软件学报 ISSN 1000-9825, CODEN RUXUEW
Journal of Software,2019,30(11):3397−3412 [doi: 10.13328/j.cnki.jos.005545]

E-mail: jos@iscas.ac.cn
http://www.jos.org.cn
Tel: +86-10-62562563

基于超图随机游走标签扩充的微博推荐方法*

马慧芳[1,2], 张 迪[1], 赵卫中[3], 史忠植[4]

[1](西北师范大学 计算机科学与工程学院,甘肃 兰州 730070)

[2](广西可信软件重点实验室(桂林电子科技大学),广西 桂林 541004)

[3](华中师范大学 计算机学院,湖北 武汉 430079)

[4](中国科学院 计算技术研究所 智能信息处理重点实验室,北京 100190)

通讯作者: 马慧芳, E-mail: mahuifang@yeah.net

摘 要: 向微博用户推荐对其有价值和感兴趣的内容,是改善用户体验的重要途径.通过分析微博特点以及现有微博推荐算法的缺陷,利用标签信息表征用户兴趣,提出一种结合标签扩充与标签概率相关性的微博推荐方法.首先,考虑到大部分微博用户未给自己添加任何标签或添加标签过少,视用户发布微博为超边,微博中的词视为超点来构建超图,并以一定的加权策略对超边和超点进行加权,通过在超图上随机游走,得到一定数量的关键词,对微博用户标签进行扩充;然后,采用相关性标签权重加权方案构建用户-标签矩阵,利用标签之间的概率相关性,构造标签相似性矩阵,对用户-标签矩阵进行更新,使该矩阵既包含用户兴趣信息,又包含标签与标签之间的关系.以新浪微博公开 API 抓取的微博信息作为实验数据进行了一系列的实验和分析,结果表明,该推荐算法具有较好的效果.

关键词: 超图;随机游走;标签扩充;概率相关性;用户-标签矩阵;微博推荐

中图法分类号: TP311

中文引用格式: 马慧芳,张迪,赵卫中,史忠植.基于超图随机游走标签扩充的微博推荐方法.软件学报,2019,30(11):3397−3412. http://www.jos.org.cn/1000-9825/5545.htm

英文引用格式: Ma HF, Zhang D, Zhao WZ, Shi ZZ. Microblog recommendation method based on hypergraph random walk tag extension. Ruan Jian Xue Bao/Journal of Software, 2019,30(11):3397−3412 (in Chinese). http://www.jos.org.cn/1000-9825/5545.htm

Microblog Recommendation Method Based on Hypergraph Random Walk Tag Extension

MA Hui-Fang[1,2], ZHANG Di[1], ZHAO Wei-Zhong[3], SHI Zhong-Zhi[4]

[1](College of Computer Science and Engineering, Northwest Normal University, Lanzhou 730070, China)

[2](Guilin University of Electronic Technology (Guangxi Key Laboratory of Trusted Software), Guilin 541004, China)

[3](School of Computer Science, Central China Normal University, Wuhan 430079, China)

[4](Key Laboratory of Intelligent Information Processing, Institute of Computing Technology, Chinese Academy of Sciences, Beijing 100190, China)

Abstract: Recommending valuable and interesting contents for microblog users is an important way to improve the user experience. In this study, tags are considered as the users' interests and a microblog recommendation method based on hypergraph random walk tag Extension and tag probability correlation is proposed via the analysis of characteristics and the existing limitations of microblog recommendation algorithm. Firstly, microblogs are considered as hyperedges, while each term is taken as the hypervertex, and the

* 基金项目: 国家自然科学基金(61762078, 61363058, 61762079); 中国科学院计算技术研究所智能信息处理重点实验室开放基金(IIP2014-4); 广西可信软件重点实验室研究课题(kx201910)

Foundation item: National Natural Science Foundation of China (61762078, 61363058, 61762079); Open Program of Key Laboratory of Intelligent Information Processing Institute of Computing Technology, Chinese Academy of Sciences (IIP2014-4); Research Program of Guangxi Key Laboratory of Trusted Software (kx201910)

收稿时间: 2017-01-08; 修改时间: 2017-06-05, 2017-08-25, 2011-11-01; 采用时间: 2018-01-06

weighting strategies for both hyperedges and hypervertexes are established. A random walk is conducted on the hypergraph to obtain a number of keywords for the expansion of microblog users. And then the weight of the tag for each user is enhanced based on the relevance weighting scheme and the user tag matrix can be constructed. Probability correlation between tags is calculated to construct the tag similarity matrix, which can be used to update the matrix is updated using the label similarity matrix, which contains both the user interest information and the relationship between tags and tags. Experimental results show that the algorithm is effective in microblog recommendation.

Key words: hypergraph; random walk; label expansion; probability correlation; user-tag matrix; microblog recommendation

随着网络的普及发展和电子产品的不断更新,全球互联网已进入互联互通的春天.在网络融合的背景下,以微博为代表的各种新技术、新应用相继涌现,其通过文字、图片、视频等多媒体形式实现信息发布、互动交流,兼具社交属性与媒体属性,快速聚集了庞大的用户群体.这些社交应用的发展给人们生产生活方式以及信息传播途径都带来了翻天覆地的变化.

微博平台中的内容涉及很多方面,从用户的日常生活到国内外的时事新闻,且微博用户数量庞大,每分钟就有几十万条微博发出,微博已经成为人们日常生活中不可缺少的信息来源和交流平台,它是 Web 2.0 的一个重要组成部分.微博具有两个典型的特点.

- 第一,它能够被所有的网络用户创建,这些用户不分背景、身份,都可以创建自己的微博.例如,工人、学生、教师等各行各业的人都可以参与进来,发表自己的微博.
- 其次,微博用户发表的内容质量参差不齐,描述语言丰富多彩,有书面的、口语的以及时下流行的网络语言.而这些用户产生的大量信息中可能含有高质量的内容,也可能含有低质量的垃圾信息[1].

通常情况下,微博平台使用时间线排序的方式向微博用户展示其关注用户的信息流,即将用户关注的最新信息展示在个人信息页面的最顶部.用户要想遍历微博信息并且从中获取自己需要的信息,不但需要花费大量的时间,而且还不一定能够找到自己感兴趣的内容,随之产生的问题就是用户体验下降、用户流失等.因而,如何从海量的微博数据中挖掘出用户感兴趣的内容并精准地推荐给目标用户,已经成为微博推荐相关研究的热点.

本文提出一种新的微博推荐框架,以向微博用户推荐高质量的微博.该框架包括微博用户标签扩充、微博用户兴趣建模以及微博推荐这 3 部分:微博用户标签扩充的目的是抽取用户微博文本中的关键词,增加用户标签数量,这些标签可以用来为用户推荐高质量的微博;微博用户兴趣建模的目的是为了使系统“在对的时间以对的方式做对的事情”,也就是为了给用户带来最好的体验[2].

本文的贡献有以下几点.

(1) 提出了一种微博标签扩充方案.该方案将微博视为超边,微博中的词视为超点构建超图,并对超边和超点进行加权,通过在超图上随机游走,得到一定数量的关键词对微博用户标签进行扩充.

(2) 设计了一种微博用户兴趣表示模型,采用相关性标签权重加权方案,构建用户-标签矩阵,利用标签之间的概率相关性,构造标签相似性矩阵对用户-标签矩阵更新,解决了用户标签矩阵稀疏的问题.

(3) 利用真实数据集验证了所提出的微博推荐方法的有效性和高效性.

本文在第 1 节介绍相关工作,包括微博用户兴趣模型以及微博个性化推荐的相关理论技术.第 2 节介绍本文提出的基于超图随机游走标签扩充的方法.第 3 节介绍本文提出的基于标签概率相关性的微博推荐方法.第 4 节中,通过实验数据集来验证算法的性能.最后总结全文,并对未来的研究方向进行展望.

1 相关工作

实现微博个性化的信息推荐[3−5],理解用户的兴趣或需求是前提和关键.目前,在微博背景下发掘用户兴趣模型已经有了一些研究工作,一部分研究人员从微博用户发布的微博内容中抽取若干个代表用户个性化特征的关键词作为对用户兴趣爱好的描述[6,7].由于微博内容形式多样,既有即时所见所闻的情感表达,又有转发的新闻时事,导致微博内容具有随意性、碎片化等特点.以上方法提取的关键词并不能很好地表征用户的兴趣爱好.另一部分研究人员利用外部知识库对微博语义进行扩充[8,9],之后引入概率主题模型,更好地表达文本的语

义.还有一部分研究人员将用户所发的微博整合成一个长的文本文档,然后利用潜在狄利克雷分配模型发现用户潜在的兴趣[10].这种方法虽然在实际应用中能够有效地向用户推荐微博,但是仍然不能满足用户的个性化需求.

