DOI: 10.3785/j.issn.1008-973X.2013.08.023

基于用户评论挖掘的产品推荐算法

扈中凯，郑小林，吴亚峰，陈德人
(浙江大学 计算机科学与技术学院，浙江 杭州 310027)

摘　要：针对电子商务推荐系统中，互联网“信息过载”所造成的难以精确定位用户兴趣并提供准确产品推荐的问题，通过深入挖掘电子商务社区中丰富的用户评论信息，开发产品特征提取算法，建立用户兴趣偏好模型，结合用户历史评分数据来改善传统协同过滤推荐算法的推荐准确性；利用相似度传递技术在一定程度上缓解推荐系统中数据稀疏性带来的问题. 实验结果表明，在数据稀疏的情况下，该算法仍可较好地拟合用户对产品的兴趣偏好，并在推荐准确性方面较传统的协同过滤算法有明显的提高.

关键词：评论挖掘；产品特征属性；用户偏好；协同过滤；相似度传递
中图分类号：TP 319；TP 391　　**文献标志码**：A　　**文章编号**：1008-973X(2013)08-1475-11

Product recommendation algorithm based on users' reviews mining

HU Zhong-kai, ZHENG Xiao-lin, WU Ya-feng, CHEN De-ren
(*College of Computer Science and Technology, Zhejiang University, Hangzhou* 310027, *China*)

Abstract: In E-commerce recommendation system, "Information overload" on Internet has brought a tough problem, which is how to precisely position users' interest and provide users with accurate product recommendation. To solve this problem, in this paper, a product characteristic extraction algorithm was developed and a user preference model was constructed by deeply mining large-scale of user reviews in E-commerce community, to improve accuracy of traditional collaborative filtering recommendation algorithm with coordination of historic user rating information; moreover, data sparsity problem was alleviated with similarity propagation technique. Experiment results show that, in condition of sparse data, algorithm in this paper can still fit product preference of users very well, and has significantly improvement in accuracy compared with traditional collaborative filtering algorithm.

Key words: reviews mining; product feature; user's preference; collaborative filtering; similarity propagation

随着电子商务的发展，互联网上的商品类目和数量不断增加，使得用户很难抉择自己想要买的商品，同时电子商务企业也难以了解用户感兴趣的内容. 推荐系统可以根据用户兴趣爱好给用户推荐可能感兴趣的信息，解决“信息过载”问题[1]，提高信息处理效率. 因此，推荐系统在各大电子商务企业和社交网络中得到了广泛的应用.

在电子商务中很多用户在购买商品后都会发表相应评论. 这些评论包括用户对商品性能、功能等方面肯定或否定的态度. 对商品评论的分析对消费者和产品销售商来说都非常有价值，尤其在被应用于推荐系统方面时，商品评论分析可以发挥它的巨大价值. 因为用户的评论信息往往隐含了用户对商品的喜好程度及对商品特定方面的关注程度等潜在信

收稿日期：2012-05-06.　　浙江大学学报(工学版)网址：www.journals.zju.edu.cn/eng
基金项目：国家科技支撑计划资助项目(2012BAH16F02)；国家自然科学基金资助项目(61003254).
作者简介：扈中凯(1987—)，男，博士生，从事为推荐系统、自然语言处理等研究. E-mail：huzhongkai2005@gmail.com
通信联系人：郑小林，男，副教授. E-mail：xlzheng@zju.edu.cn

息.将这些信息提取出来,为用户的兴趣爱好建立模型,可以更加精准地为用户推荐合适的商品.

鉴此,本文在对产品推荐相关研究进行深入分析的基础上,通过挖掘用户评论产生用户偏好,并结合用户综合相似度传播对传统的协同过滤算法进行改进,最终得到基于综合用户偏好与用户历史评分等多方面因素的产品推荐算法.

1 相关工作

推荐技术可以分为2类:基于内容和基于协同过滤的推荐技术.协同过滤技术是目前推荐系统中应用最广泛及效果最好的技术之一.由Goldberg等[2]提出的Tapestry系统是最早的协同过滤系统.该系统利用小型社区成员的直接观点来进行电子邮件分类过滤.但是该系统不适合应用到大型社区中去,所以各类型的协同过滤技术就陆续出现了.例如Konstan等[3]提出的Grouplens是用户评分自动化协同过滤推荐系统,用于向用户提供电影以及新闻的推荐.另外,协同过滤也面临着数据稀疏性,系统扩展性,新用户问题等.Sarwar等[4]提出使用矩阵奇异值分解的方法降低评分矩阵维度以减少稀疏性.邓爱林等[5]提出首先根据基于项目的协同过滤算法预测部分项目评分,减少评分稀疏性,再根据基于用户的协同过滤算法为用户进行推荐.Aggarwal等[6]提出Horting图的技术,将用户作为节点,用户间的相似度为边构成图,然后搜索用户邻居节点获得推荐.

产品评论挖掘是最近几年文本挖掘领域中兴起的研究热点,该研究主要包括2项基本任务,即产品特征词发现和评价情感词发现,国内外对这2方面研究基本都是分开进行的.产品属性词提取方面,Hu等[7]对名词和名词性短语进行关联规则挖掘高频属性词和低频属性词作为候选评价对象,然后再通过"紧凑修剪"和"冗余词修剪"去除那些可能不是产品属性词的名词或名词性短语.Etzioni等[8]在KownItAll网络信息抽取系统基础之上建立了一个无监督的信息挖掘系统OPINE,通过人工定义抽取指定关系(is a关系和part of关系)的文本模式抽取产品属性词,用OPINE来挖掘产品属性词,准确率比Hu挖掘结果高出了近22%,而召回率仅下降了3%.情感词识别方面,主要有基于统计和语义的方法.基于统计的方法主要是使用"互信息"和模板.Turney[9]建立了褒义和贬义种子词库,通过计算词语与种子词库中词的互信息来确定词语的情感倾向.Riloff等[10]研究如何使用Bootstrapping方法从语料库中获得主观表达模板并计算其得分,运用得到的模板来提取含有主观意向的词.基于语义的方法主要是利用现有的本体知识库进行分析.Hu等[7]将情感词限定在形容词集合中,运用WordNet中形容词的同义词集合和反义词集合来判断形容词的情感倾向.娄德成等[11-13]分别对HowNet中的6 564个词条和从2 454篇汽车评论中人工选择得到的极性词汇以人工标注方式建立极性词汇表,而对于在词汇表中没有的词,通过互信息确定词的极性.

在结合评论挖掘的推荐方面,Adomavicius等[14]在深入研究各种推荐算法的基础上,指出推荐系统在结合评论挖掘的方面还有待发展.Wiet-sma等[15-16]利用了评论挖掘来做产品描述和学习用户行为,并相信推荐系统广泛结合用户评论会带来更加精确的推荐.Aciar等[17]提出一种商品质量排序机制,通过用户专业程度及用户对商品某些特征的评分对商品质量进行排序,同时该方法采用了领域本体的方法把评论信息翻译成推荐系统容易处理的信息.

