,还应同时考虑其他诸如组内成员、事件位置、时间等因素对结果的影响.在此基础上,他们提出一种基于多种信息的用户个性化事件推荐模型.文献[6]为了解决用户间隐式关系的挖掘,提出了基于 EBSN 的 2 阶段事件推荐模型(two-phase group event recommendation, 2PGER).首先,利用 EBSN 的在线社交行为、用户事件参与记录、拓扑结构等信息,建立用户间的全局信任网络;然后,在预先构建的网络上为每个用户执行随机游走,以获取用户对未体验事件的预测偏好;最后,采用重启随机游走(RWR)方法对用户偏好进行聚合,并将前 N 个事件推荐给组.Pham 等人[7]提出了一种基于普通图的推荐模型,他们主张应该考虑 EBSN 下各实体间的关系,并利用普通图来对各实体间的关系建模,通过机器学习得到最终推荐结果,该算法可以实现对用户进行组推荐、事件推荐,对组进行标签推荐等.这与本文要探讨的问题有相似之处,即同样都考虑到了各实体间的关系对推荐结果的影响,不同的是,本文提出利用超图来更好地对实体间的关系进行建模.

上述提到的 EBSN 推荐算法都是通过在普通图上建模来解决,其中文献[2-4]仅仅考虑了用户和事件对推荐结果的影响,通过加入一些影响因子或者对协同过滤算法进行一定改进来实现对用户的事件推荐,这些方法都忽略了 EBSN 中其他实体对推荐结果的影响.文献[5]和文献[7]考虑到了 EBSN 中不同类型的实体可能对推荐结果有一定影响,考虑相对来说更全面一些,推荐结果也相对更准确一些,但这些算法仍然是基于普通图来解决 EBSN 异构信息问题,因而可能导致数据在多维空间下的信息丢失问题.文献[6]尽管实现了用户偏好的预测及面向组的事件推荐,但依然没有摆脱利用简单图进行建模的局限性,即忽略了各实体间复杂关系建模的准确性会在很大程度上影响推荐结果质量的问题;Li 等人在文献[8]中综合考虑了用户个人喜好和好友之间的影响,提出了如何对用户的活动进行最有效的规划和推荐;She 等人[9]则提出一种在 ESBN 下进行活动安排的算法;文献[10]中提出了一种在普通图建模的基础上运行随机游走及熵的思路来解决 EBSN 下事件推荐;文献[11]中提出的算法也是一种运行普通图来对 EBSN 异构信息图下各种信息进行建模并进行事件推荐的算法.文献[12]中提出了一种基于用户潜在好友关系活动推荐算法.

目前超图学习在研究领域比较热门[13-17],Zhou 等人[18]提出了一种流形排序算法将数据对象列为内在几何数据结构,首先将要排序的各个点根据欧氏距离进行连接构成一个连通图,之后通过迭代运算对图中每个节点进行扩展,重复迭代过程直至最终收敛状态,此时整个图中的对象达到最优相似性;Guan 等人[19]提出了一种基于图的关联排序算法多类型对象;文献[20]介绍了超图聚类从中提取最大

相干群的算法,使用高阶(而不是成对)相似性的一组对象;基于超图的个性化推荐也已经在各个领域具有很好的应用,并且取得了显著成果,如 Bu 等人[21]提出了基于超图的音乐个性化推荐、Li 等人[22]提出基于超图的新闻个性化推荐等.

结合考虑上述算法优缺点后,本文首次提出用超图来对 EBSN 各实体间的关系进行建模,并利用超图模型通过流形排序解决 EBSN 下的推荐问题.

3 相关理论知识

本节简要介绍 EBSN 个性化推荐算法用到的一些理论基础和模型,给出超图定义及相关概念,介绍流形排序算法,描述了正则化运算的基本思路.本文用到的符号及其含义如表 1 所示:

Table 1 Meaning of Symbols

表 1 符号及其含义

Symbol	Meaning
V	Point Set of Hypergraph
E	Edge Set of Hypergraph
W	Edge Weight
$\boldsymbol{H}$	Associated Matrix of Hypergraph
$d(v)$	Degree of Node
$\delta(e)$	Degree of Edge
$\boldsymbol{Y}$	Query Vector
$Q(\boldsymbol{f})$	Cost Function
$\boldsymbol{D}_v$	Degree Matrix of Nodes
$\boldsymbol{f}^*$	Query Result Vector
μ	Regularization Parameters
U	User Set
A	Event Set
G	Group Set
L	Location Set
T	Tag Set
$\boldsymbol{W}^*$	Weight Matrix
$\boldsymbol{D}_e$	Degree Matrix of Edges

3.1 超图的定义及概念

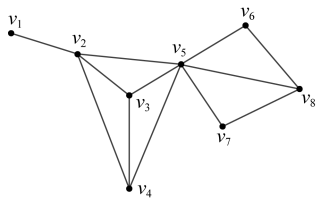
超图可以表示异构信息中各实体间的相互关联关系且不丢失信息,超图既可以表示 1 对多的关系也可以表示多对多的关系.

定义 1. 超图.设 V 是一个有限集的对象,E 是 V 中的子集的集合,即 $U_{e\in E}=V$,称 $G=(V,E)$ 是一个超图,其中 V 是点集合,E 是边集合.

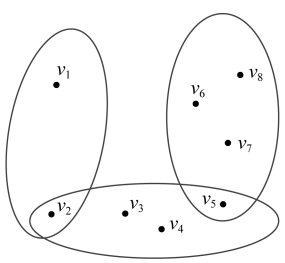
由定义 1 可知,当且仅当 E 中的每个元素 e 关联 2 个节点 v 时,超图退化为普通图.如果每个超边中包含点的个数相同,个数为 k,则可称为 k-均匀超图,普通图即为 2-均匀超图.超图包含普通图,普通图为超图的特例.超图每条超边具有不同的权重,可用 $w(e)$ 表示超边 e 的权重.超图可以表示为 $G=(V,E,w)$,其中 w 表示权重,对应于超边 e.

定义 2. 关联矩阵.将超图表示为 $|V|\times|E|$ 的矩阵 $\boldsymbol{H}$,$h(v,e)$ 表示针对该矩阵的计算规则,如果某个节点 $v\in e$,则称 e 附带 v,在超图的矩阵中 $h(v,e)$ 表示为 1;否则,如果 $v\notin e$,则在超图的矩阵中 $h(v,e)$ 表示为 0.这时矩阵 $\boldsymbol{H}$ 称为关联矩阵.

图 1 给出了解释超图相关定义的一个例子.给定一个普通图,如图 1(a)所示,其超图形式如图 1(b)所示,关联矩阵图 1(c)则表示了该超图中节点和超边的关系.



(a) Graph



(b) Hypergraph

	e_1	e_2	e_3
v_1	1	0	0
v_2	1	1	0
v_3	0	1	0
v_4	0	1	0
v_5	0	1	1
v_6	0	0	1
v_7	0	0	1
v_8	0	0	1

(c) Associated matrix

Fig. 1 A sample of the graph, hypergraph and associated matrix

图 1 普通图、超图、关联矩阵示例

根据超图的关联矩阵(如图 1(c)所示)及权重矩阵(如图 2(a)所示),计算超图节点的度 $d(v)$ 和超边的度 $\delta(e)$:

$$d(v)=\sum_{e\in E}w(e)h(v,e), \tag{1}$$

$$\delta(e)=\sum_{v\in V}h(v,e). \tag{2}$$

超图的每个顶点对应任意类的对象,而超边则被用于创建高阶关系,通过利用超图建模,可以充分挖掘出各不同实体间存在的关联关系,而不至于丢失信息.

	e_1	e_2	e_3
e_1	1	0	0
e_2	0	1	0
e_3	0	0	1

(a) Weight matrix of hyperedges

	v_1	v_2	v_3	v_4	v_5	v_6	v_7	v_8
v_1	1	0	0	0	0	0	0	0
v_2	0	2	0	0	0	0	0	0
v_3	0	0	1	0	0	0	0	0
v_4	0	0	0	1	0	0	0	0
v_5	0	0	0	0	2	0	0	0
v_6	0	0	0	0	0	1	0	0
v_7	0	0	0	0	0	0	1	0
v_8	0	0	0	0	0	0	0	1