然而,以上这些方法仅仅考虑了从微博文本的角度挖掘用户兴趣,事实上,除了微博内容能够反映用户兴趣外,用户自己标注的个性化标签也能体现用户的喜好特征.标签是用户综合自己的工作性质、年龄群体以及兴趣爱好等因素的关键词集合,涵盖了丰富且价值很大的信息,其对表征用户用户兴趣特点和关注领域具有不可估量的作用.然而,现有关于微博用户标签的研究相对较少.已有的工作仅仅只是针对用户标注标签的行为[11],标签内容的特点以及标签与其他用户信息(如微博内容、关注关系)之间的联系进行研究[12].针对微博推荐的研究,尽管已有研究者采用标签来进行微博推荐[13−15],但是却并未充分考虑利用微博内容对用户标签进行扩充,更是很少有人将之与标签间的概率相关性结合进行微博推荐,而这将是本文所研究的主要问题.

基于对以上研究的分析可知,要想更加准确地研究个性化微博推荐,必须将微博内容与用户标签进行结合.本文提出一种结合超图随机游走标签扩充与标签概率相关性的微博推荐方法.具体来说,首先,该方法将微博视为超边,微博中的词视为超点来构建超图,通过在超图上随机游走,得到一定数量的关键词对用户标签进行扩充;然后采用相关性标签权重加权方案,构建用户-标签矩阵,利用标签之间的概率相关性,构造标签相似性矩阵对用户-标签矩阵更新.上述方法有效解决了用户标签矩阵的稀疏问题,不仅使更新后的矩阵包含了微博用户的兴趣信息,而且还融合了标签标签间的相关性关系,如图 1 所示.与未考虑微博内容和标签与标签关系的推荐算法相比,本文提出的微博用户兴趣表示模型更能表征用户的兴趣特征,微博推荐性能得到提升.

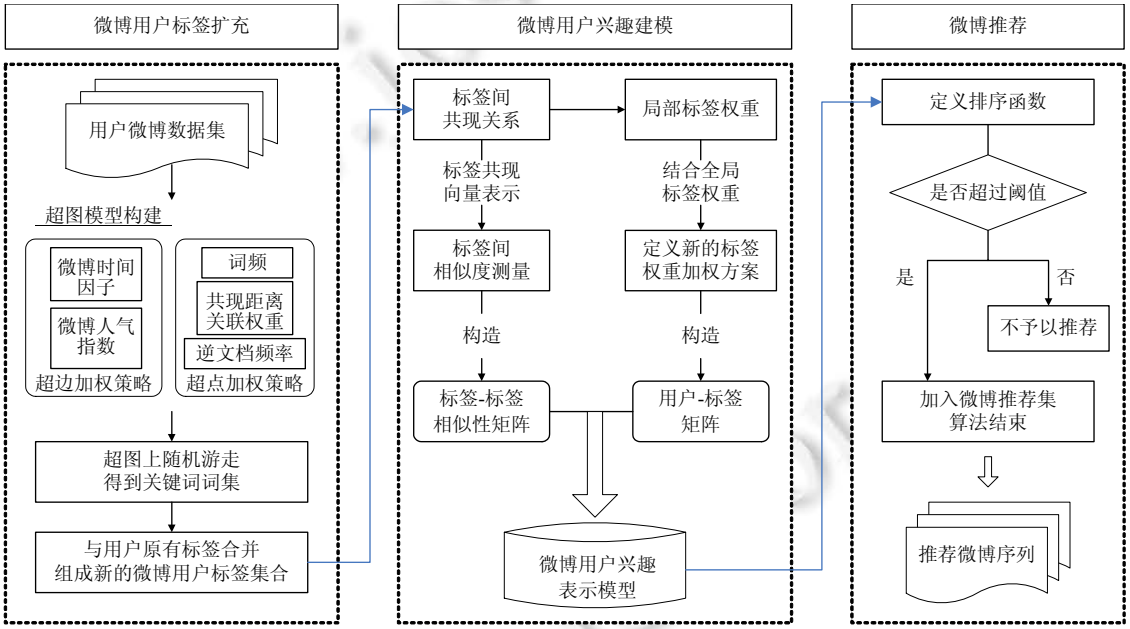


Fig.1 Research framework for microblog recommendation based on tag extension and tag probability correlation

图 1 基于标签扩充与标签概率相关性的微博推荐研究框架

2 基于超图随机游走的标签扩充

超图学习是对普通图学习的泛化和扩展,在超图中一条边可以包含任意数量的顶点,则超边是一条包含该超边中所有顶点的简单闭合曲线.因而相对于普通图而言,超图具备描述多元关系的能力,这使得超图具有更好

的性能表现.假设 $V=(v_1,v_2,\ldots,v_n)$是一个有限集,$e_i\neq\varnothing(i=1,2,\ldots,m)$和$\bigcup_{i=1}^{m}e_i=V$, 则称 $HG(V,E)$是一个普通超图,其中,V中的元素 $v_1,v_2,\ldots,v_n$称为超图的顶点,即超点;$E=\{e_1,e_2,\ldots,e_i,\ldots,e_m\}$为超图的边集合,$E$ 中的元素 $e_i=\{v_{i1},v_{i2},\ldots,v_{ij}\}(1\leqslant j\leqslant n)$称为超图 $HG(V,E)$中的超边.一个普通超图可以用指示矩阵 $H_{|V|\times|E|}$来表示.若 $v\in e$,矩阵中的元素 $h(v,e)=1$;否则,$h(v,e)=0$.从定义可知,当且仅当超边 E 中的任一超边e包含 2 个节点时,普通超图将退化为普通图,故可将普通图看作是超图的特例.

本文所提出的基于超图随机游走的微博关键词提取算法,首先,根据微博用户所发布的微博内容为每一个微博用户都构建一个超图;其次,从微博的时间因子和微博的人气指数这两方面对超边加权,通过词语之间的关联度以及词语在特定微博下的共现距离对超点加权.在构建超图模型的过程中,超边是不同的微博文本,超点是微博中不同词语.本质上,该模型将每一篇微博视为一个由不同关键词所组成的词袋模型.而这些微博的集合即为本文所定义的词汇超图.

2.1 加权超图模型的构建

给定微博用户集合 $U=\{u_1,u_2,\ldots,u_i,\ldots,u_N\}$,$N$ 为用户的个数.用户 u_i 所发布的微博集合为 $D_i=\{d_{i1},d_{i2},\ldots,d_{iM_k},\ldots,d_{iM_i}\}$, M_i 为用户 u_i 所发布的微博个数.其中,$1\leqslant i\leqslant N$,则所有用户发布的微博集合为 $D=\bigcup_{i=1}^{N}D_i$;用户 u_i 的微博集合 D_i 中的词集为 $L_i=\{l_{i1},l_{i2},\ldots,l_{im_i}\}$, m_i 为词集 L_i 中词语的个数且 $m_i>>M_i$,则所有用户的微博词语集合为 $L=\bigcup_{i=1}^{N}L_i$; D_i 进一步被划分为 D_i^{train} 和 D_i^{test}, 且 $D_i^{train}\cap D_i^{test}=\varnothing$; L_i 进一步被划分为 L_i^{train} 和 L_i^{test}, 它们分别与 D_i^{train} 和 D_i^{test} 相对应.本文标签扩充和用户兴趣建模这两个部分都是在 D_i^{train} 上进行,而微博推荐部分则是在 D_i^{test} 上进行.

设 $WHG(V,E,w(e),w(v_i,e_i))$为一个加权超图,其中,$w(e):e\rightarrow\mathbf{R}^+$代表超边 e 的权重,$w(v_i,e_i):(v_i,e_i)\rightarrow\mathbf{R}^+$代表超点 v_i 在特定超边 e_i 上的权重.一个带权重的超图指示矩阵 $H_{w|V|\times|E|}$中的元素定义如下.

$$h_w(v,e)=\begin{cases}w(v_e), & v\in e\\ 0, & v\notin e\end{cases} \tag{1}$$

在加权超图中,超点的度 $d(v)$与超边的度 $d(e)$定义如下.

$$d(v)=\sum_{e\in E}w(e)\times h(v,e) \tag{2}$$

$$d(e)=\sum_{v\in V}w(v_e)\times h(v,e) \tag{3}$$

超点的度被定义为其所在超边权重的和,超边的度被定义为在该超边上所有超点的权重和.本节将详细介绍对超边和超点加权的具体策略.值得注意的是,下文中提及到的 d_{iM_k} 与 e 同义,皆代表某条特定的超边.

2.1.1 超边加权策略

对微博用户而言,微博内容所反映出的用户兴趣是随着时间的变化而变化的.提出微博时间因子来表征随着微博发表时间的增长,其内容对用户重要性的变化.则用户 u_i 的某一篇微博 d_{iM_k} 的时间因子 $F_{time}(d_{iM_k})$ 如公式(4)所示.

$$F_{time}(d_{iM_k})=Q^{\frac{cur_{time}-d_{iM_k\text{-}time}}{TS_{u_i}}} \tag{4}$$

其中,cur_{time} 是当前时间, $d_{iM_k\text{-}time}$ 是微博 d_{iM_k} 的发布时间, TS_{u_i} 表示用户 u_i 从发布第 1 篇微博到目前的时间跨度.离当前时间越近的微博,其时间因子越大,这条微博就越能表征用户最近的兴趣爱好.