针对目前推荐系统中结合产品评论研究相对缺乏的情况,本文提出一种基于产品评论挖掘的推荐算法.该算法不仅考虑用户偏好对推荐结果的影响,而且改进传统的协同过滤算法,增加了相似度传递的过程[18],很大程度上提高了算法的推荐效果.

2 整体框架

本文实现的产品推荐算法首先通过用户评论挖掘提取特征情感词对,划分产品的特征属性层面,量化产品各层面的分数;然后,根据用户评论数据学习用户的个人偏好;最后,根据用户之间的兴趣和偏好的相似度,通过改进的协同过滤算法对用户进行产品推荐.整体框架如图1所示.

3 产品特征抽取与量化

3.1 特征情感词对提取

为了能够更清晰的表达和展示本文的讨论,在这里先对本文研究的问题进行形式化定义和说明.

定义1:特征情感词对.一个特征情感词对$f=(w_h,w_m)$由特征属性词w_h及它的修饰词w_m共同组成,w_h代表用户关注的产品细粒度特征,如质量,价格等,w_m为修饰特征词的情感词,表达了用户对产品特征的主观感受,如清晰,不错等.

定义2:用户评论.

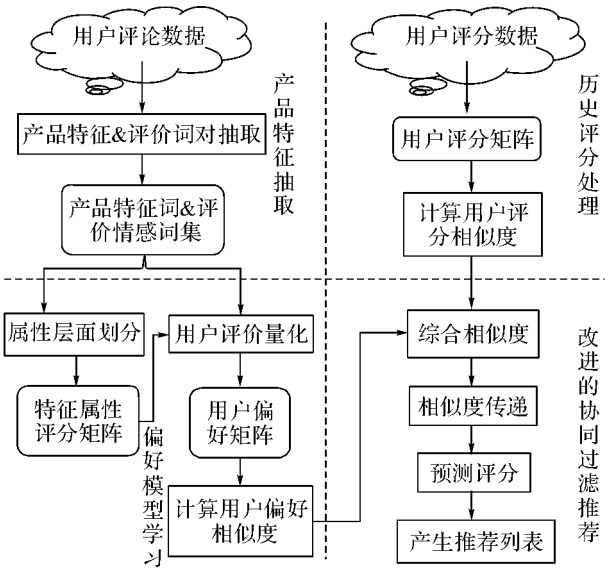

图 1　产品推荐算法整体框架

Fig. 1　Framework of product recommendation algorithm

给定用户评论数据集 $T=\{t_1,t_2,\cdots,t_n\}$，其中的一条用户评论 t 表达了用户对所评论产品各个特征的褒贬意见，可以看作是一堆特征情感词对的集合，$t=\{f=(w_h,w_m)\mid f\in t\}$.

想要从评论中提取产品在各个层面上的特征，首先要从用户评论中找出表达用户观点的主观评价语句. 在经过观察大量评论数据后发现，评论中的产品特征属性词通常以名词，名词短语或动词为主，评价情感词则常以形容词，动词或名词为主，并且产品特征词与情感词之间的搭配通常有类似于“名词＋形容词”的语言模式，如“屏幕清晰”、“电池耐用”、“反应不错”. 因此可以假设评论中产品特征词与情感词之间存在着某些句法联系. 本文将语言粒度泛化到词性 POS (part of speech)层面，采用统计的方法寻找这些频繁的联系，再经过过滤得到代表这些联系的词性路径模板，最终形成提取特征情感词对的词性路径模板提取算法[19].

词性路径模板提取算法的伪代码如下所示：

Input：预处理后的训练语料，$S_i(i=1,2,\cdots,n)$代表其中一条评论.

Output：词性路径模板集 M 及提取出来的特征情感词对集合 WordCouple

Step1：$FS_i=\mathrm{Fethch}(S_i)$

// 提取评论中的句子，如：“运行/v 比较/d 快/a”

Step2：$FS_{i-d}=\mathrm{Deleteword}(FS_i)$

// 删去 FS_i 中所有词，保留其词性标注，上例结果为“/v/d/a”

Step3：

if (length (FS_{i-d})<a and $FS_{i-d}=M_x$) then $f_{mx}+1$

elseif (length (FS_{i-d})<a and $FS_{i-d}\notin M_x$) then $M\leftarrow(FS_{i-d},1)$

// 长度过滤，对于模板 FS_{i-d} 长度小于给定的长度阈值 a 且已存在于模板库中，则将其出现频度 f_{mx} 加 1；若未在模板库中出现过，则将其频度置 1 后加入到模板库中

Step4：for (every f_{mx}) $p_{mx}=f_{mx}/\mathrm{total}(T)$

// 计算频率

Step5：if ($p_{mx}<\beta$) delete p_{mx} from M

// 频率过滤，若候选模板的频率 p_{mx} 小于给定阈值 β，则从模板库中将其删除

Step6：$\mathrm{wordCouople}_i=\mathrm{MatchFetch}(S_i,M)$

// 得到特征情感词对的候选模板集合；通过匹配评论中每个句子的模板，提取相应的特征词与情感词对

词性路径模板提取算法首先对语料库评论数据进行分词和词性标注，然后提取每个评论的分句的完整路径. 获取路径局限于词语层面，则其通用性较差，无法形成模板来处理大量数据. 所以需要将其泛化到词性这一语言粒度，得到泛化路径，这些泛化路径就是候选模板. 然而不是所有的泛化模板都是可用的，很多模板会给系统带来很多噪声. 所以本文对候选模板运用 2 种过滤策略进行过滤：

1) 依长度过滤：很多候选模板长度较长，通常是一些客观事实的陈述句，这些模板的实用性差，会降低系统性能，所以算法中根据模板的长度将较长模板过滤；

2) 依频率过滤：对于出现频率较低的候选模板，他们实用性差，另外他们还可能是因为分词或词性标注误差所引入的错误模板，所以需要根据频率滤除出现频率较低的候选模板，提高模板通用性和系统性能；

为了明确提取目标，首先对语料中特征词和情感词的词性进行了统计分析，并对语料库中部分数据进行手工标注与划分特征情感词对，这部分数据包括 300 条评论语句. 经过标注和划分之后得到 919 对特征情感词对. 在提取模板之前，首先对这部分语料中产品特征词和情感词的词性进行统计，用以指导之后对整个语料库的词性路径模板提取，统计结果如表 1、2 所示.

从表 1 中可以看出产品特征词有 86.61%为名词，其余为动词. 所以在提取产品属性词时可以仅考虑名词和动词. 根据此项统计结果，在提取产品属性词时，可以根据词性首先做简单判断，免去不必要的操作.