(b) Degree matrix of nodes

	e_1	e_2	e_3
e_1	2	0	0
e_2	0	4	0
e_3	0	0	4

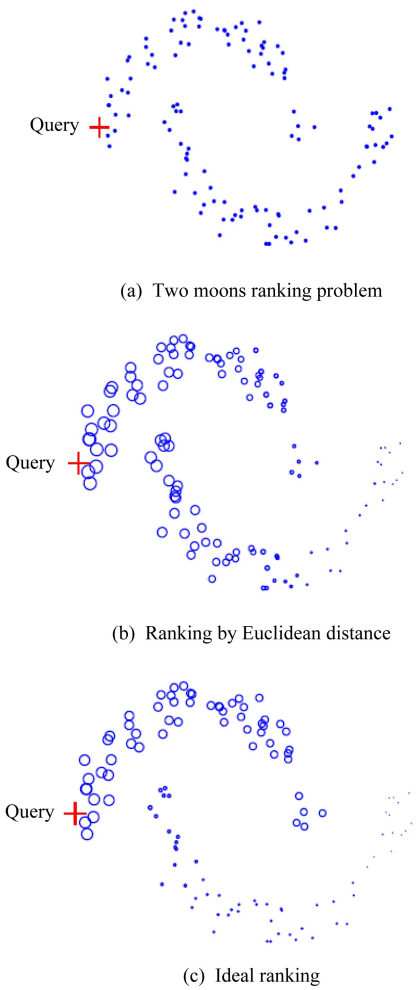
(c) Degree matrix of hyperedges

Fig. 2 Weight matrix of hyperedges, degree matrix of nodes and degree matrix of hyperedges

图 2 超边的权重矩阵、节点的度矩阵、超边的度矩阵

3.2 流形排序

基于欧氏距离的排序只能简单地根据数据点与查询点间的欧氏距离进行排序,无法表示数据点之间内在的某种关联,而流形排序[18]则能够表示数据点间的内在关联,如图 3 所示(图 3 中"+"表示查询节点).很明显,图 3(a)中的数据点分为 2 部分,即上半月牙和下半月牙,但传统基于欧氏距离的排序并不能体现出上半月牙内的点相较于下半月牙内的点与查询节点具有更紧密的关联性,如图 3(b)所示.而在图 3(c)中,通过流形排序可以更好地得到图的拓扑结构.这是由于上半月牙的某个点距离查询节点的距离更近,而上半月牙后面的点与距离查询节点近的上半月牙前面的点相比,显然要比下半月牙的点更近,那它们之间肯定会更相似.以此类推,可以得到整个上半月牙的点普遍比下半月牙的点要大,即上半月牙的点之间普遍与查询节点要更相似,这就更好地得到图了的拓扑关系.基于此,本文选择流形排作为超图模型的排序算法.



(a) Two moons ranking problem

(b) Ranking by Euclidean distance

(c) Ideal ranking

Fig. 3 An instance of the ranking manifold algorithm

图 3 流形排序示例

流形排序算法思路为:

给点集合 $\chi=\{x_1,x_2,\cdots,x_q,x_{q+1},\cdots,x_n\}\subset\mathbb{R}^m$,前 q 个点是查询,剩下的点是通过与查询点的相关性来进行排序的点.

令 $d:\chi\times\chi\to\mathbb{R}$ 表示一个在 χ 上的度量标准,例如欧氏距离,设置每对点 x_i 和 x_j 的距离为 $d(x_i,x_j)$.

令 $f:\chi\to\mathbb{R}$ 表示一个排序函数来为每个点 x_i 分配一个排序值 f_i,可以将 $\boldsymbol{f}$ 看作是一个向量 $\boldsymbol{f}=(f_1,f_2,\cdots,f_n)^{\mathrm{T}}$;同时,定义向量 $\boldsymbol{y}=(y_1,y_2,\cdots,y_n)^{\mathrm{T}}$,如果 x_i 是一个查询的话,$y_i=1$,否则 $y_i=0$.

流形排序算法步骤为:

1) 按照由小到大升序顺序,对成对关系的点的欧氏距离进行排序,然后根据得到的顺序,重复用边连接 2 个点,直至得到 1 个连通图.

2) 生成关联矩阵 $\boldsymbol{H}$,其中 $H_{ij}=\mathrm{e}^{\frac{-d^2(x_i,x_j)}{2\sigma^2}}$,如果

在 x_i 和 x_j 之间存在 1 个边连接 2 个点.注意,$H_{ii}=0$,因为图中没有环.

3) 通过 $\boldsymbol{S}=\boldsymbol{D}^{-\frac{1}{2}}\boldsymbol{H}\boldsymbol{D}^{-\frac{1}{2}}$ 将 $\boldsymbol{H}$ 对称正规化,这里,$\boldsymbol{D}$ 是对角矩阵,位于(i,i)处的元素等于矩阵 $\boldsymbol{H}$ 中第 i 行的元素和.

4) 迭代计算 $f(t+1)=\alpha\boldsymbol{S}f(t)+(1-\alpha)\boldsymbol{y}$,直至收敛,在这里 α 是一个介于$[0,1)$的参数.

5) 使 f_i^* 表示序列$\{f_i(t)\}$的界限,根据每个点 x_i 的排序得分 f_i^* 来对它们进行排序.

3.3 正则化运算

正则化代价函数 w^* 可表示为

$$w^*=\arg\min_w\sum_{i=1}^{m}L(y_i,f(x_i;w))+\lambda\Omega(w),\quad(3)$$

其中,m 为样本数,第 1 项 $L(y_i,f(x_i;w))$是平滑项,衡量模型第 i 个样本的预测值 $f(x_i;w)$和真实标签 y_i 之间的误差.为让模型尽量拟合训练数据,即保证训练误差最小,要求这一项最小.此外,还希望模型测试误差也要小,因而需要加上第 2 项 $\lambda\Omega(w)$,即正则项,它是参数 w 的正则化函数,用以约束模型尽量简单,其中 λ 为正则化因子.

机器学习中大部分带参模型都与代价函数比较相像,只是变换第 1 项和第 2 项而已.对于第 1 项 loss 函数,如果是 square loss,那就是最小二乘;如果是 hinge loss,那就是 SVM;如果是 exp-loss,那就是 Boosting 等.不同 loss 函数,具有不同拟合特性,本文中第 1 项为基于流形排序的 loss 函数.第 2 个正则项可分为 L_0,L_1,L_2 范数和核范数正则化,为防止出现过拟合,提升模型泛化能力,本文采取 L_2 范数正则化,即$\|W\|^2$.

4 PRH 推荐算法和 oPRH 优化算法

本节介绍了基于超图模型的 PRH 推荐算法,然后在 PRH 算法基础上给出了 2 种优化策略,即 oPRH 算法.

4.1 PRH 算法

PRH 算法首先分析 EBSN 中实体间的关系,然后根据已得到的关系构建超图模型;接着构造超图的关联矩阵并设置超边权重,求得节点的度矩阵及超边的度矩阵;之后,根据查询目标点设置查询向量 $\boldsymbol{y}$,将 $\boldsymbol{y}$ 带入超图的流形排序正则化运算中,通过计算得到最终的结果向量 $\boldsymbol{f}$,最终根据要求取向量 $\boldsymbol{f}$ 中前 k 个标量作为推荐结果.下面详细阐述超图建模和超图的流形排序.

4.1.1 用超图对 EBSN 进行建模

本文选取 Meetup 应用作为分析 EBSN 的样例.通过分析,得到 Meetup 中的模式有 6 种关系:

1) 1 个用户可以加入到多个事件,1 个事件也可以有多个用户.如用户 u_i 可以参加篮球、足球事件,而每个事件会有多个用户参与.

2) 1 个用户可以加入多个组,1 个组也可以有多个用户.如用户 u_i 可以同时参加户外运动、音乐等多个社交组,并且每个社交组中也可以有多个用户参与.

3) 组可以组织各种事件,即 1 个组中可以有多个事件.如户外运动组可以组织爬山事件,还可以组织篮球比赛等.

4) 事件在 1 个地点举办,1 个地点也可以有多个事件.如篮球比赛会在篮球场举办,同时篮球场可能还会举办诸如拔河等其他事件.

5) 用户可以同时具有活动、音乐、学习等多个标签.

6) 社交组可以同时具有艺术、音乐、美术等多个标签.

本文考虑了 EBSN 中 5 种实体类型和 6 种实体间关系类型.实体类型分别为:用户(U)、事件(A)、组(G)、地点(L)和标签(T).关系类型分别为:用户参加事件的关系 R_1、用户参加组的关系 R_2、用户组与事件的关系 R_3、事件和地点的关系 R_4、用户和标签的关系 R_5、用户组和标签的关系 R_6.

5 种实体类型组成了超图的节点集合 V,即 $V=U\cup A\cup G\cup L\cup T$.6 种超边,每一种超边对应一种实体间关系,如表 2 所示.定义超边集合为 $E(i)$对应于 R_i,$i=1,2,\cdots,6$,6 种超边类型构造为:

$E(1)$:构造一种用户参加活动的超边关系,并设置权重为 1.