Q 是衰减率参数,取值范围为 $0<Q<1$.此外,由于 $0<Q<1$, $F_{time}(d_{iM_k})$ 是一个单调递减函数.当 Q 趋近于 0 时,微博的时间因子越小;当 Q 趋近于 1 时,微博的时间因子越大.本文中,设定 Q 的值为 0.5.

显然,一篇微博的评论数、转发数以及点赞数越多,这篇微博就越能体现微博用户的兴趣爱好.因此,提出微博人气指数(microblog popularity index,简称 MPI)来表征微博对用户的重要程度.微博人气指数由微博的评论数 $S_{comment}$、微博的转发数 $S_{forward}$ 以及微博的被点赞数 S_{like} 等子属性衡量,则用户 u_i 的某一篇微博 d_{iM_k} 的人气

指数 $MPI(d_{iM_k})$ 如公式(5)所示.

$$MPI(d_{iM_k})=\lambda_1\frac{s_{comment\text{-}d_{iM_k}}+1}{\sum_{M_j}s_{comment\text{-}d_{iM_j}}+1}+\lambda_2\frac{s_{forward\text{-}d_{iM_k}}+1}{\sum_{M_j}s_{forward\text{-}d_{iM_j}}+1}+\lambda_3\frac{s_{like\text{-}d_{iM_k}}+1}{\sum_{M_j}s_{like\text{-}d_{iM_j}}+1} \tag{5}$$

其中,λ_1~λ_3 分别表示相应子属性的权重比例.本文中,假定微博的评论数、转发数和点赞数对微博来说同等重要,它们的和为 1.

通过结合微博时间因子和微博人气指数计算用户 u_i 的某篇微博 d_{iM_k} 的权重,如公式(6)所示.

$$w(d_{iM_k})=\alpha F_{time}(d_{iM_k})+(1-\alpha)MPI(d_{iM_k}) \tag{6}$$

其中,α为值域在(0,1)之间的平滑因子,用它来调节 F_{time} 与 MPI 的比重.α的值越大,代表更注重微博的时效性;相反,则更注重微博的人气指数.后续实验中,将通过分析α在不同取值下的实验结果来确定α的值.最终的 $w(d_{iM_k})$ 表示用户某一条超边的权重.

2.1.2 超点加权策略

在本文的超图模型中,超点是微博文本中的不同词语,通过计算词语之间的共现度[16]、关联度以及在特定微博中的共现距离对超点加权.表 1 为加权过程中所用到的符号定义.

Table 1 Notations used for hyper-vertex weighting

表 1 超点加权阶段各符号的定义

符号	定义	符号	定义
v_i	超图中的某个超点	$co(v_i,v_j)$	v_i 和 v_j 的共现度
$co_{d_{iM_k}}(v_i,v_j)$	在微博 d_{iM_k} 中,v_i 和 v_j 的共现度	$Rel(v_i,v_j)$	v_i 和 v_j 的关联度
$uRel_{v_i}(v_i,v_j)$	v_i 的单边关联度	$cow(v_i,d_{iM_k})$	词语 v_i 在微博 d_{iM_k} 中的关联权重
$\lvert d_{iM_k}\rvert$	微博 d_{iM_k} 中词语的个数	$df(v_i)$	包含词 v_i 的微博个数

给定超点集合,对于特定微博 d_{iM_k} 而言,v_i 与 v_j 的共现度可通过公式(7)求出.

$$co_{d_{iM_k}}(v_i,v_j)=\begin{cases}n_{d_{iM_k}}\times e^{-dist_{d_{iM_k}}(v_i,v_j)}, & v_i\in d_{iM_k}\text{且}v_j\in d_{iM_k}\\ 0, & v_i\notin d_{iM_k}\text{或}v_j\notin d_{iM_k}\end{cases} \tag{7}$$

公式(7)考虑了 v_i 与 v_j 的共现距离.其中,共现距离 $dist_{d_{iM_k}}(v_i,v_j)$ 是词语 v_i 与 v_j 共同出现时,中间间隔的词语个数; $n_{d_{iM_k}}$ 是 v_i 与 v_j 在微博 d_{iM_k} 中共现的次数.

根据在特定微博 d_{iM_k} 中 v_i 与 v_j 的共现度,则可求得在所有微博上词语 v_i 与 v_j 的共现度,如公式(8)所示.

$$co(v_i,v_j)=\sum_{d_{iM_k}\in D}co_{d_{iM_k}}(v_i,v_j) \tag{8}$$

基于 v_i 与 v_j 的共现度 $co(v_i,v_j)$,可进一步求解 v_i 与 v_j 的关联度.由于该关联度是非对称的,因此分别求解 v_i 与 v_j 的单边关联度,如公式(8)所示.

$$uRel(v_i\mid v_j)=\frac{co(v_i,v_j)}{\sum_{q=1}^{|L_i|}co(v_i,v_q)}\times\log_2\frac{|L_i|}{N_{nei}(v_j)} \tag{9}$$

公式(9)可以求出 v_i 的单边关联度,公式的前半部分体现了词语 v_i 出现时,词语 v_j 出现的概率.$N_{nei}(v_j)$表示与词语 v_j 共现过的词语的个数,该值越小越好.这是因为,若 v_j 相对于 v_i 而言是重要的,则 v_j 必然很少与除 v_i 之外的其他词共同出现.公式的后半部分(idf 部分)惩罚了那些和很多词都共现过的 v_j.同理,可求出 v_j 的单边关联度.

v_i 与 v_j 的关联度 $Rel(v_i,v_j)$实质上就是 v_i 与 v_j 的单边关联度的均值.

传统的加权方式所求得的词频对微博短文本而言意义不大,因此本文考虑了词语的关联权重.关联权重反映了在给定微博中词语的主题指示性,若词语 v_i 的关联权重越高,则意味着当 v_i 出现时,其他顶点随之出现的概率也就越高.换言之,与 v_i 具有强关联的 v_j 越多,v_i 与微博主题的相关性就越大.给定一个含有$|L_i|$个超点的超图后,每个超点都会有确定的初始权重 $iw(v_i)$,则超点 v_i 在超边 d_{iM_k} 中的关联权重定义如公式(10)所示.

$$cow(v_i,d_{iM_k})=iw(v_i)+\frac{\sum_{j=1}^{|d_{iM_k}|}iw(v_j)\times Rel(v_i,v_j)}{|d_{iM_k}|} \tag{10}$$

其中,$iw(v_i)$代表超点 v_i 的初始权重,其值是 v_i 在微博 d_{iM_k} 中的词频.

超点的权重不仅应该考虑该点与其他超点的关联性权重,而且还应该考虑词语对微博的标识度.即某个词语在这篇微博中出现,但是却很少在其他微博中出现,则认为此词语对该微博是重要的.所以综合词语的关联性权重和全局统计权重对其进行加权,超点 v_i 的权重定义如公式(11)所示.

$$w(v_i,d_{iM_k})=cow(v_i,d_{iM_k})\times idf(v_i)=cow(v_i,d_{iM_k})\times\log_2\frac{|D_i|}{df(v_i)} \tag{11}$$

2.2 标签扩充方法

为了给超图中的顶点排序,Bellaachia 等人将随机游走的方法在超图上进行了推广[17].本文中,基于加权超图的随机游走过程如下:首先选定起始超点 u,根据超边权重 $w(e)$选择一条包含当前超点 u 的特定超边 e;然后,在已经选中的这条超边中,根据超点权重选择转移顶点 v.设 $\boldsymbol{P}$ 为随机游走的转移概率矩阵,其计算方法如公式(12)所示.

$$\boldsymbol{P}(u,v)=\sum_{e\in E}w(e)\times\frac{h(u,e)}{\sum_{\hat{e}\in E}w(\hat{e})}\times\frac{h_w(v,e)}{\sum_{\hat{v}\in e}h_w(\hat{v},e)} \tag{12}$$

或者直接用矩阵的形式来表示转移概率矩阵 $\boldsymbol{P}$.

$$\boldsymbol{P}=\boldsymbol{D}_v^{-1}\boldsymbol{H}\boldsymbol{W}_e\boldsymbol{D}_e^{-1}\boldsymbol{H}_w^T.$$

其中,$h_w(v,e)$是目的顶点 v 在超边 e 中的权重;$\boldsymbol{D}_v$ 是超点度的对角线矩阵,矩阵中元素的计算方法如公式(2)所示;$\boldsymbol{H}$ 是普通超图的指示矩阵;$\boldsymbol{W}_e$ 为超边权重的对角线矩阵;$\boldsymbol{D}_e$ 是超边度的对角线矩阵,矩阵中元素的计算方法如公式(3)所示;$\boldsymbol{H}_w$ 是加权超图的指示矩阵.值得注意的是,计算所得的转移概率矩阵 $\boldsymbol{P}$ 是行归一化后的结果.