从表 2 可以看出情感词主要为形容词，名词，动词，其中 76.28%为形容词. 所以在提取产品情感词时可以仅这几类词，这样可以在提取特征词和情感

词时剔除一些不必要的句子成分.

通过将长度阈值设为5,频率阈值设为5%,统计模板进行过滤,最终从整个语料库提取出12个高频词性路径模板,如表3所示.

表1 产品特征词的词性统计表

Tab.1 POS statistical table of product feature words

特征词词性	示例	频率	比例
名词	屏幕	796	86.61%
动词	显示	123	13.39%

表2 情感词的词性统计表

Tab.2 POS statistical table of opinion words

情感词词性	示例	频率	比例
形容词	清晰	701	76.28%
名词	垃圾	126	13.71%
动词	可以	92	10.01%

表3 词性路径模板统计表

Tab.3 Statistical table of POS path templates

词性路径模板	示例	频率
名词+形容词	屏幕大	23.7%
名词+副词+形容词	外形不好看	11.8%
形容词+名词	好东西	10.4%
动词+形容词	显示清晰	7.56%
名词+副词+形容词+形容词	东西很好不错	7.08%
动词+副词+形容词	运行比较快	6.19%
名词+副词+动词	重量还可以	6.02%
名词+名词+形容词	屏幕颜色不错	6.01%
名词+名词+副词+形容词	机器手感还不错	6.01%
名词+副词+副词+形容词	做工都很好看	6.0%
动词+名词+形容词	感觉系统不错	6.0%
名词+形容词性名词	系统稳定	5.0%

从表3结果可知:在语料库中23.7%的产品特征词和情感词符合“名词+形容词”这一词性路径模板,11.8%符合“名词+副词+形容词”这一词性路径模板.实验结果验证了本文先前的假设,产品属性词与情感词间确实存在某种语言模式,可以将这些语言模式用一些可用的词性路径模板来表示.

3.2 产品特征词过滤

通过词性路径模板提取出来的名词,名词短语等,不一定都是产品特征词.实验中发现一些词虽然在评论语料中频繁出现(例如:“问题”,“同事”等),但与主题是不相关的.本文利用“互信息”(point mutual information, PMI)[9]的方法对候选特征词进行过滤,提高提取结果的准确率.

在从训练语料中出现次数最多的前100个特征词中,手工挑选出手机6个典型属性,分别是屏幕,电池,按键,待机时间,音效,外观,并加入产品类别名称“手机”,将其组成领域性特征词的种子集合Seeds,PMI-IR计算公式如下:

$$\text{PMI-IR}(w_1)=\sum_{w\in \text{seeds}} \ln \frac{\text{hits}\ (w_1 \& w)}{\text{hits}\ (w_1)\text{hits}\ (w)}. \tag{1}$$

式中:Seeds={屏幕,电池,按键,待机时间,音效,外观},hits (w_1),hits(w)分别代表特征词和经过Google双引号技术精确匹配后返回的结果页面数,而hits $(w_1 \& w)$则代表w_1和w通过Google共同精确匹配的结果页面数.候选特征词的PMI-IR值越高,则越可能是真正的产品特征词.这里需要设定一个阈值,大于阈值的候选特征词才被认为是真正的产品特征词,公式如下所示:

$$\text{isAttr}\ (w)=\begin{cases}1, & \text{PMI-IR}(w)\geqslant\beta;\\ 0, & \text{PMI-IR}(w)<\beta.\end{cases} \tag{2}$$

为了确定PMI方法中阈值的取值,首先从已抽取的候选特征词中挑选了456个产品属性特征词作为标准答案,然后利用准确率、召回率及调和平均值评估阈值取何值的情况下其结果最准确.实验表明,当$\beta=-169$时,可达到最优过滤效果.

3.3 特征属性层面划分

通过基于词性路径模板的特征情感词对提取方法,提取得到了语料库中关于手机的大部分特征,经过过滤之后的部分特征如表4所示.

表4 部分手机特征词列表

Tab.4 Part of mobile phone feature words list

速度,性价比,价格,外观,功能,手感,屏幕,质感,服务,系统,性能,散热,键盘,颜色,价钱,电池,触屏,画面,按键,款式,信号,游戏,色彩,分辨率,内存,反应速度,发热,电池容量…

从表4中可以看出,手机的特征词有很多,而且很多单词指的是同一个特征,如,机器、整体、手机等均指手机的整体,所以如果针对每一个特征都单独考虑,会使得主题特征空间维度很大,大大增加算法的复杂度.因此本文提出了特征属性层面的概念,通过将相似产品特征归类到同一特征属性层面中,可以降低特征空间维度,减少算法复杂度.

定义3:特征属性层面.产品的特征属性层面A是指该产品的某一固有特征属性,它可表示为一组

词意相近的属性特征词集合 $A=\{w_h \mid w_h \in A\}$. 如，对手机产品，“性能”是它的一个特征属性层面，包括“运行速度”、“散热效果”等不同的特征属性词. 若某个特征属性词 $w_h \in A$，且存在 $f=(w_h, w_m)$，则 f 是针对层面 A 的特征情感词对；一个产品可具有多个特征属性层面.

本文基于这样的理念，提出一个简单而较为准确的方法，即根据其中词意类似的特征词进行人工划分，将大量的特征词根据其表达意思分为几个层面，这样做能够极大的减少关注层面的维数，使每个层面的评分更趋于准确. 所以，本文从评论提取出来的手机特征词中根据特征词之间的词语相似性与词意相似性，将其划分为整体，性能，质量和服务 4 个层面. 具体划分如表 5 所示.

表 5　手机层面划分表

Tab. 5　Mobile phone level division table

层面	特征集合
整体	总体 产品 包装 价位 印象 使用 操作 销量 经济 得分 设计 大小 界面 机器设计 烤漆 外观 样式 实物 厚度 …
性能	速度 系统 散热 效果 噪音 字体 画面 反应 亮度 时间 上网 游戏 色彩 开机时间 分辨率 重量 智能 开机 图片 热量 …
质量	内存 快捷键 光驱 电池容量 处理器 产品质量 接口 耳机 内存卡 蓝牙 电容屏 镜头 视频 电板 触摸 程序 侧滑 触感 滑盖 …
服务	态度 送货 物流 服务 送货上门 发货 赠品 配送 服务质量 送货员 购物 特价 快递 信誉 提货 …

从结果中可以看出，虽然在部分层面的划分上也存在一些偏差，但整体的划分还是比较准确的.

3.4　量化特征属性层面评分

经过词性模板提取和特征词过滤后，从用户评论中获得较准确的特征情感词对，每一个词对都表达了用户对产品某个特征的褒贬态度，其中用户的褒贬态度将由情感词的极性决定. 本文借助目前比较权威的情感词典——《知网》(HowNet). 通过情感词典对情感词的极性判断，可得到每个用户对哪些产品层面持何种态度. 本文借助 HowNet 中情感词极性的划分，定义了量化产品层面分数的方法. 本文同时考虑了否定词的修饰，主要体现为句子含有否定修饰前缀，如“诺基亚的价格一点儿都不便宜”，因此本文定义了部分典型的否定词组成否定词典来处理否定词带来的极性变化，如“不”，“不会”，“没有”和“不是”等.