$E(2)$:对于每个组,构建了一种组内包含所有用户的超边,组本身也是超边的对象,设置权重为 $w(e_{ij}^{(2)})$,$w(e_{ij}^{(2)})$表示为组内包含的用户个数,为了消除偏差,标准化后 $w(e_{ij}^{(2)})'$可表示为

$$w(e_{ij}^{(2)})'=\frac{w(e_{ij}^{(2)})}{\max(w(e_{ij}^{(2)}))}.\quad(4)$$

$E(3)$:对于每个组,构建了一种组内包含所有事件的超边,组本身也是超边的对象,设置权重为 $w(e_{ij}^{(3)})$,$w(e_{ij}^{(3)})$表示为组内包含的事件个数,为了消除偏差,标准化后 $w(e_{ij}^{(3)})'$可表示为

$$w(e_{ij}^{(3)})'=\frac{w(e_{ij}^{(3)})}{\max(w(e_{ij}^{(3)}))}. \tag{5}$$

$E(4)$:事件与地点的关系作为一种超边,权重为1.

$E(5)$:用户与标签的关系作为一种超边,权重为1.

$E(6)$:用户组与标签的关系作为一种超边,权重为1.

另外,由分析可知,不同类超边关系对最终推荐结果的影响程度是不同的,因而根据每类超边对结果的影响程度,给关联矩阵中不同类超边关系加一个系数 C_i, $i=1,2,\cdots,6$,满足 $0<C_i<1$ 并且 $\sum_{i=1}^{6}C_i=1$. 这点在 oPRH 算法中会有进一步说明.

Table 2 Associated Matrices Among Nodes and Edges

表 2 顶点-超边的关联矩阵

Node	$E(1)$	$E(2)$	$E(3)$	$E(4)$	$E(5)$	$E(6)$
U	$UE(1)$	$UE(2)$	0	0	$UE(5)$	0
A	$AE(1)$	0	$AE(3)$	$AE(4)$	0	0
G	0	$GE(2)$	$GE(3)$	0	0	$GE(6)$
L	0	0	0	$LE(4)$	0	0
T	0	0	0	0	$TE(5)$	$TE(6)$

4.1.2 超图流形排序

推荐问题进一步可以转换为与输入向量相似度高低的排序问题,排序问题需要找到最优解,如何找到最优解就需要构造一个代价函数:

$$Q(\boldsymbol{f})=\frac{1}{2}\sum_{i,j=1}^{|V|}\sum_{e\in E}\frac{1}{\delta(e)}\sum_{\{v_i,v_j\}\subseteq e}w(e)\times\left\|\frac{f_i}{\sqrt{d(v_i)}}-\frac{f_j}{\sqrt{d(v_j)}}\right\|^2+\mu\sum_{i=1}^{|V|}\|f_i-y_i\|^2, \tag{6}$$

其中,$\boldsymbol{f}$ 为输入向量,正则化参数 $\mu>0$.

该代价函数的主要思想是对两两节点在所有包含它们的超边中求方差,即进行相似度比较.显然,如果2个节点在所有包含它们的超边中都更相似的话,那么说明两者一定存在更强的相似性,即两者间的方差和越小,说明2个节点更相似.另外,超边的度越大,两者在这种超边中的相似性越显得不重要.举个例子,现在有节点 A, B, C,如果要给节点 A 推荐一个相似性更高的节点,那么可以看节点 B, C 所属的超边与节点 A 所属的超边的关系,如果节点 B 所在的超边与节点 A 所在的一模一样,而节点 C 只有部分所属的超边与节点 A 相同,那么显然节点 B 相较于节点 C, A 具有更高的相似性,就会选择将节点 B 推荐给节点 A.进一步,对整个模型中所有两两节点求方差以得到方差和,当方差和达到最小时,对整个模型来说,达到了最好的相似性.对 $Q(\boldsymbol{f})$ 加正则化约束项可使之更准确.当 $Q(\boldsymbol{f})$ 最小时,即对 $Q(\boldsymbol{f})$ 求导为0时,可以得到最优排序结果.通过对 $Q(\boldsymbol{f})$ 表达式进行变形和化简,并依据流行排序公式,可以得到模型达到最优时的结果向量 $\boldsymbol{y}$.

最优排序结果是当 $Q(\boldsymbol{f})$ 最小时实现 $\boldsymbol{f}^*$:

$$\boldsymbol{f}^*=\arg\min_{\boldsymbol{f}}Q(\boldsymbol{f}). \tag{7}$$

在式(6)中,等式右侧的第1项可以改写为

$$\frac{1}{2}\sum_{i,j=1}^{|V|}\sum_{e\in E}\frac{1}{\delta(e)}\sum_{\{v_i,v_j\}\subseteq e}w(e)\left\|\frac{f_i}{\sqrt{d(v_i)}}-\frac{f_j}{\sqrt{d(v_j)}}\right\|^2=\frac{1}{2}\sum_{i,j=1}^{|V|}\sum_{e\in E}\frac{w(e)h(v_i,e)h(v_j,e)}{\delta(e)}\left\|\frac{f_i}{\sqrt{d(v_i)}}-\frac{f_j}{\sqrt{d(v_j)}}\right\|^2=$$

$$\sum_{i=1}^{|V|}f_i^2\sum_{e\in E}\frac{w(e)h(v_i,e)}{d(v_i)}\sum_{j=1}^{|V|}\frac{h(v_j,e)}{\delta(e)}-\sum_{i,j=1}^{|V|}\sum_{e\in E}\frac{f_iw(e)h(v_i,e)h(v_j,e)f_j}{\sqrt{d(v_i)d(v_j)}\delta(e)}=$$

$$\sum_{i=1}^{|V|}f_i^2-\sum_{i,j=1}^{|V|}\sum_{e\in E}\frac{f_iw(e)h(v_i,e)h(v_j,e)f_j}{\sqrt{d(v_i)d(v_j)}\delta(e)}=\boldsymbol{f}^{\mathrm{T}}\boldsymbol{f}-\boldsymbol{f}^{\mathrm{T}}\boldsymbol{D}_v^{-\frac{1}{2}}\boldsymbol{H}\boldsymbol{W}^*\boldsymbol{D}_e^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{D}_v^{-\frac{1}{2}}\boldsymbol{f}.$$

定义一个矩阵 $\boldsymbol{A}$:

$$\boldsymbol{A}=\boldsymbol{D}_v^{-\frac{1}{2}}\boldsymbol{H}\boldsymbol{W}^*\boldsymbol{D}_e^{-1}\boldsymbol{H}^{\mathrm{T}}\boldsymbol{D}_v^{-\frac{1}{2}}, \tag{8}$$

则 $\boldsymbol{L}=\boldsymbol{I}-\boldsymbol{A}$ 是超图的半正定拉普拉斯矩阵[16].

然后,可将代价函数写成矩阵形式:

$$Q(\boldsymbol{f})=\boldsymbol{f}^{\mathrm{T}}(\boldsymbol{I}-\boldsymbol{A})\boldsymbol{f}+\mu(\boldsymbol{f}-\boldsymbol{y})^{\mathrm{T}}(\boldsymbol{f}-\boldsymbol{y}).$$

要使代价函数最小,对 $Q(\boldsymbol{f})$ 求导使其等于0,即:

$$\frac{\partial Q}{\partial \boldsymbol{f}}\Big|_{\boldsymbol{f}=\boldsymbol{f}^*}=(\boldsymbol{I}-\boldsymbol{A})\boldsymbol{f}^*+\mu(\boldsymbol{f}^*-\boldsymbol{y})=0.$$

通过一些简单的代数步骤,可以得到:

$$\boldsymbol{f}^*=\frac{\mu}{1+\mu}\left(\boldsymbol{I}-\frac{1}{1+\mu}\boldsymbol{A}\right)^{-1}\boldsymbol{y}. \tag{9}$$

假设 $\alpha=\frac{\mu}{1+\mu}$.注意到 $\frac{\mu}{1+\mu}$ 是一个常数并不改变排序结果,可重写 $\boldsymbol{f}^*$.

$$\boldsymbol{f}^*=(\boldsymbol{I}-\alpha\boldsymbol{A})^{-1}\boldsymbol{y}, \tag{10}$$

从式(10)可以发现矩阵 $\boldsymbol{I}-\alpha\boldsymbol{A}$ 是可逆的.注意矩阵 $\boldsymbol{I}-\alpha\boldsymbol{A}$ 是高度稀疏的,因此,计算效率非常高.

运算得到的 $\boldsymbol{f}^*$ 为结果向量,对 $\boldsymbol{f}^*$ 中各分量值进行排序,选取其中符合条件的前 k 个点实体作为推荐列表推荐给用户.

4.2 oPRH 优化算法

实际上,可以从 2 个角度对 PRH 算法做推荐优化:1)通过改进设置查询向量方式;2)对不同类超边施以不同权重.下面,分别在 4.2.1 节和 4.2.2 节中加以阐述.

4.2.1 基于改进查询向量方式的 oPRH 算法

根据查询目标点,有 3 种设置查询向量方式,具体操作:

1) 设置查询向量 $\boldsymbol{y}$ 与查询目标用户的分量为 1,其他分量为 0.

2) 设置查询向量 $\boldsymbol{y}$ 与查询目标用户的分量为 1,同时设置与查询用户相关的分量也为 1,其他分量为 0.