随机游走过程刚开始时,将初始分布向量 $\boldsymbol{v}^0\in R^{|V|\times 1}$ 视为等概率的,即其每一个元素值都为 $1/|V|$,这些元素之和为 1.首先,将转移矩阵 $\boldsymbol{P}$ 的转置矩阵 $\boldsymbol{P}^T$(转置的目的是实现列归一化)与初始向量 $\boldsymbol{v}_0$ 相乘,得到 $\boldsymbol{v}^1=\boldsymbol{P}^T\boldsymbol{v}^0$;然后,对此过程进行迭代,直至向量 $\boldsymbol{v}$ 不再变化.将概率分布向量 $\boldsymbol{v}$ 与矩阵 $\boldsymbol{P}^T$ 相乘可以得到下一步的概率分布 $\boldsymbol{x}=\boldsymbol{P}^T\boldsymbol{v}^0$.原因如下:设 x_i 为当前位于节点 i 的概率 $x_i=\sum_j p_{ij}v_j$,其中,v_j 代表预设节点,p_{ij} 为从节点 j 跳转到节点 i 的概率.

随机游走在经过 n 步之后,若所有节点已被遍历,则概率分布向量 $\boldsymbol{v}$ 不再发生变化.随机游走遍历所有节点需满足以下两个条件:马尔可夫链是不可约的和非周期性的.为了保证这两个条件,在此使用 PageRank 算法来实现随机游走过程,该算法加入了心灵转移的思想.所谓心灵转移,就是在任何一条超边的超点都有可能以一个较小的概率瞬间转移到另外一条超边上.当然,这两条超点可能不存在连边,因此不可能真的直接转移过去.本文中这个小概率用阻尼因子β来表示,如公式(13)所示.

$$\boldsymbol{v}^{i+1}=(1-\beta)\boldsymbol{P}^T\boldsymbol{v}^i+\beta\boldsymbol{e}/n \tag{13}$$

其中,n 是超图中超点的个数,$\boldsymbol{e}\in R^{n\times 1}$ 是超图中长度为 n 的单位向量.$(1-\beta)\boldsymbol{P}^T\boldsymbol{v}$ 表示随机游走将从当前超点选择一条超边跳转到另外一个超点上.$\beta\boldsymbol{e}/n$ 表示随机游走将以β/n 的概率跳转到任意的其他超点.根据经验[18,19],本文中β的取值为 0.15.

当随机游走的过程停止,即向量 $\boldsymbol{v}$ 不再发生变化时,对向量 $\boldsymbol{v}$ 中各顶点按权重由大到小依次排序.通常,选取对用户重要程度最高的 Top-Q 个词项作为用户的标签,与用户原有的标签合并,组成新的微博用户标签集合 $T_i=\{t_{i1},t_{i2},...,t_{in_i}\}$, n_i 代表用户 u_i 最终的标签个数.所有标签的集合为 $T=\bigcup_{i=1}^{N}T_i$, 且标签集合 T 中标签的总数为 K.至此,微博用户的标签得到扩充,接下来将采用标签概率相关性对微博进行推荐.

3 基于标签概率相关性的微博推荐

3.1 概率相关性构建标签相似性矩阵

纵览用户标签集合,标签与标签之间并不是完全独立的,它们彼此间存在着一定程度的关联性,这种潜在的关联性使得每个标签对不同用户的重要性是不一样的.此外,标签的多义性常常使得标签在表征用户特征时出现歧义,例如某用户的部分标签集{“苹果”,“美食”,“果粉”,…},在该标签集中,标签“苹果”就具有多义性,无法确定该标签表示的是一种水果还是一种通信设备.但是通过计算标签间的概率相关性[20],可以得到用户的倾向性.

3.1.1 标签概率相关性

从整体观测,假如任意两个标签经常一同被用户标记,则可以推测这两个标签之间有很大概率的关联性.从局部观测,假如因某一标签被用户标记后,另一标签被用户标记的概率也很大,那么推测这两个标签存在较强的共现关系,定义如公式(14)所示.

$$p(t_i \mid t_j)=\frac{p(t_i t_j)}{p(t_j)} \tag{14}$$

其中,分母代表标签 t_j 出现的概率,分子代表标签 t_i 和标签 t_j 共同出现的概率,即 $p(t_it_j)\approx uf(t_it_j)/N$,$uf(t_it_j)$表是同时标注标签 t_i 和标签 t_j 的用户数.

从公式(14)可以看出,标签 t_i 与标签 t_j 之间的条件概率是一个非对称的值.然而,标签与标签之间的相似性是对称关系.因此,对标签之间的共现关系进行改进,具体办法如公式(15)所示.

$$cor(t_i,t_j)=p(t_i|t_j)\times p(t_j|t_i) \tag{15}$$

综合公式(14)和公式(15),标签 t_i 与标签 t_j 之间的概率相关性被改写为

$$cor(t_i,t_j)=\frac{p(t_i t_j)^2}{p(t_i)\times p(t_j)} \tag{16}$$

3.1.2 标签相似性矩阵

一般通过向量空间模型构建用户-标签矩阵来表征用户,矩阵中的每一个元素 w_{ij} 是用户 u_i 在第 j 个标签上的权重,标签也可以被视为是一个用户向量$[u_{i,1},u_{i,2},\ldots,u_{i,N}]$.对于用户标签之间相似性的测量,传统做法是计算用户向量的相似程度.但是,受到用户向量存在极度稀疏情况的限制,传统方法并不能很好地测度标签间的相似性.因此,本文通过标签相关性向量来表征标签.

利用 3.1.1 节计算得到的标签概率相关性,微博标签集合中的每一个标签 t_i 都可以被表示成标签相关性向量$[t_{i,1},t_{i,2},\ldots,t_{i,n},\ldots,t_{i,K}]$,$t_{i,n}$ 是标签 t_i 和 t_n 之间的概率相关性,定义如公式(17)所示.

$$\boldsymbol{t}_i=[cor(t_i,t_1),cor(t_i,t_2),\ldots,cor(t_i,t_n),\ldots,cor(t_i,t_K)] \tag{17}$$

采用余弦相似度计算标签间的相似程度,如公式(18)所示.

$$sim(\boldsymbol{t}_1,\boldsymbol{t}_2)=\frac{\boldsymbol{t}_1\cdot\boldsymbol{t}_2}{\|\boldsymbol{t}_1\|\times\|\boldsymbol{t}_2\|}=\frac{\sum_{i=1}^{K}cor(t_1,t_i)cor(t_2,t_i)}{\sqrt{\sum_{i=1}^{K}cor(t_1,t_i)^2}\sqrt{\sum_{i=1}^{K}cor(t_2,t_i)^2}} \tag{18}$$

由标签相关性向量构造标签相似性矩阵 $\boldsymbol{S}$,S_{ij} 表示标签向量 $\boldsymbol{t}_i$ 和标签向量 $\boldsymbol{t}_j$ 的余弦相似度,其计算公式如公式(19)所示.

$$S_{ij}=\begin{cases}1, & i=j\\ sim(\boldsymbol{t}_i,\boldsymbol{t}_j), & i\neq j\end{cases} \tag{19}$$

矩阵中元素的取值范围在(0,1]之间,S_{ij} 的取值等于 1 时,表示两个标签是一样的,S_{ij} 的取值越接近 1,则表明标签间的关联性越显著.在用户标签集合中,标签之间都会存在一定程度的关联性.因此,S_{ij} 的取值是非零的.

3.2 微博用户兴趣表示与推荐算法

标签作为用户对身份特征、兴趣爱好等综合描述的简单载体,其包含的丰富信息对构建精准的用户画像具有重要作用.拥有相似标签越多的用户,其博文类型也越相似.本节从相关性权重从发,通过构建精准的用户兴

趣模型矩阵,进而进行个性化内容推荐.

3.2.1 用户标签权重

微博用户为自己标注的标签在整体上表现出幂率分布,即少量具有代表性的标签会时常被标注,而其他个性化的标签通常很少被标注,从而导致在传统的标签权重加权方案中大多数标签的频率为 1.针对该情况,利用标签间的概率相关性,提出一种新的标签权重加权方案相关性权重.

微博用户在为自身加注标签后,这些被标注的标签之间就存在着一定的关联性.假如用户的某一标签与其他任一标签都具有较强的关联度,则该标签对用户具有较强的标识度,定义如公式(20)所示.

$$cow(u_i,t_k)=\frac{\sum_{t_j\in u_i} cor(t_k,t_j)}{|u_i|} \tag{20}$$

其中,$|u_i|$表示用户 u_i 的标签个数,$cor(t_k,t_j)$表示标签 t_k 和 t_j 的概率相关性.标签的相关性权重是标签在标签集合中重要程度的体现.标签具有的权重越高,其对用户兴趣爱好的描述能力就越好.