每个词对的评分 $r(f)$，将由词对中的情感词决定，度量标准与整体评分的标准相同，范围为 1 分到 5 分. 如果通过情感极性判断情感词为褒义，则这个词对的评分是 5 分，如是为贬义，则评分为 1 分，否则，认为是中性，评分为 3 分. 其评分规则如下：

$$r(f)=\begin{cases}5, & \text{若 } w_m \text{ 为褒义；}\\ 1, & \text{若 } w_m \text{ 为贬义；}\\ 3; & \text{其他情况.}\end{cases} \tag{3}$$

产品某个特征属性层面的好坏由全体用户评价来决定，即产品的某个层面 A_i 的分数 $r(A_i)$ 等于针对该层面的所有用户评论词对的评分均值，公式如下：

$$r(A_i)=\frac{\sum_{f\in A_i} r(f)}{c_{A_i}(f)}. \tag{4}$$

式中：$c_{A_i}(f)$ 表示针对 A_i 层面的词对的数目，$i=1, 2, \cdots, k$，k 代表划分的层面数，且 $r(A_i)\in[1,5]$.

经过以上定义的量化规则，就可以从某个产品 T 的所有评论中区分出针对不同层面的评论并量化得到各个层面的分数，其中部分产品层面的分数计算结果如表 6 所示.

表 6　部分手机各层面分数

Tab. 6　Part of mobile phone level scores

手机	整体	性能	质量	服务
诺基亚 X2GSM 手机	4.71	4.57	4.73	4.79
三星 S55703G 手机	4.93	4.95	4.94	5
LG GT350 GSM 手机	3.67	3	3	3

从表 6 中可以看出，产品各层面的分数一定程度上反映了产品在该层面的优劣好坏，不同的产品在各个层面上也有较明显的区别.

本文用向量 $v(T)=r(A_1), r(A_2), \cdots r(A_k))$ 来表示某个产品 T 的 k 个层面的分数，通过计算所有产品的向量值，可以得到产品特征属性评分矩阵

$$\boldsymbol{G}=\begin{bmatrix}(r_1(A_1), r_1(A_2), \cdots r_1(A_k)\\ (r_2(A_1), r_2(A_2), \cdots r_2(A_k)\\ \cdots\\ (r_t(A_1), r_t(A_2), \cdots r_t(A_k)\end{bmatrix}$$

$\boldsymbol{G}$ 表示所有 t 个产品在 k 个特征属性层面的分数，这个矩阵用于后续的用户偏好学习.

4　用户偏好建模

现实中用户在给予一个评论对象评分时，是基于如下事实给出的：被评论对象的各个特征属性层

面的好坏优劣,因此,是用户对这些特征属性层面的关注程度影响了他们对评论对象的评分,本文把用户对评论对象各个特征属性层面的关注程度称为用户偏好.

传统的基于用户的协同过滤算法只考虑了用户之间历史评分的相似度,也就是兴趣方面的相似关系,然而对于同一个感兴趣的东西来说,具有不同的关注偏好的用户可能具有不同的观点,也就是说不同的用户因不同的原因可能对同一个东西产生兴趣.本文从已抽取的评论中学习出用户的偏好,通过用户偏好来修正传统的基于用户的协同过滤算法的准确性.

4.1 用户偏好定义与量化

用户的偏好表达了用户对评论对象各个特征属性层面的关注程度,不同的用户可能有不同的偏好.如对手机,有的用户关注功能,有的用户关注价格与售后服务,而有的用户更关注外观与性能等.所以当用户要对手机进行整体评分时,他会主要根据自己关注的层面和偏好来进行评分.

本文将用户的关注偏好记为 $\boldsymbol{S}=[s_1,s_2,\cdots s_k]^{\mathrm{T}}$,其中 s_i 是代表用户对产品某一层面的偏好程度的实数.用户对产品的整体评分是产品各层面的分数与用户偏好的一次线性组合,即

$$r(t)=\boldsymbol{v}(T)\boldsymbol{S}, \tag{5}$$

式中:$r(t)$是用户在评论中对某产品给出的整体评分,$r(t)\in[1,5]$,$\boldsymbol{v}(T)$是产品 T 的各个特征属性层面的分数向量.

对于该用户参与的所有产品的评论,可以将其联立,得到

$$\begin{cases} r(t_1)=\boldsymbol{v}(T_1)\boldsymbol{S} \\ r(t_2)=\boldsymbol{v}(T_2)\boldsymbol{S} \\ \vdots \\ r(t_n)=\boldsymbol{v}(T_n)\boldsymbol{S} \end{cases}. \tag{6}$$

对于同一个用户,他的偏好 $\boldsymbol{S}=[s_1,s_2,\cdots s_k]^{\mathrm{T}}$ 在同类产品中是不变的,这个假设是合理的,因为对于一个用户其偏好在一段时间内通常是稳定的.

从式(6)可以看出这是一次线性回归问题,因为用户的偏好向量是 k 维的,所以只需要选取其中 k 条评论就可以解出用户的偏好向量.而选取 k 条评论有几种策略,比如选择用户最近的 k 条评论,或者对比较冷门产品的 k 条评论,或者比较热门的产品的 k 条评论,又或者随机选择 k 条评论等,本文采用随机选择的方法,并得到其矩阵表达式如下:

$$\begin{bmatrix} r(t_1) \\ r(t_2) \\ \vdots \\ r(t_k) \end{bmatrix}=\begin{bmatrix} r_1(A_1) & r_1(A_2) & \cdots & r_1(A_k) \\ r_2(A_1) & r_2(A_2) & \cdots & r_2(A_k) \\ \vdots & \vdots & \cdots & \vdots \\ r_k(A_1) & r_k(A_2) & \cdots r_k(A_k) & \end{bmatrix}\begin{bmatrix} s_1 \\ s_2 \\ \vdots \\ s_k \end{bmatrix}, \tag{7}$$

式中:矩阵

$$\begin{bmatrix} r_1(A_1) & r_1(A_2) & \cdots & r_1(A_k) \\ r_2(A_1) & r_2(A_2) & \cdots & r_2(A_k) \\ \vdots & \vdots & \cdots & \vdots \\ r_k(A_1) & r_k(A_2) & \cdots & r_k(A_k) \end{bmatrix}$$

可以从产品特征属性评分矩阵 $\boldsymbol{G}$ 中选取.

要使得 $\boldsymbol{S}=[s_1,s_2,\cdots s_k]^{\mathrm{T}}$ 有解,产品层面分数的矩阵必须可逆,所以在选择 k 条评论时始终要满足这个条件.