3) 设置查询向量 $\boldsymbol{y}$ 与查询目标用户的分量为 1,同时设置与查询用户相关的分量为 $A_{u,v}$($0<A_{u,v}<1$),$A_{u,v}$ 主要根据各实体与查询用户的相关度设置,其他分量为 0.

方式 1 没有考虑查询目标点与其他节点的关系,关联性差.方式 2 只是简单地将查询目标点与其他节点的关系设置为 1,尽管考虑了关联性,但关系描述尚不够准确,因为查询目标点与其他节点的关联程度是不同的.基于上述 2 点,本文采取方法 3 设置查询向量方式,旨在进一步提升推荐结果的准确度.

4.2.2 基于不同类超边权重的 oPRH 算法

在 4.1.1 节中曾论述过,除了同类的各超边会对推荐结果有不同程度影响,不同类的超边也会对推荐结果有不同程度影响,为此,本文尝试通过对关联矩阵中不同类超边关系增加系数 C_i 的方法,分别从几何学、多元统计分析和线性回归 3 个角度,即基于单形的体积、散布矩阵的迹和线性重构误差,重新度量了点集合间的相似性,由此提出了 3 种超边加权方法.为阐述清晰,将其统称为 oPRH 算法,旨在进一步提升推荐精准度,具体细节为:

1) 基于单形体积的超边加权法

从几何学角度看,每个超边可以被看作一个单形,因此,从几何学来衡量点集合可以通过计算单形的体积来获得,因为一个小的单形体积意味着超边中的节点具有更紧密的集合关系,反之相反.

通过使用单形中的节点来计算它的体积

$$Vol(E_j)=\frac{\sqrt{|\det(\boldsymbol{G}^{\mathrm{T}}\boldsymbol{G})|}}{k!},\tag{11}$$

其中,$\det(\cdot)$是矩阵的行列式,$k!$是 k 的阶乘,$\boldsymbol{G}$ 是定义的 $k\times k$ 阶的矩阵,其中列向量 $\boldsymbol{g}_i=(\boldsymbol{x}_0-\boldsymbol{x}_i)$.

在得到单形体积之后,与单形 E_j 相关联的超边 e_j 可以表示为

$$w(e_j)=\mathrm{e}^{-\frac{Vol(E_j)}{\mu}},\tag{12}$$

其中,μ 是一个正参数.

2) 基于散布矩阵迹的超边加权法

从多元统计分析和数据挖掘角度看,每个超边可以被看作为样例空间中的聚簇,因此可以使用散布矩阵来衡量超边聚簇的紧密型.定义 $k\times d$ 的矩阵 $\boldsymbol{X}=(\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_k)$为与 k 度超边 e_j 节点相关联的样例矩阵.然后,散布矩阵 $\boldsymbol{S}$ 的计算:

$$\boldsymbol{S}=\sum_{i=1}^{k}(\boldsymbol{x}_i-\bar{\boldsymbol{x}})(\boldsymbol{x}_i-\bar{\boldsymbol{x}})^{\mathrm{T}}=(\boldsymbol{X}-\bar{\boldsymbol{X}})(\boldsymbol{X}-\bar{\boldsymbol{X}})^{\mathrm{T}},\tag{13}$$

其中,$\bar{\boldsymbol{x}}$ 是样例的平均值,$\bar{\boldsymbol{X}}$ 是 $d\times k$ 维向量,其中列全部为 $\bar{\boldsymbol{x}}$.可以设置超边 e_j 的权重:

$$w(e_j)=\mathrm{tr}\left(\frac{-\mathrm{e}^{S}}{\mu}\right),\tag{14}$$

其中,$\mathrm{tr}(\cdot)$表示为矩阵的迹,μ 是一个正参数.

3) 基于线性重构误差的超边加权法

受到在机器学习和计算机视觉方向线性回归的启发[23-26],可以使用线性回归误差来衡量超边,因为重构误差在衡量同构体例子时会得到比异构体例子更小的值.在无向超图中,每个节点可以得到重构误差,假设 $k\times d$ 维的样例 $\boldsymbol{x}_1,\boldsymbol{x}_2,\cdots,\boldsymbol{x}_k$ 与在 k 度超边 e_j 中有序节点相对应.样例 x_i 的重构系数 $\boldsymbol{c}_i$ 是用来最小化重构误差的,可以被当作一个最小二乘方问题来解决:

$$\hat{\boldsymbol{c}}_i=\arg\min_{\boldsymbol{c}_i}(\|\boldsymbol{x}_i-\boldsymbol{X}_{t\neq i,t\in e_j}\boldsymbol{c}_i^{\mathrm{T}}\|^2),\tag{15}$$

其中,$\boldsymbol{X}_{t\neq i,t\in e_j}$ 是一个 $d\times(k-1)$维的矩阵,矩阵的第 t 列是 $\boldsymbol{x}_t$,$t\neq i$ 并且 $1\leqslant t\leqslant k$.在得到 $\boldsymbol{c}_i$ 后则可以得到相关重构误差 r_i:

$$r_i=\frac{\|\boldsymbol{x}_i-\boldsymbol{X}_{t\neq i,t\in e_j}\hat{\boldsymbol{c}}_i^{\mathrm{T}}\|^2}{\|\boldsymbol{x}_i\|^2}.\tag{16}$$

对于一个无向超图来说,其一条超边的整个重构误差 R 可以灵活地表示为所有重构误差样例的平均值:

$$R=\frac{1}{k}\sum_{i=1}^{k}r_i.\tag{17}$$

最终,可以得到超边 e_j 的权重 $w(e_j)$:

$$w(e_j)=\mathrm{e}^{-\frac{R}{\mu}},\tag{18}$$

其中,μ 是一个正参数.

5 性能测试与分析

5.1 实验环境与实验数据

本文实验数据爬取自 Meetup 网站,采用 MongoDB 对爬取的数据进行存储,使用 Java 语言编程算法,并用 Matlab 实现矩阵的正则化运算.

设备相关参数包括处理器:Intel® Core™ i7-6700;主频:2.8 GHz,4 核;内存:16 GB;操作系统:Windows 10.

实验采用的数据集分为 2 种:1)2015 年 1～6 月旧金山(SF)地区的 Meetup 数据;2)2015 年 1～6 月洛杉矶(LA)地区的 Meetup 数据.这些数据都可以从 Meetup API 网站[①]自行爬取.

由第 1 节 EBSN 特性分析可知,实验中需要将用户数目过多的事件和组去掉,因此在实验中将用户规模大于 200 的事件和组筛选掉.同时,考虑到用户数目过少的事件和组对实验结果的影响也很微弱,因而将用户规模小于 5 的事件和组也筛选掉.接下来,在处理后的数据集中选取 20%的数据作为测试数据,并去掉实验数据与测试数据之间的关联,之后对实验数据进行运算,并使用测试数据验证实验性能.实验数据中涉及到的对象和关系信息如表 3～6 所示:

Table 3 Objects Information of SF Data Set

表 3 SF 数据集中的对象信息

Object	Number	Object	Number
U	26 631	L	292
A	2 262	T	11 415
G	569		

Table 4 Relationships Information of SF Data Set

表 4 SF 数据集中的关系信息

Relation	Number	Relation	Number
R_1	26 631	R_4	292
R_2	569	R_5	26 631
R_3	569	R_6	569

Table 5 Objects Information of LA Data Set

表 5 LA 数据集中的对象信息

Object	Number	Object	Number
U	31 442	L	275
A	2 135	T	12 873
G	578		

Table 6 Relationships Information of LA Data Set

表 6 LA 数据集中的关系信息

Relation	Number	Relation	Number
R_1	31 442	R_4	275
R_2	578	R_5	31 442
R_3	578	R_6	578

5.2 性能评估指标

本文用准确率(precision, P)、召回率(recall, R)、$F1$ 值($F1$ measure),$P_{M,A}$(mean average precision)以及 $G_{N,D,C}$(normalize discounted cumulative gain)等评测指标来衡量算法性能.其中:

准确率=推荐准确的个数/推荐总个数,本文中准确率=推荐准确事件数/推荐总事件数:

$$P=\frac{\sum_{u\in U}|R(u)\cap T(u)|}{\sum_{u\in U}|R(u)|}. \tag{19}$$

召回率=推荐准确的个数/样本数目,本文中召回率=推荐准确事件数/用户参加事件数:

$$R=\frac{\sum_{u\in U}|R(u)\cap T(u)|}{\sum_{u\in U}|T(u)|}. \tag{20}$$

式(19)和式(20)中,$R(u)$表示对用户推荐的事件集合,$T(u)$表示用户参加过的事件集合.

F 值是准去率和召回率的加权调和平均:

$$F=\frac{(\alpha^2+1)P\times R}{\alpha^2(P+R)}. \tag{21}$$

当参数 $\alpha=1$ 时,F 就是 F_1:

$$F1=\frac{2\times P\times R}{P+R}. \tag{22}$$

$P_{M,A}$指的是所有用户的平均准确率(average precision, P_A)的平均值. P_A 指的是在推荐列表中,每个正确推荐项目点计算得到的准确率的平均值:

$$P_A=\frac{1}{M}\sum_{i=1}^{N}P_i\times C_i, \tag{23}$$

其中,N 是推荐项目的数目,M 为正确推荐项目的数目,P_i 是在排序点 i 处的准确率,如果在 i 处的项目被准确推荐,那么 $C_i=0$,否则 $C_i=1$.