公式(20)仅仅从标签的局部特征出发计算出了标签对用户的权重.一个全面的标签权重不仅要从局部考虑到其自身与其他标签的关系,而且还需要从整个微博集合上思考标签对用户的标识性,取名为逆用户频率 IUF(inverse user frequency).具体思路是采用数据集中的用户总数与加注某标签的用户数的比值并取其对数,定义如公式(21)所示.

$$iuf(t_k)=\log_2\left(\frac{N}{uf(t_k)}+1\right) \tag{21}$$

其中,$uf(t_k)$表示拥有标签 t_k 的用户数.综合标签的相关性权重和 iuf 值,则用户 u_i 中标签 t_k 的权重定义如公式(22)所示.

$$w_{ik}=cow(u_i,t_k)\times iuf(t_k) \tag{22}$$

3.2.2 用户标签矩阵

针对用户 u_i 构造一个标签权重向量 $\boldsymbol{V}_i=(w_{i1},w_{i2},\ldots,w_{iK})$来存储标签的权重[21],基于以上用户权重向量,构建 $N\times K$ 的用户-标签矩阵 $\boldsymbol{M}$.

由于用户-标签矩阵的列数为所有待加标用户的标签集合,并且该集合中的标签并不可能被所有用户标注,故该矩阵存在高维、稀疏的问题,并不能精准表征用户兴趣.考虑标签间的概率相关性,构建标签-标签相似度矩阵,以此更新该用户-标签矩阵,不仅可以解决原始矩阵的局限,而且还包含丰富的语义信息.公式(23)是更新后的用户-标签矩阵.

$$\boldsymbol{M}'=\boldsymbol{M}\times\boldsymbol{S}=\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1K}\\ w_{21} & w_{22} & \cdots & w_{2K}\\ \vdots & \vdots & \vdots & \vdots\\ w_{N1} & w_{N2} & \cdots & w_{NK}\end{bmatrix}\begin{bmatrix} sim_{11} & sim_{12} & \cdots & sim_{1K}\\ sim_{21} & sim_{22} & \cdots & sim_{2K}\\ \vdots & \vdots & \vdots & \vdots\\ sim_{K1} & sim_{K2} & \cdots & sim_{KK}\end{bmatrix} \tag{23}$$

其中,$\boldsymbol{M}$ 是初始的用户-标签矩阵,$\boldsymbol{S}$ 是标签相似性矩阵,$\boldsymbol{M}'$是更新后的用户-标签矩阵,该矩阵可以更好地表示用户兴趣.为了更好的解释更新后的矩阵稀疏问题得到缓解的问题,分解矩阵 $\boldsymbol{S}$ 如下.

$$\boldsymbol{M}'=\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1K}\\ w_{21} & w_{22} & \cdots & w_{2K}\\ \vdots & \vdots & \vdots & \vdots\\ w_{N1} & w_{N2} & \cdots & w_{NK}\end{bmatrix}\begin{bmatrix} sim_{11} & 0 & \cdots & 0\\ 0 & sim_{22} & \cdots & 0\\ \vdots & \vdots & \vdots & \vdots\\ 0 & 0 & \cdots & sim_{KK}\end{bmatrix}+\begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1K}\\ w_{21} & w_{22} & \cdots & w_{2K}\\ \vdots & \vdots & \vdots & \vdots\\ w_{N1} & w_{N2} & \cdots & w_{NK}\end{bmatrix}\begin{bmatrix} 0 & sim_{12} & \cdots & sim_{1K}\\ sim_{21} & 0 & \cdots & sim_{2K}\\ \vdots & \vdots & \vdots & \vdots\\ sim_{K1} & sim_{K2} & \cdots & 0\end{bmatrix} \tag{24}$$

标签与其自身的相似度是 1,所以公式(24)的左半部分相乘结果为原始矩阵.此外,由于标签集合中的所有标签都存在一定的概率相关性,所以标签相似性矩阵中的元素都是非零的,这保证了公式(24)的右半部分是非零的.因此映射之后,每个用户的标签向量的稀疏性将得到有效缓解.

3.2.3 微博推荐算法描述

定义推荐算法排序函数 f:给定用户 u_i 和微博 d_p,则排序函数 $f(\boldsymbol{u}_i,\boldsymbol{d}_p)$的定义[21]如公式(25)所示.

$$f(\boldsymbol{u}_i,\boldsymbol{d}_p)=\frac{\boldsymbol{d}_p\cdot\boldsymbol{u}_i}{\|\boldsymbol{d}_p\|\times\|\boldsymbol{u}_i\|} \tag{25}$$

其中,$f(\boldsymbol{u}_i,\boldsymbol{d}_p)$表示用户 u_i 与微博 d_p 之间的相关性.$u_i=(w_{i1},w_{i2},\ldots,w_{iK})$为更新后的用户 u_i 的标签权重向量.微博 d_p 可被表示为 $\boldsymbol{d}'_p=(t'_{p1},t'_{p2},...,t'_{pK})$,若微博 d_p 包含标签 t_i,则 $t'_{pi}=1$;否则,$t'_{pi}=0$.将微博文本向量映射到标签语义空间 $\boldsymbol{S}$ 上,则 $\boldsymbol{d}_p=\boldsymbol{d}'_p\times S=(t_{p1},t_{p2},...,t_{pK})$.预先设定阈值$\mu$,若该排序函数的值大于阈值$\mu$,则微博 d_p 将被推荐给用户 u_i.算法的框架如算法 1 所示.

算法 1. 基于超图随机游走标签扩充的微博推荐算法.

输入:用户 u_i 所发布的微博集合 D_i^{train} 和 D_i^{test},且 $D_i^{train}\cap D_i^{test}=\varnothing$;

微博用户集合 $U=\{u_1,u_2,\ldots,u_i,\ldots,u_N\}$;

阈值μ.

输出:微博推荐序列 D_i^{recom}.

//Step 1. 标签扩充

for $u_i\in U$ **do**

使用公式(6)和公式(11)分别对超边和超点加权,构建 $WHG(V,E,w(e),w(v_i,e_i))$;

使用公式(12)构建随机游走转移概率矩阵 $\boldsymbol{P}$;

使用公式(13)在加权超图 WHG_i 随机游走获得 top-Q 词项;

结合获得的 top-Q 词项和用户原有的标签集合形成新的标签集合 T_i;

end for

将所有用户的标签集合 T_i 合并形成全部的标签集合 T;

//Step 2. 用户兴趣建模

for $t_i\in T$ **do**

使用公式(17)将任意一个标签 t_i 表示成标签相关性向量;

end for

使用公式(19)构建标签相似度矩阵 S;

使用公式(22)构建原始的用户-标签矩阵 M;

使用公式(23)更新用户-标签矩阵得到矩阵 $\boldsymbol{M}'=\boldsymbol{M}\times\boldsymbol{S}$;

//STEP 3. 微博推荐

for $u_i\in U$ **do**

初始化 $D_i^{recom}=\varnothing$

for $\forall d_p\in D^{test}$ **do**

if $f(\boldsymbol{u}_i,\boldsymbol{d}_p)>\mu$ **then**

$D_i^{recom}=\{d_p\}\cup D_i^{recom}$;

end if

end for

end for

4 实 验

本节首先介绍了实验所用数据集以及评价标准,然后设计实验对本文的方法进行验证并对实验结果进行分析讨论.

4.1 实验数据集描述及评价指标

目前,没有同时包括微博用户标签和微博相关信息(内容、转发数以及点赞数等)的公开可用数据集,通过监

视新浪博客的最近更新列表,下载程序间歇性地抓取了 19 427 位用户 2015 年 7 月 16 日~2016 年 8 月 17 日发布的微博,并存储在数据库中.这些数据库记录包括了博文的标签、发布时间、微博内容、微博转发数、评论数及点赞数等信息.

对实验数据进行预处理,首先过滤文本中的@用户名、地址链接、和其他无意义字符等噪声信息后,对其进行分词,去除停用词.其中,分词采用 python 开源分词组件 jieba,停用词表采用新浪提供的 1 208 个停用词.实验中随机选取 13 000 名用户,删除拥有少于 4 个词汇的微博以及拥有少于 50 篇微博的用户,得到最终的实验数据集合.数据集中有用户 10 390 名,微博 2 186 283 条,标签个数 5 897 个.新浪微博允许用户最多添加 10 个关键词对自己进行描述,表 2 统计了数据集中添加不同标签数量的用户分布,其中,48.3%的用户至少添加了一个标签,而 51.7%的用户没有为自己添加标签,这充分表明了标签扩充对进行微博推荐的必要性.

Table 2 Distribution of users with different number of tags in dataset

表 2 数据集中添加不同标签个数的用户分布

标签数量	0	1	2	3	4	5	6	7	8	9	10
用户个数	5 371	964	427	373	312	293	364	196	178	267	1 645

为了验证推荐算法的准确性,将微博数据集分为训练集和测试集:训练集用来学习推荐方法中的相关参数,测试集用来验证推荐算法的准确性.为了避免数据过拟合,本文采用十折交叉验证的方法,将每个微博用户的数据样本随机划分成 10 个大小相等的子样本集,交叉验证过程重复 10 次.每次一个样本集被保留作为测试集的验证数据,其余 9 个样本集作为训练数据,其中,训练集中有 1 967 655 条样本,测试集中有 218 628 条样本.