4.2 用户偏好相似度计算

由于用户的偏好都是 k 维向量,用户之间偏好相似度可以理解成 2 个用户的偏好向量在 k 维空间上夹角的余弦值,所以本文使用余弦相似度来求解 2 个用户之间在偏好方面的相似度,公式如下:

$$\mathrm{sim}^p(\boldsymbol{s}_i,\boldsymbol{s}_j)=\cos(\boldsymbol{s}_i,\boldsymbol{s}_j)=\frac{\boldsymbol{s}_i\cdot\boldsymbol{s}_j}{\|\boldsymbol{s}_i\|\ \|\boldsymbol{s}_j\|}. \tag{8}$$

式中:$\boldsymbol{s}_i$ 和 $\boldsymbol{s}_j$ 分别代表用户 i 和用户 j 的偏好向量,$\mathrm{sim}^p(\boldsymbol{s}_i,\boldsymbol{s}_j)$为两者之间的偏好相似度,$\mathrm{sim}^p(\boldsymbol{s}_i,\boldsymbol{s}_j)$的值域为$[0,1]$,其值越大则表示 2 个用户的偏好就越相似.

如果用户间的偏好越相似,他们对同一个产品的观点越趋于相似,在整体评分方面分数就越相近.例如,现在已知 3 个用户及其偏好向量,他们对同一部手机都做过整体评分,具体情况如表 7 所示.

表 7 相似用户对手机的整体评分

Tab.7 Overall rating of mobile phone by similar users

用户	偏好向量	手机特征层面分数	整体评分
用户 a	$\boldsymbol{s}_a$ (−1.654,−0.048,2.553,0.144)	$\boldsymbol{v}(T)$ (4.85,4.81,4.9,4.96)	5
用户 b	$\boldsymbol{s}_b$ (−0.619,−0.525,1.795,0.344)	$\boldsymbol{v}(T)$ (4.85,4.81,4.9,4.96)	5
用户 c	$\boldsymbol{s}_c$ (0.778,−3.896,5.076,−1.144)	$\boldsymbol{v}(T)$ (4.85,4.81,4.9,4.96)	4

观察用户 a 和用户 c 分别与用户 b 的偏好相似度,验证相似用户是否具有相似观点.通过计算得到用户 a 与用户 b 的偏好相似度 $\mathrm{sim}^p(\boldsymbol{s}_a,\boldsymbol{s}_b)=0.932\ 7$,用户 b 与用户 c 的偏好相似度 $\mathrm{sim}^p(\boldsymbol{s}_b,\boldsymbol{s}_c)=0.586\ 5$.从计算结果可知,用户 a 与用户 b 在偏好

方面更加相似，而且用户 a 与用户 b 都做了整体评分 5 分，而用户 b 与用户 c 在偏好方面的差距比较大. 因此，说明了偏好相似的用户对同一产品的观点也趋于相似.

5　结合相似度传递的协同过滤推荐

5.1　评分相似度

基于用户的(user-based)协同过滤算法，其主要思想是认为兴趣爱好相似的用户之间具有相似的评分行为，所以可通过用户对共同评价商品的评分，来计算用户之间的相似程度. 本文使用相关相似度 $\mathrm{sim}(u_a,u_b)$ 来表示用户评分间的相似程度.

$$\mathrm{sim}(i,j)=\frac{\Sigma_{c\in I}(R_{i,c}-\bar{R}_i)(R_{j,c}-\bar{R}_j)}{\sqrt{\Sigma_{c\in I}(R_{i,c}-\bar{R}_i)^2}\sqrt{\Sigma_{c\in I}(R_{j,c}-\bar{R}_j)^2}}. \tag{9}$$

式中：I 表示用户 i 和用户 j 共同评价过的项目集合，$R_{i,c}$ 和 $R_{j,c}$ 分别表示用户 i 和用户 j 对共同评分项目 c 的评分，$\bar{R}_i$ 和 $\bar{R}_j$ 分别表示用户 i 和用户 j 对所评分项目的评分均值.

相关相似度的计算依赖于用户之间共同评分的项目，共同评分的项目越多则相似度计算结果越准确. 共同评分的项目比较少时，则相似度度量存在一定的偶然性，比如 2 个用户都只买过同一个产品，且给予了相同的评分，根据相关相似度计算，则 2 个用户是完全相似的. 因此根据一个产品就判断 2 个用户完全相似，显然不合理. 为了消除这种偶然性带来的影响，Herlocker 等[20-21]等提出要增加一个关联权重因子来进行相似度计算. 在此基础上，Ma 等[22]也提出影响性权重的设置. 本文定义用户 u_a 与 u_b 之间共同评分的项目集合为 $I=I_{u_a}\cap I_{u_b}$，通过设定阈值 γ，与共同评分的项目数目 $|I|$ 进行比较，得到改进的相关相似度计算公式：

$$\mathrm{sim}(u_a,u_b)=\frac{\min(|I|,\gamma)}{\gamma}\times\mathrm{sin}(u_a,u_b) \tag{10}$$

式中：$\mathrm{sim}(u_a,u_b)$ 是通过式(9)计算的用户相关相似度. 从式中可以看出 $\frac{\min(|I|,\gamma)}{\gamma}\leqslant 1$，改进后的相似度 $\mathrm{sim}^r(u_a,u_b)$ 的值域仍然在[0,1]区间上.

5.2　综合相似度

传统协同过滤算法在计算用户相似度时，只考虑了用户之间在共同评分项目上整体评分的相似度，而未考虑用户在产品各个层面上关注程度的不同，因此评分的相似度不足以完整表达用户相似性. 本文考虑两者加权后的综合相似度，以此代替评分相似度来度量用户间的相似性，并应用到协同过滤的算法过程中. 综合相似度公式如下：

$$\mathrm{sim}(u_a,u_b)=\alpha\cdot\mathrm{sim}^r(u_a,u_b)+(1+\alpha)\cdot\mathrm{sin}^p(u_a,u_b). \tag{11}$$

式中：$\mathrm{sim}^r(u_a,u_b)$ 与 $\mathrm{sim}^p(u_a,u_b)$ 分别代表用户 u_a 与用户 u_b 的评分相似度与偏好相似度，而权重因子 α 的取值范围则在[0,1]区间上，α 的取值需要通过实验确定以使得何时算法精确性最高，结果见实验部分.

5.3　相似度传递

由于推荐系统中的项目数量往往十分庞大，而用户之间共同评价的项目去很少，因此就会导致部分用户之间无法计算相似度的情况，这就是传统协同过滤算法数据稀疏性问题.

关于相似度的传递目前还没有统一的计算方法，因此本文提出一种简单的传递方法，来计算用户间的相似度.

定义 4：相似度传递路径长度. 用户间的相似关系组成一个网络，其中结点代表用户，边代表用户之间相似关系. 相似度传递路径长度是指，在用户相似关系网络中，用户节点之间相似关系路径上边的数量.