$G_{N,D,C}$是衡量搜索引擎算法的指标,但它也可以作为衡量排序质量的指标.在 n 处 $G_{N,D,C}$:

$$G_{N,D,C_n}=\frac{G_{D,C}}{G_{I,D,C}}=\frac{1}{G_{I,D,C}}\times\sum_{i=1}^{n}\frac{2^{r_i}-1}{\mathrm{lb}(i+1)}, \tag{24}$$

其中,r_i 是排在 i 处的项目相关度程度,即排在 i 处的事件相关性程度.本文中,如果用户参加了该事件,

① https://www.meetup.com/meetup_api/

则 $r_i=1$,否则 $r_i=0$.式(24)中,$G_{D,C}=\dfrac{2^{r_i}-1}{\mathrm{lb}(i+1)}$,其体现的主要思想是,如果在排序结果中,等级比较高的结果排名却比较靠后,那么在最终统计分数时,就应该对这个排序结果的得分进行打折.$G_{I,D,C}$ 则是理想的 $G_{D,C}$,也就是排序中最好的状态.

5.3 PRH 算法性能测试与分析

为检测 PRH 算法推荐效果,将其与 6 种算法进行了比较,6 种算法分别为协同过滤(CF)、基于普通图的面向异构信息的 EBSN 推荐算法(HeteRS)[6]、不加影响因子的 HeteRS 算法(uni_HeteRS)[6]、基于随机游走的推荐算法(RWR)、基于普通图熵的推荐算法[9]以及基于多特征的事件推荐算法[10].

首先,针对 2 个不同数据集(SF 和 LA),比较了 6 种算法在时间粒度分别为 1 个月、3 个月、6 个月下的准确率和召回率,实验结果如图 4 和图 5 所示:

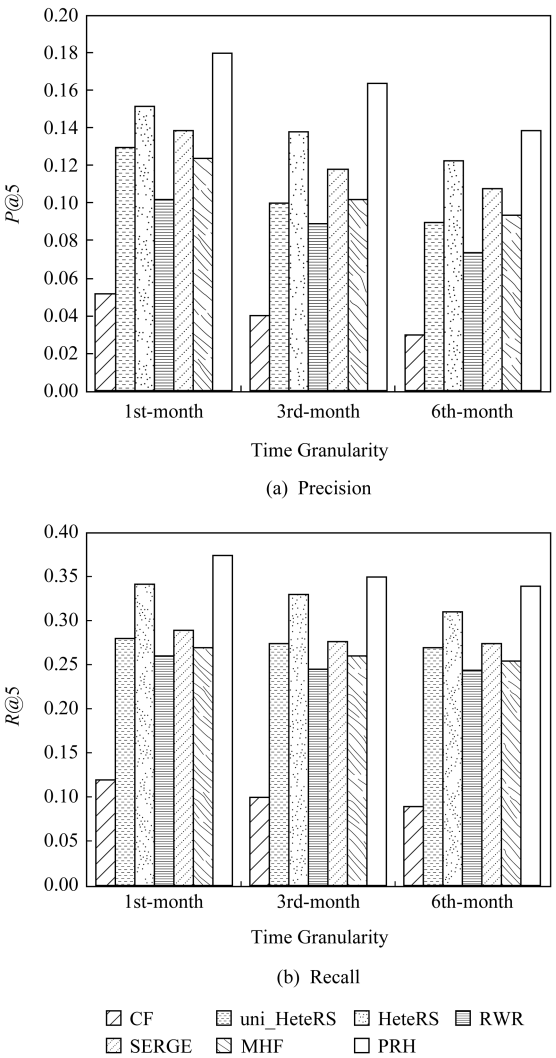


Fig. 4 Precision and recall of SF data set in different time

图 4 SF 数据集在不同时间粒度下的准确率和召回率

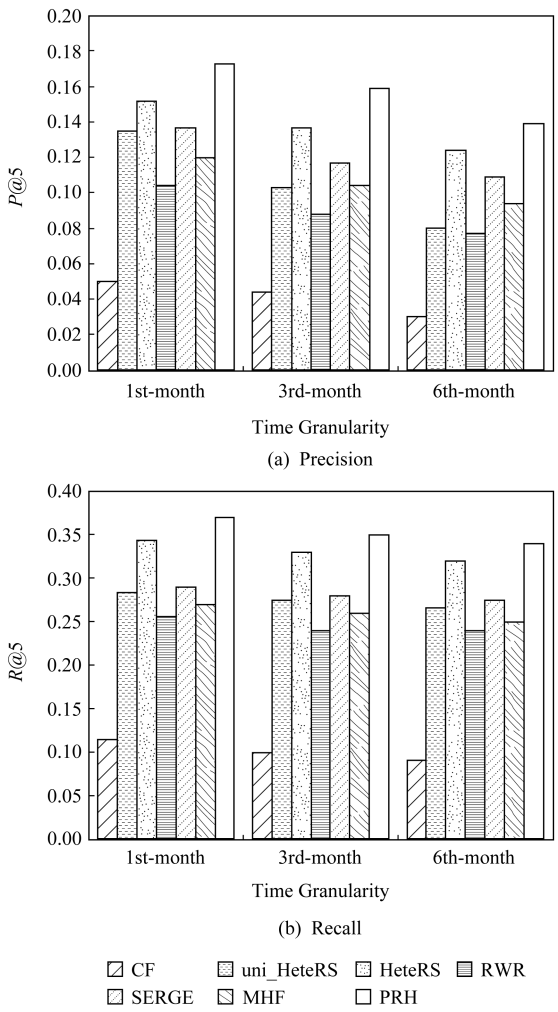


Fig. 5 Precision and recall of LA data set in different time

图 5 LA 数据集在不同时间粒度下的准确率和召回率

从图 4 和图 5 中可以看到,随着时间粒度变大,算法的推荐准确性都会下降,因为用户的兴趣点可能会随时间流逝而发生变化,比如,用户会在这段时间内不断参加别的事件,因而可能会造成推荐结果的准确性下降.因此,要预测一个用户距现在较长时间可能感兴趣的事件是很困难的,这对选取多长时间粒度的数据提出了要求,数据时间粒度过小可能会造成数据信息不全面,进而影响推荐结果准确率,时间粒度过长的数据也同样可能造成推荐准确性下降.本文最终选取了时间粒度为 6 个月的数据,实验证明该时间粒度比较合理.

其次,针对 2 个不同数据集(SF 和 LA),比较了 7 种算法在位置数 N 的准确率和召回率,实验结果如图 6 和图 7 所示.