本文实验环境为:Windows 7 操作系统,4GB 内存,Intel Core(TM) 2 Duo CPU 2.66GHz,实验程序使用 Java1.6 语言开发,数据库为 Mysql5.0.

准确率(precision)、召回率(recall)、*F*1 值(*F*-measure)和平均正确率(average precision,简称 AP)是广泛用于信息检索和推荐领域的 4 个度量值,用来评价结果的质量.为了评估微博推荐质量,本文采用前 L 条结果的准确率 $P@L$、前 L 条结果的召回率 $R@L$、前 L 条结果的 $F1$ 值 $F1@L$ 以及前 L 条结果的平均正确率 $AP@L$ 来评价微博推荐质量.$P@L$,$R@L$,$F1@L$ 和 $AP@L$ 定义如下.

$$P@L=\frac{1}{|User|}\sum_{i=1}^{|User|}\frac{\text{前}L\text{条推荐结果中用户}u_i\text{喜欢的微博个数}}{\text{算法向用户}u_i\text{推荐的微博个数}L} \tag{26}$$

$$R@L=\frac{1}{|User|}\sum_{i=1}^{|User|}\frac{\text{前}L\text{条推荐结果中用户}u_i\text{喜欢的微博个数}}{\text{用户}u_i\text{测试数据集中的微博个数}} \tag{27}$$

$$F1@L=\frac{1}{|User|}\sum_{i=1}^{|User|}\frac{2\times P@L\times R@L}{P@L+R@L} \tag{28}$$

$$AP@L=\frac{1}{|User|}\sum_{i=1}^{|User|}\frac{\sum_{k=1}^{L}P@k\times rel(k)}{\text{前}L\text{条推荐结果中用户}u_i\text{喜欢的微博个数}} \tag{29}$$

其中,$rel(k)$是一个指示函数,当推荐返回结果的第 k 个位置为相关微博,$rel(k)=1$;否则,$rel(k)=0$.微博中没有明确表明用户喜好的数据,本文中把用户发布的微博都认为是用户喜欢的微博.

4.2 实验结果与相关分析

为了验证本文方法的有效性及推荐结果的准确性,本节设计了 4 个实验,对本文提出的方法进行验证并对实验结果进行分析.一是通过改变参数α和阈值μ,比较微博推荐算法的性能,从而确定最优参数值;二是在参数值确定的基础上,验证标签扩充个数对微博推荐性能的影响;三是通过比较标签扩充前后的内容,展示标签扩充方法的性能;四是本文的微博推荐算法与其他算法的比较.

4.2.1 参数设置对方法性能的影响

下面将通过实验来考察方法中涉及到的参数对算法性能的影响,它们分别是参数α和阈值μ.当测试其中一个参数值对算法的影响时,另外一个参数值保持不变.

对超边加权时,α用于调节微博时间因子和微博人气指数的比重,其值越高,意味着用户发布微博的时间对于用户兴趣的提取影响越大;其值越小,就意味着微博的评论数、转发数等对于用户兴趣提取影响提高.为了计算参数α对于推荐结果的影响,本文分别对不同α取值下算法在微博推荐个数为 L=5、L=10、L=15 及 L=20 的推荐结果进行对比.设参数μ=0.5,分别在α取不同值时,比较方法的性能,图 2 展示了实验结果.对比图 2 左、右两图,可以发现以下几点.

(1) 左图中,当 L=15 时算法的准确率 $P@L$ 达到最佳;在右图中,当 L=20 时算法的召回率 $R@L$ 达到最佳.这是由于召回率依赖于用户测试样本数目,随着推荐数目的增加,算法召回率也逐渐增加.

(2) 当α=0.7 时,算法的准确率 $P@L$ 和召回率 $R@L$ 均达到最大值,算法的性能在各个微博推荐个数上都达到最佳状态.值得注意的是,当α=1 时,算法的推荐性能在准确率和召回率上都优于α=0 时,因此,微博时间因子对用户兴趣提取准确性的影响大于微博人气指数.在接下来的实验中,设定α=0.7.

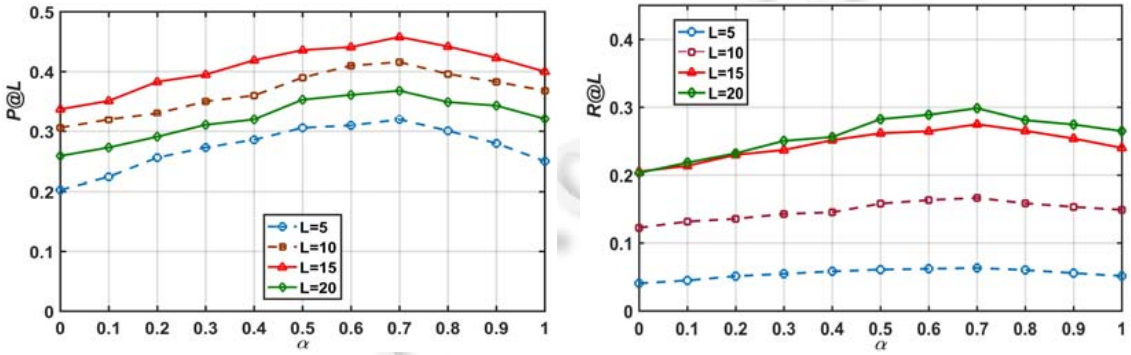


Fig.2 Impact of parameter α on recommendation algorithms

图 2 参数α对推荐算法的影响

阈值μ的大小决定了推荐方法向用户推荐微博数量的大小,阈值μ越小,则向用户推荐的微博数量越多;阈值μ越大,则向用户推荐的微博数量越少.为了清楚地了解阈值μ的取值对推荐算法的影响,令参数α=0.7,μ取不同的值,在测试数据集上计算方法取得的实验结果如图 3 所示.

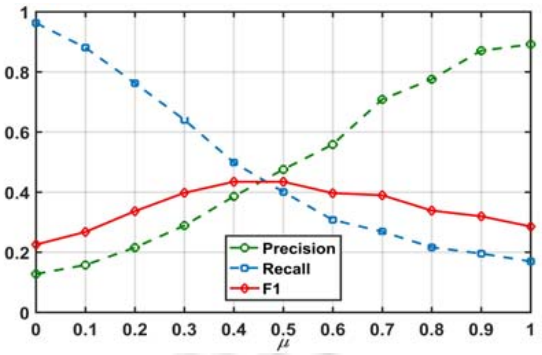


Fig.3 Impact of threshold μ on recommendation algorithms

图 3 阈值μ对推荐算法的影响

从图中可以看出,随着阈值μ的增大,算法的准确率逐渐上升,而算法的召回率却逐渐减小.这是因为随着阈值μ的增大,方法要求所推荐微博与用户的相似度也在不断增大,因而推荐给用户的微博越来越少.当μ=0.4 或 0.5 时,算法在 $F1$ 这一评价指标上都达到了最佳.因此,本文又在综合考虑μ=0.4 或 0.5 时算法的准确率和召回率这两个评价指标后,确定μ=0.45.

4.2.2 不同标签扩充个数对推荐算法的影响

为了验证标签扩充个数 Top-Q 对微博推荐方法的影响,分别选取{1,3,5,7,9,10}个关键词对用户标签进行扩充,计算在不同标签扩充个数的情况下本文算法的准确率,进而确定标签扩充个数 Top-Q 的值,如图 4 所示.从图中可以看出,随着标签扩充个数的增加,算法的准确率 $P@L$ 也逐渐增加.当标签扩充个数 P>9 时,算法的准确率 $P@L$ 不增反降.这是由于随着标签扩充个数的增大,一些排名靠后的关键词也被扩充到用户标签集合中,这部分标签并不能很好地表征用户的兴趣爱好.从图中可以看出,当 Q=7 或 9 时,算法的准确率 $P@L$ 并无明显增加,算法的性能在各个微博推荐个数上都达到最佳状态.因此,取 7 和 9 的均值 8 作为标签扩充个数 Q 的值.

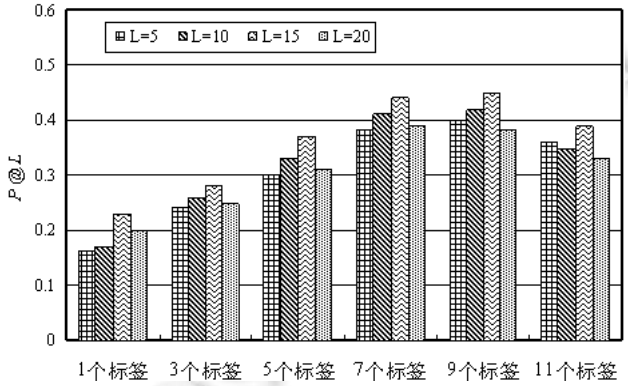


Fig.4 Impact of the number of tag extensions on recommendation algorithms

图 4 标签扩充个数对推荐算法的影响

4.2.3 标签扩充前后用户标签详情对比

限于篇幅,本部分只展示了 10 位用户自己添加的标签以及经过超图随机游走算法为用户添加的标签的情况,见表 3.