相似度传递的规则如下：

1)若两用户间具有直接相似度，即两用户间有共同评价过的产品，则不需要传递.

2)如果两用户间不存在直接相似度，且在相似度传递路径长度小于阈值 L 的情况下，两用户间只存在一条传递路径，则两用户的相似度是此路径上的最小相似度乘以传递衰减因子.

传递衰减因子 β 的计算方法使用胡福华等[23]提出的定义，方法如下：

$$\beta=\frac{L-N+1}{L}. \tag{12}$$

式中：L 为传递路径长度的阈值，N 为当前传递路径的长度，β 随当前传递路径的增长而递减.

3)若两用户间不存在直接相似度，且在相似度传递路径长度小于阈值 L 的情况下，两用户间存在多条传递路径，则两用户的相似度为通过规则 2 计算得到的所有相似度的平均值.

4)若两用户间不存在直接相似度，且在传递路径长度阈值 L 范围内不存在连通路径，则不计算 2 个用户之间的相似度，其相似度为 0.

如图 2 所示，假设传递路径长度阀值设为 $L=3$，用户 A 到用户 D 有一条传递路径：A→C→D，此路径上最小相似度为 0.6，当前传递路径为 2，传递

衰减因子 $\beta=\frac{L-N+1}{L}=2/3$,用户 A 到用户 D 的间接相似度为 sim (A,D)=0.4.又如用户 A 到用户 E 之间有两条路径分别为:A→B→E 和 A→C→D→E,根据规则 1 分别计算得到两条路径的相似度为 0.4 和 0.167,则 sim (A,E)=0.283 5.再看用户 A 虽然与用户 G 之间存在路径,但是已经超过传递长度阀值,所以 2 个用户之间不计算相似度,相似度仍为 0.

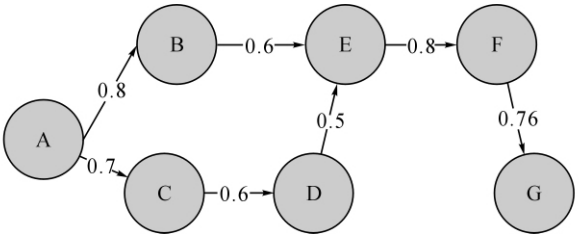

图 2 用户相似度网络图

Fig. 2 Network chart of user similarity

在计算过程中,本文定义了一个传递路径长度的阈值 L,当传递路径长度超过此阈值时,不进行相似度计算.这样做可避免传递过多,使得间接相似度出现较大偏差.经过实验,发现在数据稀疏的情况下,传递 1 到 2 步,即可以得到较好效果.

5.4 评分预测与产品推荐

在协同过滤算法中,通过目标用户相似邻居集合,可以计算目标用户对商品的预测评分.目标用户相似邻居指的是与目标用户具有相似爱好的其他用户,他们在预测用户对商品的评分时,提供了评分参考.传统的用户最近邻居选择方法有 2 种:

1) 选择与目标用户相似度最高的 X 个用户.

2) 选择相似度大于某个阈值的相似用户.

本文采用第 1 种方法,即选择与目标用户综合相似度最高的 X 个用户作为目标用户的邻居集.

本文将综合相似度代替原有评分相似度作为权重,评分预测公式如下:

$$P_{u,i}=\bar{R}_u+\frac{\sum_{m\in U}(R_{m,i}-\bar{R}_m)\cdot \mathrm{sim}\ (u,m)}{\sum_{m\in U}\mathrm{sim}\ (u,m)}. \quad (13)$$

式中:$\bar{R}_u$ 表示目标用户 u 对所评分项目的评分均值,U 表示最近邻居集,$R_{m,i}$ 表示邻居用户 m 对项目 i 的评分,R_m 表示邻居用户 m 对所评分项目的评分均值,sim (u,m) 表示目标用户 u 和邻居用户 m 的经过相似度传递后的综合相似度.

通过式(13)可以得到目标用户对某个商品的预测评分,再与用户实际评分进行比较,就可以衡量本文推荐算法的准确度.

6 实验结果及分析

6.1 实验数据准备

通过定制爬虫抓取京东商城上 800 多款手机 80 多万条评论,建立手机主题评论语料库.经过数据清理,本文提取了其中 894 个用户对 439 部手机的 16 835 条评论,时间跨度从 2008 年 12 月到 2011 年 12 月,每条评论包括用户对产品的评论和评分,所有用户均至少对 10 部手机做过评论,用户评分分布和用户共同评价产品数分布如图 3、4 所示.

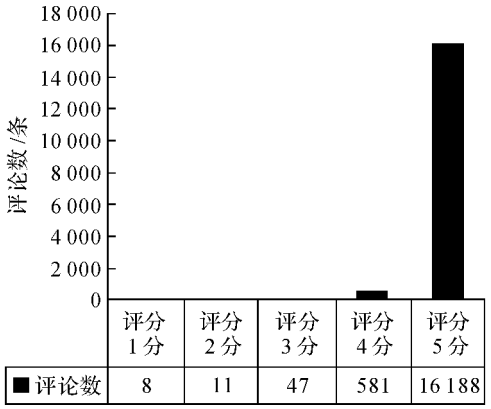

图 3 用户评分分布图

Fig. 3 Distribution figure of user ratings

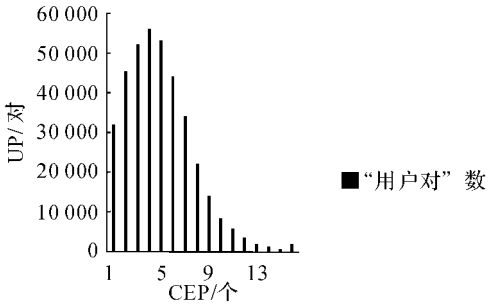

图 4 用户共同评价产品数分布图

Fig. 4 Number of products commonly evaluated by users

图 4 中,CEP/count of product 为 2 个用户间共同评价产品个数(count of commonly evaluated products by two users),UP/pair of users 为“用户对”数(pair of users).

本文采用数据全部来源于现实交易数据,可从图中看出,用户给出评分大部分都是 5 分,这一点表明仅使用用户评分记录进行产品推荐的局限性.

实验采用交叉验证的方法,先抽取实验数据集的 80%作为训练集,用以计算用户之间的相似度并计算预测评分,而剩下的 20%为测试集作为待预测的目标数据,用以衡量基于用户的协同过滤算法与

本文算法的评分预测精确度.