从图 6 和图 7 可知,CF 算法在位置数 N 的准确率和召回率相较于其他 6 种算法差距较大,因为

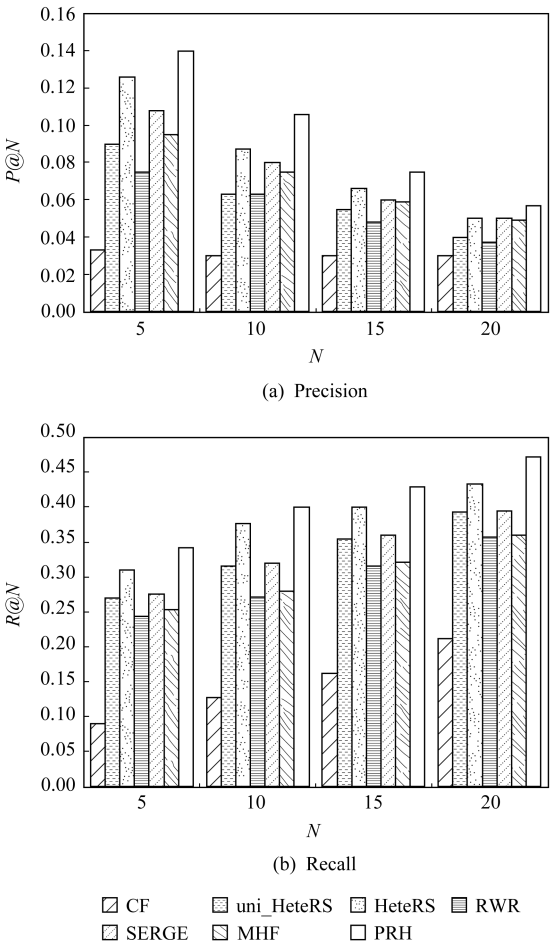


Fig. 6 Precision and recall of SF data set when the number of Location is N

图 6 SF 数据集在位置数 N 处的准确率和召回率

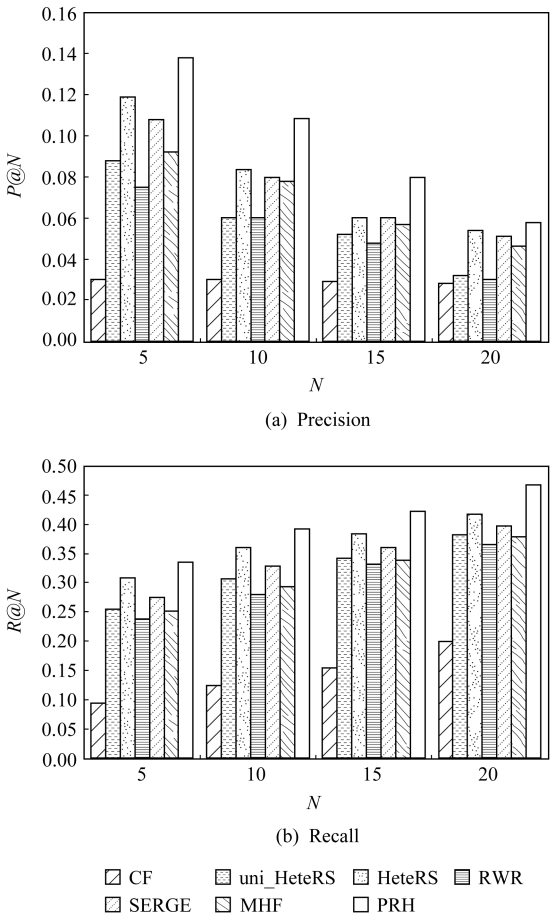


Fig. 7 Precision and recall of LA data set when the number of Location is N

图 7 LA 数据集在位置数 N 处的准确率和召回率

CF 算法相对来说比较简单,因而在推荐结果的准确性上可能比较欠缺,HeteRS 是除本文提出的算法外表现最好的一种算法,SERGE 和 MHF 算法推荐性能也不错,但由于跟 HeteRS 一样都是基于普通图对 EBSN 异构信息图进行建模,虽然考虑了事件推荐中其他各因素对推荐结果的影响,但由于普通图在对异构信息图进行的建模不如超图,所以本文中提出的基于超图建模的推荐方法在推荐性能上还是更优一些.7 种算法的准确率都随着位置数 N 的变大而变小,与此相反,召回率都随着位置数 N 的变大而变大.

紧接着,针对 2 个不同数据集(SF 和 LA),比较了各算法的 $P_{M,A}$,$F1$,$G_{N,D,C}$ 指标,实验结果如表 7~10 所示.由表 7~10 可知,PRH 算法在这 4 个指标上性能均优于其他推荐算法(PRH 算法加以 * 表示).尤其是在位置数 N 较小时,这种优越性表现得更为明显,这是因为 PRH 算法是利用超图对高维数据进行建模,而其余集中算法都是基于普通图的基础上进行建模,PRH 算法相较于其他算法性能更优,说明了利用超图对复杂的社交网络数据关系建模相较于普通图确实是一种更好的选择.同时,从实验结果中可以观察到,CF 算法性能是最差的,这可能是 CF 算法没有考虑复杂的社交网络中各种实体都可能对推荐结果有一定影响的原因.

Table 7 $P_{M,A}$ and $F1$ of Seven Algorithms Based on SF Data Set

表 7 7 种算法在 SF 数据集下的 $P_{M,A}$ 和 $F1$

Algorithm	$P_{M,A}$	$F1@5$	$F1@10$	$F1@15$	$F1@20$
CF	0.1194	0.0414	0.0432	0.0457	0.0492
uni_HeteRS	0.1932	0.1289	0.0975	0.0742	0.0614
HeteRS	0.2123	0.1645	0.1294	0.0904	0.0692
RWR	0.1521	0.1212	0.0913	0.0716	0.0578
SERGE	0.1989	0.1613	0.1279	0.0885	0.0742
MHF	0.1876	0.1544	0.1246	0.0844	0.0721
PRH	0.2286*	0.1879*	0.1561*	0.1179*	0.0943*

Table 8 $G_{N,D,C}$ of Seven Algorithms Based on SF Data Set

表 8 7 种算法在 SF 数据集下的 $G_{N,D,C}$

Algorithm	$G_{N,D,C}$@5	$G_{N,D,C}$@10	$G_{N,D,C}$@15	$G_{N,D,C}$@20
CF	0.153 2	0.244 5	0.342 3	0.381 8
uni_HeteRS	0.451 2	0.386 4	0.411 7	0.431 2
HeteRS	0.476 5	0.397 8	0.424 7	0.449 7
RWR	0.335 4	0.343 4	0.352 4	0.364 7
SERGE	0.459 8	0.388 7	0.418 7	0.433 4
MHF	0.424 6	0.377 9	0.406 4	0.424 8
PRH	0.509 8*	0.432 7*	0.453 5*	0.473 3*

Table 9 $P_{M,A}$ and $F1$ of Seven Algorithms Based on LA Data Set

表 9 7 种算法在 LA 数据集下的 $P_{M,A}$ 和 $F1$

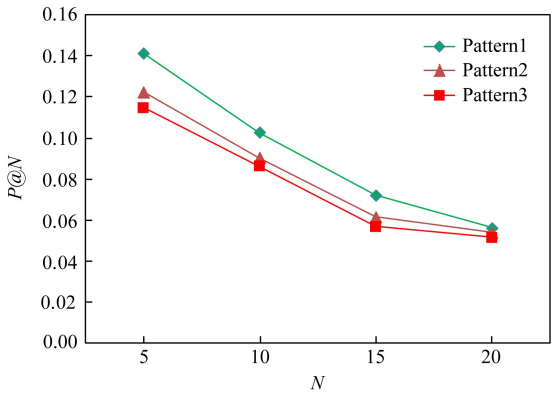
Algorithm	$P_{M,A}$	$F1$@5	$F1$@10	$F1$@15	$F1$@20
CF	0.125 3	0.046 8	0.048 6	0.049 3	0.051 1
uni_HeteRS	0.197 6	0.134 5	0.104 2	0.084 4	0.068 6
HeteRS	0.216 8	0.173 7	0.136 6	0.095 5	0.076 7
RWR	0.154 3	0.123 3	0.094 5	0.077 1	0.058 9
SERGE	0.201 1	0.161 1	0.130 1	0.089 6	0.073 4
MHF	0.187 5	0.154 1	0.121 9	0.085 4	0.070 1
PRH	0.239 1*	0.199 8*	0.165 4*	0.129 8*	0.102 3*

Table 10 $G_{N,D,C}$ of Seven Algorithms Based on LA Data Set

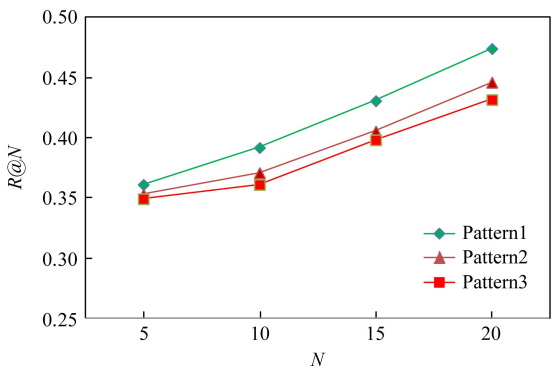
表 10 7 种算法在 LA 数据集下的 $G_{N,D,C}$

Algorithm	$G_{N,D,C}$@5	$G_{N,D,C}$@10	$G_{N,D,C}$@15	$G_{N,D,C}$@20
CF	0.147 9	0.236 4	0.334 6	0.381 8
uni_HeteRS	0.441 5	0.374 6	0.404 1	0.431 2
HeteRS	0.468 7	0.382 4	0.411 4	0.449 7
RWR	0.341 4	0.349 5	0.360 4	0.366 2
SERGE	0.455 7	0.391 2	0.408 2	0.439 4
MHF	0.432 4	0.380 4	0.402 3	0.423 2
PRH	0.499 7*	0.427 8*	0.448 1*	0.466 4*

首先,利用准确率和召回率 2 个指标,验证不同查询向量设置方式对最终实验结果的准确性产生的影响,实验结果如图 8 所示.从图 8(a)中可以发现,随着位置数 N 的上升,3 种设置查询向量方式的准确率都会下降,而由图 10(b)可以发现,随着位置数 N 的上升,召回率则都会上升.另外,从图 8 中还可以发现,方式 3 设置查询向量的方式不管是准确率还是召回率都要优于方式 1、方式 2,方式 2 则稍微比方式 1 要好一点.原因是,由于方式 2 只是简单地将与查询节点相关的点全部设置成 1,所以对最终推荐结果的提升并不明显,而方式 3 则根据对象与查询节点相关程度的紧密性将向量设置为 0~1 之间的值,所以方式 3 要明显优于方式 1、方式 2.