Table 3 Comparison of user tag before and after the tag expansion

表 3 标签扩充前后用户标签详情对比

微博用户	扩充前用户自己添加的标签	扩充后由推荐算法给用户添加的标签(Q=8)
User-1	淘宝控;摄影;睡觉;宅;旅游	马云;优惠券;音乐;90 后;美食;单反;时尚;大学生
User-2	美食;国际;娱乐;人生;时尚	成都;舌尖上的中国;旅游;电影;听歌;看书;自由;华人
User-3	艺术;收藏;历史	故宫;拍卖会;时尚;新闻;媒体;电影;财经;养生
User-4	信息化;科技;社会	计算机;手机;互联网+;创新;财经;创业;电子商务
User-5	文学;摄影;新媒体;旅行;电影;新闻资讯;普利策	音乐;时尚;奋斗;美食;诺贝尔;媒体人;科技;创新
User-6	设计;建筑	旅游;电影;音乐;美食;文学;摄影;历史;财经
User-7	美食;电影;音乐;90 后;奋斗	综艺;旅游;电子控;游泳;宅;大学生;优惠券;红包
User-8	历史;财经;健康;养生;保健	心灵鸡汤;平常心;情感;摄影;文学;新闻;教育;防诈骗
User-9	旅游;电影;音乐;时尚;自由;游泳;摄影;美食;90 后;旅游	大学生;舌尖上的中国;张艺谋;新媒体;韩剧;创业;奋斗;科技
User-10	互联网;创新;电子控	电子商务;财经;创业;互联网+;科技;奋斗;青年;新闻

可以看到,少数的高频词出现在相当多的微博用户标签中,这些热门标签的内容多是大众性的兴趣爱好的描述,如“音乐”“电影”“美食”等;或者是对一些常见人群的描述,如“大学生”“90 后”“宅”.这些标签之所以被频繁使用,一是因为这其中的一些标签在用户添加标签的页面作为系统推荐选项出现,因此有更大的概率被用户看到和选中,而不用手动输入;二是此类标签对于新浪微博用户具有普适性,即很多微博用户都会发现这样的标签在某种程度上符合对自己的描述.例如在实验数据集中,有 52.7%的用户是大学生, “奋斗”“90 后”两个标签非常符合对这些用户的描述,因此成为高频标签.

4.2.4 不同微博算法的性能比较

为了验证该推荐算法的有效性,比较本文提出的 LeALpc 算法与基于标签关联关系推荐算法(label

correlation,简称 LC)[13]、基于标签概率相关性推荐算法(label probability correlation,简称 LPC)[14]、融合标签关系与用户关系推荐算法(label correlation and user social relation,简称 ILCAUSR)[15]、协同个性化微博推荐(collaborative personalized tweet recommendation,简称 CTR)[3]、基于用户嵌入的学术微博推荐(user embedding for scholarly microblog recommendation,简称 UEMR)[4]和基于背景和内容的微博推荐(microblog recommendation based on profile and content,简称 BPACMR)[22]的预测效果.选择以上 6 种算法作为对比算法是基于以下几点考虑.

(1) 本文算法是在 LPC 算法的基础上改进而来,LPC 算法与本文的算法最相似.

(2) LC 算法、ILCAUSR 算法、LPC 算法以及本文的算法都是基于标签进行微博推荐的.

(3) ILCAUSR 算法已被证明在微博推荐算法上优于其他算法.

(4) 由于前面 3 种比较算法都是从标签角度出发的微博推荐算法,为了更好地验证本文方法的有效性,采用从其他角度出发且具有较好性能的微博推荐算法(CTR 算法、UEMR 算法和 BPACMR 算法)进行对比.

利用不同微博推荐列表长度 L=5、L=10、L=15 及 L=20 对以上算法进行实验,比较在不同推荐列表长度的情形下,几种推荐算法的准确率 $P@L$、$F1$ 值 $F1@L$ 以及平均正确率 $AP@L$,结果见表 4.

Table 4 Comparison of different recommendation algorithms

表 4 不同推荐算法比较

名称	L=5		L=10		L=15		L=20	
	P	*AP*	*P*	*AP*	*P*	*AP*	*P*	*AP*
LC	0.281	0.42	0.316	0.386	0.334	0.524	0.231	0.442
LPC	0.283	0.4	0.359	0.452	0.379	0.465	0.226	0.469
ILCAUSR	**0.315**	0.52	0.427	0.5	0.436	0.58	**0.255**	0.542
CTR	0.295	0.448	0.396	0.524	0.418	0.563	0.247	0.52
UEMR	0.299	0.432	0.403	0.469	0.413	0.549	0.25	0.488
BPACMR	0.293	0.54	0.388	0.488	0.408	0.517	0.246	0.506
LeALpc	0.308	**0.559**	**0.428**	**0.586**	**0.439**	**0.643**	0.253	**0.587**

从表中可以看出,本文提出的 LeALpc 算法与从内容角度出发的 CTR 算法、UEMR 算法和 BPACMR 算法以及从标签角度出发的 LPC 算法、算法 LC 和 ILCAUSR 算法在平均准确率方面相比都更优异.这是由于这些算法过分关注用户的整体兴趣而忽视了用户的个性化兴趣,导致推荐列表前几位的命中率低.而 LeALpc 算法结合了微博文本和用户标签这两个体现用户兴趣的重要方面,它更能展现用户的个性化兴趣.在实际应用中,推荐正确的次序尤其重要,因为用户不可能耐心浏览完所有推荐的微博.在其他评价指标上,LeALpc 算法明显高于除 ILCAUSR 算法之外的 5 种算法,但是与 ILCAUSR 算法相比并没有明显优势.这是由于 ILCAUSR 算法将用户间社交关系融入到微博推荐算法中,较为准确地表示出了用户的兴趣.而本文尚未考虑,这也将是本文今后继续研究的方向.

接着,为了进一步比较 LeALpc 算法和其他 6 种算法的推荐性能,从表 4 中选取推荐性能(正确率 $P@L$)最好情况(L=15)和最坏情况的(L=20)的正确率和平均正确率展开分析,分别是 L=15 和 L=20 时,7 种算法在不同评价指标上 10 次交叉验证所得结果的分布情况,如图 5 所示.箱线图是一种数据样本统计图,它可以看出数据是否具有对称性以及数据分布的分散程度等信息.因此,从图 5 可以看出,

- 在评价指标 $AP@L$ 上,无论是最好情况(L=15)还是最坏情况(L=20),本文提出的 LeALpc 算法不但具有较高的平均值,而且 10 次所得结果也比较稳定.
- 在评价指标 $P@L$ 上,在最好情况(L=15)下,本文提出的算法与 ILCAUSR 算法相比,在平均值上虽然并没有明显优势,但是在结果分布上要优于其他算法.

这更加验证了根据微博用户以往发布的微博内容对其标签进行扩充以及根据标签概率相关性对用户进行微博推荐这一方法的有效性.

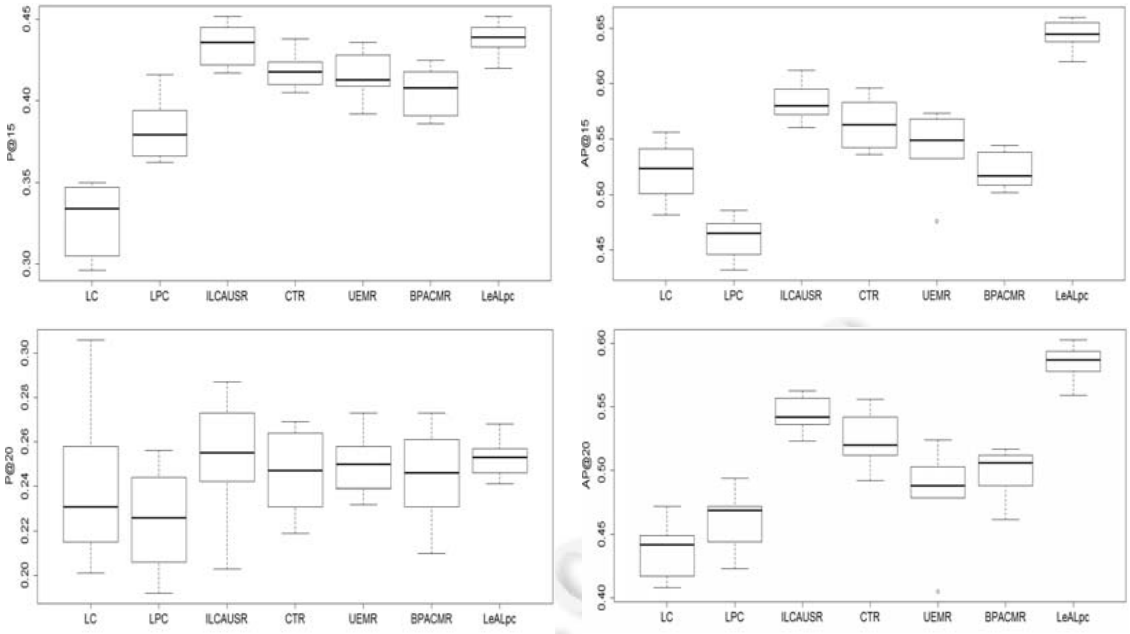


Fig.5 Performance comparison among LeALpc algorithm and other baselines on L=15 and L=20