6.2　评测标准

预测评分的评价主要将预测得到的用户评分与用户实际对项目的评分进行比较. 平均绝对误差 MAE (mean absolute error)通常作为预测评分的评价标准，一般实验数据被分为训练集和测试集 2 个部分，训练集中的数据被用来预测测试集中的评分，然后与测试集的实际评分进行对比，计算得到 MAE 值，即预测得到的预测分值与项目实际评分之间绝对平均误差. 假设预测得到的项目评分为 $\{p_1, p_2, \cdots, p_k\}$，而项目实际评分为 $\{q_1, q_2, \cdots, q_k\}$，则 MAE 的计算公式如下：

$$\mathrm{MAE}=\frac{\sum_{i=1}^{k}|p_i-q_i|}{N},\ i=1,2,\cdots,K. \quad (14)$$

式中：K 表示预测项目的个数，MAE 越小表示误差越小，预测质量越好.

6.3　实验结果

实验在设定不同的最近邻居数(count of nearest neighbor-CNN) X 为 20，40，60 的条件下，测试公式(11)中评分相似度与偏好相似度权重因子取不同值时，本文算法的预测评分与实际评分的 MAE 值，得到数据，如图 5 所示.

结果表明，随着权重 α 从 0.1 逐渐增大的 0.9，算法的预测评分 MAE 值曲线先下降后逐渐升高，当 $\alpha=0.8$ 时，即表明用户综合相似度中评分相似度占 80%，而偏好相似度占 20%，算法的准确度相对较高. 这说明，用户偏好相似度是以用户兴趣相似为前提的，只有当用户之间兴趣比较相似时，通过用户偏好的相似性修正才可以改善预测评分的准确度，而如果以用户偏好直接作为相似度的主导部分，反而导致准确度的降低，所以用户偏好相似度可以看作是用户评分相似度的一种改善和修正. 因此可以肯定用户偏好的引入能改善传统协同过滤算法中用户之间的相似性，进而提高预测评分的准确性. 为了获得较好的算法结果，在接下来的实验中，α 将取值为 0.8.

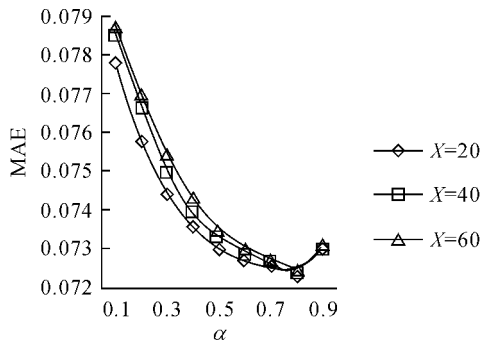

图 5　权重因子对 MAE 值的影响

Fig. 5　Impact of weight factor to MAE value

将本文提出的算法(PP-CF)与传统的基于用户的协同过滤算法(CF)作比较. 将相似度传递的路径长度 L 阈值设置为 2，并选择用户的最近邻居个数从 5 到 60 依次递增，结果如图 6 所示. 从图中可以看到，随着最近邻居的增加，2 种算法的 MAE 值随之下降. 但是整体上本文提出 PP-CF 的评分预测准确度要好于 CF 算法，并且在最近邻居数为 20 左右达到最优，之后一直保持稳定的趋势. 其中的主要原因是，随着数据稀疏程度的增加，能够和当前用户计算相似度的用户越来越少，传统的基于用户的协同过滤算法在选择最近邻居时，没有足够的邻居用来进行评分预测. 而本文提出的 PP-CF 算法通过相似度传递较好的解决了这个问题.

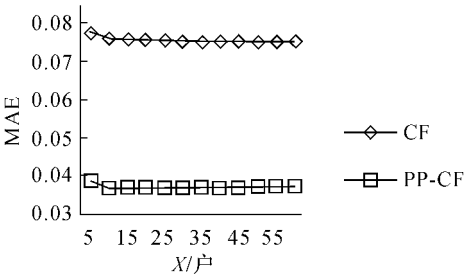

图 6　与传统的基于用户的协同过滤算法的比较

Fig. 6　Comparison with traditional user-based collaborative filtering algorithm

另外，近年有学者提出将人们相互之间的信任关系运用到传统的协同过滤推荐技术中去，用来解决传统协同过滤算法的数据稀疏和推荐精确性问题. 因此，本文与卢竹兵[24]提出的基于信任关系的协同过滤算法进行比较，对比结果如图 7、8 所示：

通过本文提出算法(PP-CF)、基于信任关系的协同过滤算法(TS-CF)与及传统协同过滤算法(CF)的比较，可以看出，本文提出的算法与基于信任关系的协同过滤算法在准确性方面都较传统的协

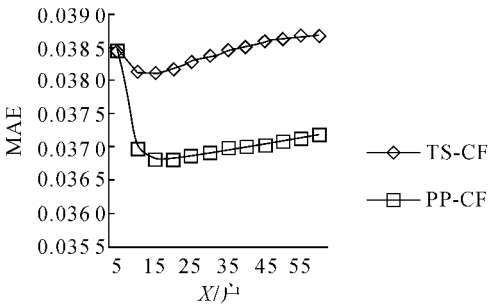

图 7　与基于信任关系的协同过滤算法的比较

Fig. 7　Comparison with trust relationship based collaborative filtering algorithm

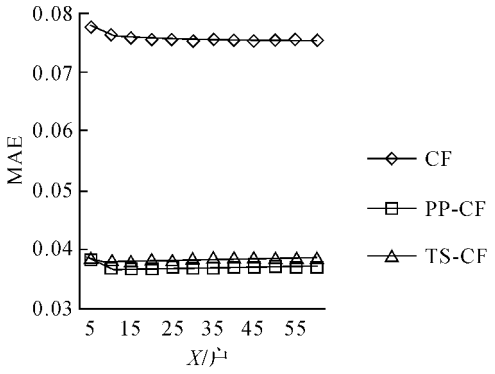

图8 3种算法的比较

Fig. 8 Comparison of three algorithms

同过滤算法有明显的提高,其中本文提出的算法在准确性方面略胜一筹.

本文算法与基于信任关系的协同过滤算法从不同角度做了改进,并取得明显提高.基于信任关系的协同过滤算法在计算邻居用户相似度时引入信任概念,从而产生更高推荐准确度,但在计算相似度时仍然存在数据稀疏的问题,使得部分相似度计算存在比较大的误差,从而使信任度的计算不准确.

7 结 语

本文提出基于评论挖掘的产品推荐算法,通过用户评论挖掘,得到产品各个层面的客观分数,据此建立用户偏好模型,并将用户偏好相似度与评分相似度加权得到综合相似度,以此修正传统协同过滤方法的片面性.同时,通过相似度传递的方法来缓解数据稀疏性问题,推荐的准确性比传统的协同过滤算法有较大的提高.

从方法实用性上看,本文提出的推荐算法有较为广泛的使用面,尤其是针对当下电子商务网站的评价体系结构.对仅有用户评分及主观描述评价的数据,本文方法有良好的通用性和有效性.