(a) Precision



(b) Recall

Fig. 8 Precision and recall in three different query vectors when the number of Location is N

图 8 3 种不同查询向量在位置数 N 处的准确率和召回率

5.4 oPRH 算法性能测试与分析

为验证不同种类超边对最终推荐结果也确有不同影响,分别测试了 PRH 算法在不考虑系数和考虑系数情况下且处于位置数 N 的准确率和召回率.为避免混淆,将不考虑系数的 PRH 算法称为 uni_PRH 算法,而将考虑系数的 PRH 算法仍简称为 PRH 算法,实验结果如图 9 所示.

由图 9 可以看到,加系数的 PRH 算法其准确率和召回率曲线与坐标轴围成的面积要大于不加系数的 uni_PRH 算法,因而可以得到加系数的 PRH 算法的推荐准确性要好于不加系数的 uni_PRH,这说明对超边加系数的方法确实可以提高推荐算法的性能.

进一步,为验证所提 3 种不同种类超边加权方法哪种能更有效提升推荐质量,又将原始未考虑系数的 PRH 算法与 3 种加权方法下的 oPRH 算法进行了准确率和召回率指标上的测试,结果如图 10 所示.由图 10(a)可以看到,随着位置数 N 的增加,4 种算法的推荐准确率都会随之下降,同时 3 种改进的加权方式在位置数 N 下得到的准确率都要高于

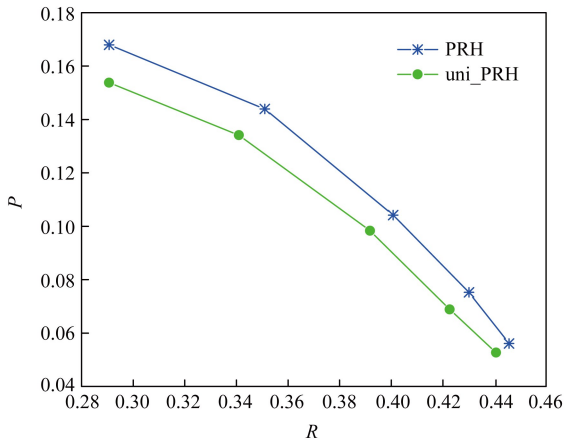


Fig. 9 Comparison between PRH and uni_PRH

图 9 PRH 算法和 uni_PRH 算法比较

原始 PRH 算法.除此之外,还可以发现加权方式 1 和加权方式 3 在准确率上要略差于加权方式 2.同样,由图 10(b)可以看到,随着位置数 N 的增加,4 种算法召回率都会随之上升,而 3 种改进加权方式在位置数 N 的召回率也要高于原始 PRH 算法,另外加权方式 2 在召回率方面也要优于另外 2 种方式.接下来,基于上述实验分析,本文选择采用加权

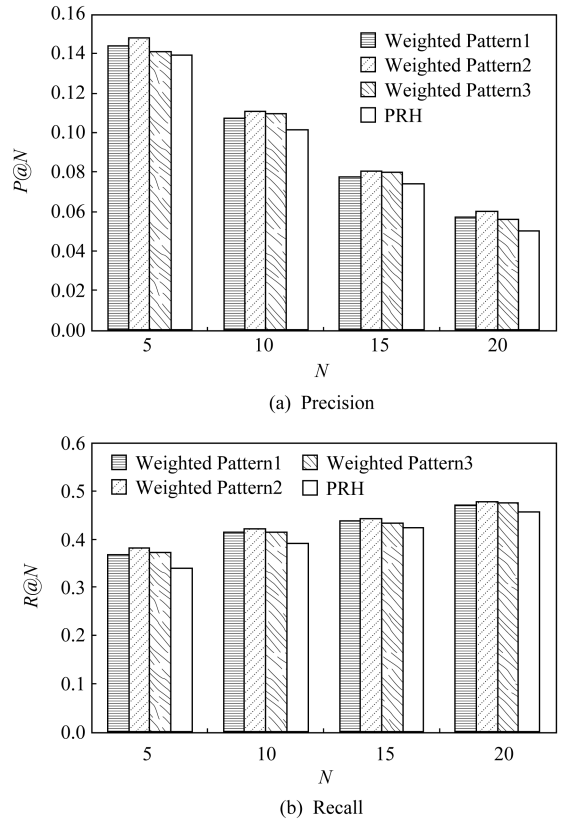


Fig. 10 Precision and recall based on optimized hyperedge weight when the number of location is N

图 10 超边加权优化在位置数 N 处的准确率和召回率

方式 2,从而得到一个在推荐准确性上更加出色的优化算法 oPRH,下文中将 oPRH 算法与 PRH 算法分别在 SF 数据集和 LA 数据集上进行了实验对比,测试结果如图 11～12 所示.从图 11～12 中可以看到,oPRH 在位置数 N 处的准确率和召回率都要高于 PRH,说明优化效果明显.

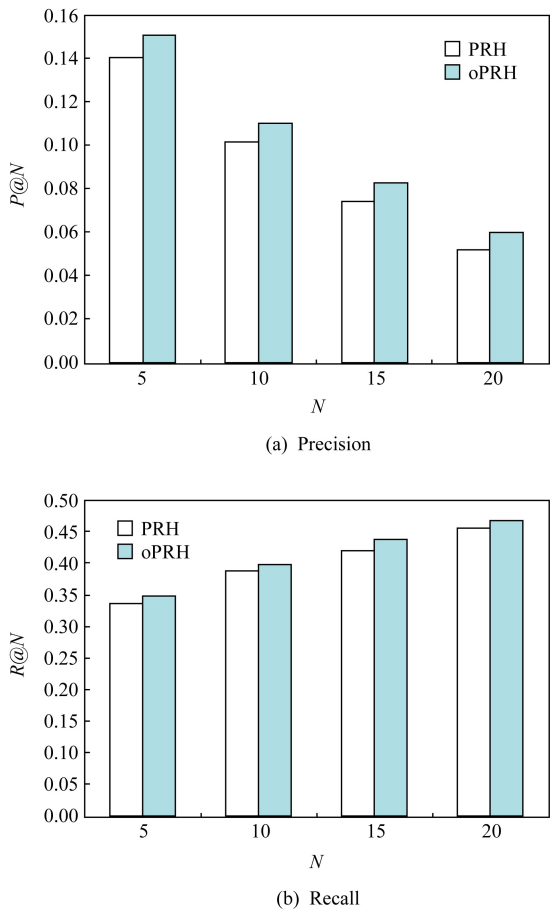
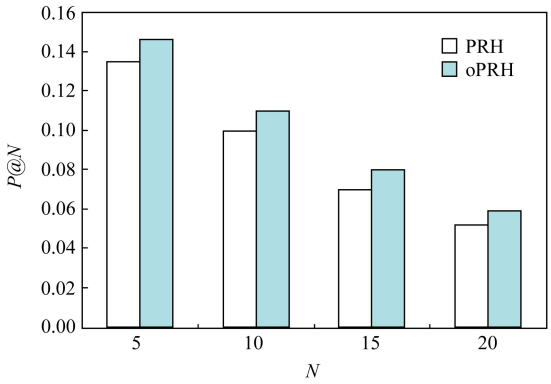


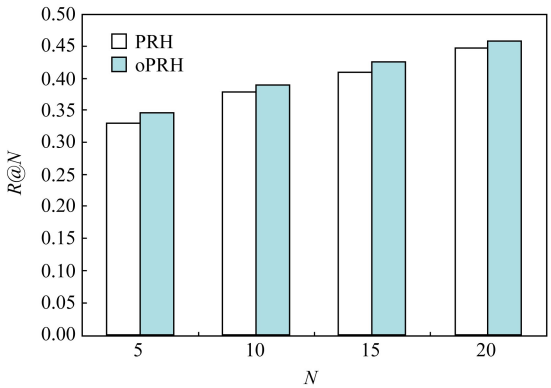
Fig. 11 Comparison results between oPRH and PRH based on SF data set

图 11 oPRH 和 PRH 在 SF 数据集上的性能比较

最后,针对 2 个不同数据集(SF 和 LA),比较了各算法的 $P_{M,A}$,$F1$,$G_{N,D,C}$ 指标,实验结果如表 11～14 所示.由表 11～12 可以观察到,在 2 个不同数据集下的 $P_{M,A}$ 值和 $F1$ 值,oPRH 算法都要高于 PRH 算法,说明优化措施确实可以提高算法推荐结果的准确性.另外,由表 13～14 可以观察到,PRH 算法和 oPRH 算法在位置数 N 处的 $G_{N,D,C}$ 值基本相同,这是由于 $G_{N,D,C}$ 是用来衡量最终推荐结果中的排序效果的,而本节中采取的优化方法只能提高最终推荐结果的准确性,但不能提高最终推荐结果中的排序效果,因而表 13 和表 14 中实验结果相差无几是合理的.