图 5 LeALpc 算法与对比算法在 L=15 和 L=20 时的性能比较

5 结论与展望

随着移动互联网的快速发展以及社交网络规模的不断增大,个性化的信息推荐越来越受到信息接收者的青睐.为了提升用户浏览信息的体验度,面对海量复杂的微博消息,实现内容精准推荐.本文从微博用户标签入手,针对绝大多数微博用户没有给自己加注标签或标签较少的问题,提出一种结合标签扩充与标签概率相关性的微博推荐方法.首先,该方法将微博视为超边,微博中的词视为超点来构建超图,并以一定的加权策略对超边和超点进行加权,通过在超图上随机游走得到一定数量的关键词对微博用户标签进行扩充;然后,采用相关性标签权重加权方案,构建用户-标签矩阵,利用标签间的概率相关性,构造标签相似性矩阵,对用户-标签矩阵进行更新,更新后的用户标签矩阵不仅稀疏性得到了有效缓解,而且还包含了丰富的标签关联关系;最后,依据构建的兴趣模型对用户进行信息推荐.在未来的工作中,将进一步对用户与用户之间的社交属性进行研究,提升用户模型的准确度,实现更加精准的推荐.

References:

[1] Chen Y, Cheng XQ, Yang S. Finding high quality threads in Web forums. Ruan Jian Xue Bao/Journal of Software, 2011,22(8): 1785−1804 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/3857.htm [doi: 10.3724/SP.J.1001.2011.03857]

[2] Zhang D. Research on microblog recommendation method based on user social behavior [Ph.D. Thesis]. Lanzhou: Northwest Normal University, 2018.

[3] Chen KL, Chen TQ, Zheng GQ, Jin O, Yao EP, Yu Y. Collaborative personalized tweet recommendation. In: Proc. of the 35th Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. Portland: ACM Press, 2012. 661−670.

[4] Yang Y, Wan XJ, Zhou XJ. User embedding for scholarly microblog recommendation. In: Proc. of the 54th Annual Meeting of the Association for Computational Linguistics. Berlin: Association for Computational Linguistics, 2016. 449−453.

[5] Gao M, Jin CQ, Qian WN, Wang XL, Zhou AY. Real-time and personalized recommendation on microblogging systems. Chinese Journal of Computers, 2014,37(4):963−975 (in Chinese with English abstract).

[6] Sun A. Short text classification using very few words. In: Proc. of the 35th Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. Portland: ACM Press, 2012. 1145−1146.

[7] Meng XW, Liu SD, Zhang YJ, Hu X. Research on social recommender systems. Ruan Jian Xue Bao/Journal of Software, 2015, 26(6):1356−1372 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4831.htm [doi: 10.13328/j.cnki.jos.004831]

[8] Ramage D, Dumais ST, Liebling DJ. Characterizing microblogs with topic models. In: Proc. of the Int'l AAAI Conf. on Weblogs and Social Media. Washington: AAAI Press, 2010. 130−137.

[9] Liu WY, Quan XJ, Feng M, Qiu B. A short text modeling method combining semantic and statistical information. Information Sciences, 2010,180(20):4031−4041.

[10] Weng JS, Lim EP, Jiang J, He Q. TwitterRank: Finding topic-sensitive influential twitterers. In: Proc. of the 3rd ACM Int'l Conf. on Web Search and Data Mining. New York: ACM Press, 2010. 261−270.

[11] Zhang B, Zhang Y, Gao KN, Guo PW, Sun DM. Combining relation and content analysis for social tagging recommendation. Ruan Jian Xue Bao/Journal of Software, 2012,23(3):476−488 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4001.htm [doi: 10.3724/SP.J.1001.2012.04001]

[12] Xing QL, Liu L, Liu YQ, Zhang M, Ma SP. Study on user tags in Weibo. Ruan Jian Xue Bao/Journal of Software, 2015,26(7): 1626−1637 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/4655.htm [doi: 10.13328/j.cnki.jos.004655]

[13] Ma HF, Jia MHZ, Xie M, Lin XH. A microblog recommendation algorithm based on multi-tag correlation. In: Proc. of the 8th Int'l Conf. on Knowledge Science, Engineering and Management. Chongqing: Springer-Verlag, 2015. 483−488.

[14] Zhang D, Ma HF, Jia JJ, Yu L. A microblog recommendation method based on label probability correlation. Computer Engineering and Science, 2017,39(9):1742−1748 (in Chinese with English abstract).

[15] Ma HF, Jia MHZ, Zhang D, Lin XH. Combining tag correlation and user social relation for microblog recommendation. Information Sciences, 2017,385:325−337.

[16] Hua W, Wang ZY, Wang HX, Zheng K, Zhou XF. Short text understanding through lexical-semantic analysis. In: Proc. of the 31st Int'l Conf. on Data Engineering. Seoul: IEEE Press, 2015. 495−506.

[17] Bellaachia A, Al-Dhelaan M. HG-rank: A hypergraph-based keyphrase extraction for short documents in dynamic genre. In: Proc. of the 4th Workshop on Making Sense of Microposts. Seoul: CEUR Workshop, 2014. 42−49.

[18] Liu Q, Li ZG, Lui JCS, Cheng JF. PowerWalk: Scalable personalized pagerank via random walks with vertex-centric decomposition. In: Proc. of the 25th ACM Int'l Conf. on Information and Knowledge Management. Indianapolis: ACM Press, 2016. 195−204.

[19] Tu NN, Kanhabua N, Zhu X. A time-aware random walk model for finding important documents in web archives. In: Proc. of the 38th Int'l ACM SIGIR Conf. on Research and Development in Information Retrieval. Santiago: ACM Press, 2015. 915−918.

[20] Song SX, Zhu H, Chen L. Probabilistic correlation-based similarity measure on text records. Information Sciences, 2014,289: 8−24.

[21] Zhou XK, Wu S, Chen C, Chen G, Ying SS. Real-time recommendation for microblogs. Information Sciences, 2014,279:301−325.

[22] Zhong ZM, Guan Y, Hu Y, Li CH. Mining user interests on microblog based on profile and content. Ruan Jian Xue Bao/Journal of Software, 2017,28(2):278−291 (in Chinese with English abstract). http://www.jos.org.cn/1000-9825/5030.htm [doi: 10.13328/j.cnki.jos.005030]

附中文参考文献:

[1] 陈友,程学旗,杨森.面向网络论坛的高质量主题发现.软件学报,2011,22(8):1785−1804. http://www.jos.org.cn/1000-9825/3857.htm [doi: 10.3724/SP.J.1001.2011.03857]

[2] 张迪.基于用户社交行为的微博推荐方法研究[硕士学位论文].兰州:西北师范大学,2018.

[5] 高明,金澈清,钱卫宁,王晓玲,周傲英.面向微博系统的实时个性化推荐.计算机学报,2014,37(4):963−975.

[7] 孟祥武,刘树栋,张玉洁,胡勋.社会化推荐系统研究.软件学报,2015,26(6):1356−1372. http://www.jos.org.cn/1000-9825/4831.htm [doi: 10.13328/j.cnki.jos.004831]

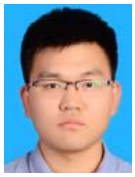
[11] 张斌,张引,高克宁,郭朋伟,孙达明.融合关系与内容分析的社会标签推荐.软件学报,2012,23(3):476−488. http://www.jos.org.cn/1000-9825/4001.htm [doi: 10.3724/SP.J.1001.2012.04001]

[12] 邢千里,刘列,刘奕群,张敏,马少平.微博中用户标签的研究.软件学报,2015,26(7):1626−1637. http://www.jos.org.cn/1000-9825/4655.htm [doi: 10.13328/j.cnki.jos.004655]

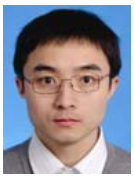
[14] 张迪,马慧芳,贾俊杰,余丽.一种基于标签概率相关性的微博推荐方法.计算机工程与科学,2017,39(9):1742−1748.

[22] 仲兆满,管燕,胡云,李存华.基于背景和内容的微博用户兴趣挖掘.软件学报,2017,28(2):278–291. http://www.jos.org.cn/1000-9825/5030.htm [doi: 10.13328/j.cnki.jos.005030]

马慧芳(1981－),女,甘肃兰州人,博士,教授,CCF 专业会员,主要研究领域为机器学习,数据挖掘.



张迪(1992－),男,硕士,主要研究领域为互联网数据挖掘.



赵卫中(1981－),男,博士,副教授,CCF 专业会员,主要研究领域为机器学习,数据挖掘,算法分析与设计.

史忠植(1941－),男,研究员,博士生导师,CCF 会士,主要研究领域为人工智能,机器学习,神经计算,认知科学.