当前推荐技术研究较为成熟,但评论挖掘方面的研究仍处于探索阶段,评论挖掘与推荐技术的结合方向的研究还有待发展.本文从用户偏好的切入点入手,重点研究了评论挖掘技术与协同过滤推荐技术的结合方案.今后工作将围绕以下几方面展开:

(1)由于中文评论挖掘算法优劣将直接影响提取特征词与情感词的查全率与准确性,从而影响产品层面划分与各层面分数的客观性.所以,对评论挖掘算法,将采用更加优化的方法以获得更好提取结果,从而提高本文算法准确性.

(2)提高算法性能的研究.本文提出算法,在产品特征词提取、用户偏好学习和相似度传递等过程中,一般需要采用离线与在线计算相结合方式,以保证算法实时性.而在大规模数据场景中,可以通过分布式计算充分利用系统资源以提高算法性能.

参考文献(References):

[1] SPINELLIS D, RAPTIS K. Component mining: A process and its pattern language [J]. **Information and Software Technology**, 2000, 42(9): 609-617.

[2] GOLDBERG D, NICHOLS D, OKI B M, et al. Using collaborative filtering to weave an information tapestry [J]. **Communications of the ACM**, 1992, 35(12): 61-70.

[3] KONSTAN J A, MILLER B N, MALTZ D, et al. Grouplens: applying collaborative filtering to Usenet news [J]. **Communications of the ACM**, 1997, 40(3): 77-87.

[4] SARWAR B, KARYPIS G, KONSTAN J, et al. Application of dimensionality reduction in recommender system-a case study [R]. **Minnesota Univ Minneapolis Dept of Computer Science**, 2000.

[5] 邓爱林,左子叶,朱扬勇,等. 基于项目聚类的协同过滤推荐算法[J]. 小型微型计算机系统. 2004, 25(9): 1665-1670.
DENG Ai-lin, ZUO Zi-ye, Zhu Yang-yong, et al. Collaborative filtering recommendation algorithm based on item clustering [J]. **Mini-micro Systems**, 2004, 25(9): 1665-1670.

[6] AGGARWAL C C, WOLF J L, WU K L, et al. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering [C]// **Proc. Fifth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining.** San Diego: ACM, 1999: 201-212.

[7] HU M, LIU B. Mining opinion features in customer reviews [C]// **Proc. of AAAI 2004**. San Jose: AAAI, 2004: 755-760.

[8] ETZIONI O, CAFARELLA M, DOWNEY D, et al. Unsupervised named-entity extraction from the web: An experimental study [J]. **Artificial Intelligence**, 2005, 165(1): 91-134.

[9] TURNEY P D. Thumbs up or thumbs down?: semantic orientation applied to unsupervised classification of reviews [C]// **Proc. of the** 40**th Annual Meeting on Association for Computational Linguistics.** Philadelphia: Association for Computational Linguistics, 2002: 417-424.

[10] RILOFF E., WIEBE J. Learning extraction patterns for subjective expressions [C]// **Proc. of the 2003 conference**

on Empirical methods in NLP. Sapporo：Association for Computational Linguistics，2003：105－112.

[11] **娄德成，姚天昉. 汉语句子语义极性分析和观点抽取方法的研究**[J]. **计算机应用**. 2006，26(11)：2622－2625.
LOU De-cheng，YAO Tian-fang. Semantic polarity analysis and opinion mining on Chinese review sentences [J]. **Computer Applications**，2006，26(11)：2622－2625.

[12] **姚天昉，聂青阳，李建超，等. 一个用于汉语汽车评论的意见挖掘系统**[C]// **中国中文信息学会二十五周年学术会议论文集. 北京：清华大学出版社**，2006：260－281.
YAO Tian-Fang，NIE Qing-yang，Li Jian-chao，et al. An opinion mining system for Chinese automobile reviews [C]// **Proc. of 25th Anniversary of Chinese Information Processing Society of China**. BeiJing：TsingHua University Press，2006：260－281.

[13] **姚天昉，娄德成. 汉语语句主题语义倾向分析方法的研究**[J]. **中文信息学报**. 2007，21(5)：73－79.
YAO Tian-fang，LOU De-cheng. Research on semantic orientation analysis for topics in Chinese sentences [J] **Journal of Chinese Information Procession**，2007，21(5)：73－79.

[14] ADOMAVICIUS G，TUZHILIN A. Toward the next generation of recommender systems：A survey of the state-of-the-art and possible extensions [J]. **Knowledge and Data Engineering，IEEE Transactions on**，2005，17(6)：734－749.

[15] WIETSMA R T A，RICCI F. Product reviews in mobile decision aid systems [C]// **The 3rd International Conference on Pervasive Computing（PERVASIVE 2005**）. Munich：PERMID，2005：15－18.

[16] RICCI F，WIETSMA R T A. Product reviews in travel decision making [J]. **Information and Communication Technologies in Tourism**，2006：296－307.

[17] ACIAR S，ZHANG D，SIMOFF S，et al. Recommender system based on consumer product reviews [C]// **Proc. of the 2006 IEEE/WIC/ACM International Conference on Web Intelligence.** Hong Kong：IEEE Computer Society，2006：719－723.

[18] **吴亚峰. 基于评论挖掘的协同过滤推荐算法研究**[D]. **杭州：浙江大学**，2012.
WU Ya-feng，Research of a collaborative filtering recommendation algorithm based on review mining[D]. Hangzhou：Zhejiang University，2012.

[19] **赵文婧. 产品描述词及情感词抽取模式的研究**[D]. **北京邮电大学**，2010.
ZHAO Wen-jing. Research on Extraction patterns of product description words and sentiment words [D]，Beijing University of Posts and Telecommunications，Beijing：2010.

[20] HERLOCKER J，KONSTAN J A.，RIEDL J. An empirical analysis of design choices in neighborhood-based collaborative filtering algorithms [J]. **Information retrieval**，2002，5(4)：287－310.

[21] MCLAUGHLIN M R，HERLOCKER J L. A collaborative filtering algorithm and evaluation metric that accurately model the user experience [C]// **Proc. of the 27th Annual International ACM SIGIR Conference on Research and development in Information retrieval.** University of Sheffield：ACM，**2004**：**329－336**.

[**22**] MA H，KING I，LYU M R. Effective missing data prediction for collaborative filtering [C]// **Proc. of the 30th Annual International ACM SIGIR Conference on Research and Development in Information retrieval.** Amsterdam：ACM，2007：39－46.

[23] **胡福华，郑小林，干红华. 基于相似度传递的协同过滤算法**[J]. **计算机工程**. 2011(10)：50－51.
HU Fu-hua，ZHENG Xiao-lin，GAN Hong-hua. Collaborative filtering algorithm based on similarity propagation [J]. **Computer Engineering**，2011(10)：50－51.

[24] **卢竹兵. 基于信任关系的协同过滤推荐策略研究**[D]. **重庆：西南大学**. 2008：26－41.
LU Zhu-bing. Study on trust relationship based collaborative，filtering recommender strategy [D]. Chongqing：Southwest University，2008：26－41.