(a) Precision



(b) Recall

Fig. 12 Comparison results between oPRH and PRH based on LA data set

图 12 oPRH 和 PRH 在 LA 数据集上的性能比较

Table 11 $P_{M,A}$ and $F1$ of Two Algorithms Based on SF Data Set

表 11 2 种算法在 SF 数据集下的 $P_{M,A}$ 和 $F1$

Algorithm	$P_{M,A}$	$F1@5$	$F1@10$	$F1@15$	$F1@20$
PRH	0.239 1	0.199 8	0.165 4	0.129 8	0.102 3
oPRH	0.246 7*	0.206 7*	0.171 2*	0.135 4*	0.107 2*

Table 12 $P_{M,A}$ and $F1$ of Two Algorithms Based on LF Data Set

表 12 2 种算法在 LA 数据集下的 $P_{M,A}$ 和 $F1$

Algorithm	$P_{M,A}$	$F1@5$	$F1@10$	$F1@15$	$F1@20$
PRH	0.228 6	0.187 9	0.156 1	0.117 9	0.094 3
oPRH	0.234 1*	0.195 2*	0.164 3*	0.126 5*	0.101 1*

Table 13 $G_{N,D,C}$ of Two Algorithms Based on SF Data Set

表 13 2 种算法在 SF 数据集下的 $G_{N,D,C}$

Algorithm	$G_{N,D,C}@5$	$G_{N,D,C}@10$	$G_{N,D,C}@15$	$G_{N,D,C}@20$
HPR	0.509 8	0.432 7	0.453 5	0.473 3
oHPR	0.508 7	0.433 1	0.452 4	0.474 2

Table 14 $G_{N,D,C}$ of Two Algorithms Based on LA Data Set

表 14 2 种算法在 LA 数据集下的 $G_{N,D,C}$

Algorithm	$G_{N,D,C}@5$	$G_{N,D,C}@10$	$G_{N,D,C}@15$	$G_{N,D,C}@20$
PRH	0.499 7	0.427 8	0.448 1	0.466 4
oPRH	0.499 3	0.426 9	0.451 0	0.465 7

6 总 结

由于超图相比于普通图能更准确地刻画异构图中各实体间的关系,且信息不易丢失,本文首先开创性地将超图用于 EBSN 来解决异构实体网络的推荐问题,并在该超图模型下提出了面向 EBSN 的个性化推荐 PRH 算法.PRH 算法主要利用了流行排序本身具有准确体现对象间拓扑结构的特点,经过正则化运算给出了较优的初始推荐结果.本文又从改进查询向量设置方式和对不同类超边施以不同权重等方面,对 PRH 算法进行了优化,继而提出了 oPRH 算法,进一步提高了推荐质量.实验结果表明,利用超图解决 EBSN 下的推荐问题相较于普通图方法确实更有效,而优化的 oPRH 算法在推荐效果上也比 PRH 算法更为精准.

参 考 文 献

[1] Liu Xingjie, He Qi, Tian Yuanyuan, et al. Event-based social networks: Linking the online and offline social worlds [C] //Proc of the 18th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2012: 1032-1040

[2] Li Xiao, Cheng Xiang, Su Sen, et al. A hybrid collaborative filtering model for social influence prediction in event-based social networks [J]. Neurocomputing, 2017, 230: 197-209

[3] Dong Cailing, Shen Yilin, Zhou Bin, et al. I²Rec: An iterative and interactive recommendation system for event-based social networks [C] //Proc of Int Conf on Social Computing, Behavioral-Cultural Modeling, and Prediction and Behavior Representation in Modeling and Simulation. Berlin: Springer, 2016: 250-261

[4] Ding Hao, Yu Chenguang, Li Guangyu, et al. Event participation recommendation in event-based social networks [C] //Proc of Int Conf on Social Informatics. Berlin: Springer, 2016: 361-375

[5] Macedo A Q, Marinho L B, Santos R L T. Context-Aware event recommendation in event-based social networks [C] // Proc of the 9th ACM Conf on Recommender Systems. New York: ACM, 2015: 123-130

[6] Liao Guoqiong, Huang Xiaomei, Mao Mingsong, et al. Group event recommendation in event-based social networks considering unexperienced events [J]. IEEE Access, 2019, 7: 96650-96671

[7] Pham T A N, Li Xutao, Cong Gao, et al. A general graph-based model for recommendation in event-based social networks [C] //Proc of the 31st IEEE Int Conf on Data Engineering. Piscataway, NJ: IEEE, 2015: 567-578

[8] Li Keqian, Lu Wei, Bhagat S, et al. On social event organization [C] //Proc of the 20th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2014: 1206-1215

[9] She Jieying, Tong Yongxin, Chen Lei, et al. Confict-aware event-participant arrangement [C] //Proc of the 31st IEEE Int Conf on Data Engineering. Piscataway, NJ: IEEE, 2015: 735-746

[10] Liu Shenghao, Wang Bang, Xu Minghua. SERGE: Successive event recommendation based on graph entropy for event-based social networks [J]. IEEE Access, 2017, 6: 3020-3030

[11] Cao Jiuxin, Zhu Ziqing, Shi Liang, et al. Multi-feature based event recommendation in event-based social network [J]. International Journal of Computational Intelligence Systems, 2018, 11(1): 618-633

[12] Yu Yaxin, Zhang Haijun. Activity recommendation algorithm based on latent friendships in EBSN [J]. Computer Science, 2018, 45(3): 196-203 (in Chinese)
(于亚新, 张海军. EBSN 中基于潜在好友关系的活动推荐算法[J]. 计算机科学, 2018, 45(3): 196-203)

[13] Agarwal S. Ranking on graph data [C] //Proc of the 23rd Int Conf on Machine Learning. New York: ACM, 2006: 25-32

[14] Agarwal S, Branson K, Belongie S. Higher order learning with graphs [C] //Proc of the 23rd Int Conf on Machine Learning. New York: ACM, 2006: 17-24

[15] Sun Liang, Ji Shuiwang, Ye Jieping. Hypergraph spectral learning for multi-label classification [C] //Proc of the 14th ACM SIGKDD Int Conf on Knowledge Discovery and Data Mining. New York: ACM, 2008: 668-676

[16] Zhou Dengyong, Huang Jiayuan, Schölkopf B. Learning with hypergraphs: Clustering, classification, and embedding [C] //Advances in Neural Information Processing Systems. Cambridge, MA: The MIT Press, 2007: 1601-1608

[17] Chen Shouchun, Wang Fei, Zhang Changshui. Simultaneous heterogeneous data clustering based on higher order relationships [C] //Proc of the 7th IEEE Int Conf on Data Mining Workshops. Piscataway, NJ: IEEE, 2007: 387-392

[18] Zhou Dengyong, Weston J, Gretton A, et al. Ranking on data manifolds [C] //Advances in Neural Information Processing Systems. Cambridge, MA: The MIT Press, 2004: 169-176

[19] Guan Ziyu, Bu Jiajun, Mei Qiaozhu, et al. Personalized tag recommendation using graph-based ranking on multi-type interrelated objects [C] //Proc of the 32nd Int ACM SIGIR Conf on Research and Development in Information Retrieval. New York: ACM, 2009: 540-547

[20] Bulò S R, Pelillo M. A game-theoretic approach to hypergraph clustering [C] //Advances in Neural Information Processing Systems. Cambridge, MA: The MIT Press, 2009: 1571-1579

[21] Bu Jiajun, Tan Shulong, Chen Chun, et al. Music recommendation by unified hypergraph: Combining social media information and music content [C] //Proc of the 18th ACM Int Conf on Multimedia. New York: ACM, 2010: 391-400

[22] Li Lei, Li Tao. News recommendation via hypergraph learning: Encapsulation of user behavior and news content [C] //Proc of the 6th ACM Int Conf on Web Search and Data Mining. New York: ACM, 2013: 305-314

[23] Yu Jun, Rui Yong, Tang Yuanyan, et al. High-order distance-based multi view stochastic learning in image classification [J]. IEEE Transactions on Cybernetics, 2014, 44(12): 2431-2442

[24] Li Xi, Hu Weiming, Shen Chunhua, et al. Context-aware hypergraph construction for robust spectral clustering [J]. IEEE Transactions on Knowledge & Data Engineering, 2014, 26(10): 2588-2597

[25] Agarwal S, Branson K, Belongie S. Higher order learning with graphs [C] //Proc of the 23rd Int Conf on Machine Learning. New York: ACM, 2006: 17-24

[26] Rodriguez J A. On the Laplacian spectrum and walk-regular hypergraphs [J]. Linear and Multilinear Algebra, 2003, 51(3): 285-297

Yu Yaxin, born in 1971. PhD, associate professor, MS supervisor. Member of IEEE, ACM and CCF. Her main research interests include data mining and social network.

Zhang Wenchao, born in 1992. Master. His main research interests include data mining, machine learning, social network and hypergraph.

Li Zhenguo, born in 1994. Master. His main research interests include AR, SR and GAN.

Li Ying, born in 1994. Master. Her main research interests include blockchain, cloud computing.