专题评述

个性化推荐系统的研究进展*

刘建国[1,2] 周 涛[1,2] 汪秉宏[1,3**]

1. 中国科学技术大学, 近代物理系, 理论物理研究所, 合肥 230026; 2. Department of Physics, University of Fribourg, Switzerland, CH-1700; 3. 上海系统科学研究院 复杂适应系统研究所, 上海理工大学, 上海 200093

摘要 互联网技术的迅猛发展把我们带进了信息爆炸的时代. 海量信息的同时呈现, 一方面使用户很难从中发现自己感兴趣的部分, 另一方面也使得大量少人问津的信息成为网络中的"暗信息", 无法被一般用户获取. 个性化推荐系统通过建立用户与信息产品之间的二元关系, 利用已有的选择过程或相似性关系挖掘每个用户潜在感兴趣的对象, 进而进行个性化推荐, 其本质就是信息过滤. 个性化推荐系统不仅在社会经济中具有重要的应用价值, 而且也是一个非常值得研究的科学问题. 事实上, 它是目前解决信息过载问题最有效的工具. 文中根据推荐算法的不同, 分别介绍了协同过滤系统, 基于内容的推荐系统, 混合推荐系统, 以及最近兴起的基于用户—产品二部图网络结构的推荐系统. 并结合这些推荐系统的特点以及存在的缺陷, 提出了改进的方法和未来可能的若干研究方向. 推荐系统的研究受到了信息科学、计算数学、统计物理学、认知科学等多学科的关注, 它与管理科学、消费行为等研究也密切相关. 能够为不同学科领域的科研工作者研究推荐系统提供借鉴, 有助于我国学者了解该领域的主要进展.

关键词 推荐系统 个性化推荐 协同过滤 基于内容的推荐 基于网络的推荐

随着Internet的迅猛发展, 接入Internet的服务器数量和World-Wide-Web上的网页的数目都呈现出指数增长的态势. 互联网技术的迅速发展使得大量的信息同时呈现在我们面前, 例如, Netflix上有数万部电影, Amazon上有数百万本书, Del icia us上面有超过10亿的网页收藏, 如此多的信息, 别说找到自己感兴趣的部分, 即使是全部浏览一遍也是不可能的. 传统的搜索算法只能呈现给所有的用户一样的排序结果, 无法针对不同用户的兴趣爱好提供相应的服务. 信息的爆炸使得信息的利用率反而降低, 这种现象被称之为信息超载. 个性化推荐, 包括个性化搜索, 被认为是当前解决信息超载问题最有效的工具之一. 推荐问题从根本上说就是代替用户评估它从未看过的产品[1—5]. 这些产品包括书、电影、CD、网页、甚至可以是饭店、音乐、绘画等等——是一个从已知到未知的过程.

个性化推荐研究直到20世纪90年代才被作为一个独立的概念提出来[1,2]. 最近的迅猛发展, 来源于Web2 0技术的成熟. 有了这个技术, 用户不再是被动的网页浏览者, 而是成为主动参与者[3]. 在一个实际的推荐系统中需要推荐的产品可能会有成千上万, 甚至超过百万, 例如Amazon, eBay, Youtube等, 用户的数目也会非常巨大. 准确、高效的推荐系统可以挖掘用户潜在的消费倾向, 为众多的用户提供个性化服务. 在日趋激烈的竞争环境下, 个性化推荐系统已经不仅仅是一种商业营销手

2008-06-23收稿, 2008-07-09收修改稿

* 国家重点基础研究发展计划(批准号: 2006CB705500)和国家自然科学基金(批准号: 10635040, 60744003, 10532060, 10472116)资助项目

** 通信作者, E-mail: bhwang@ustc edu cn

段，更重要的是可以增进用户的黏着性．个性化推荐系统已经给电子商务领域带来巨大的商业利益．据VentureBeat统计，Amazon的推荐系统为其提供了35%的商品销售额．尽管现有的推荐系统已经在电子商务等领域取得了巨大的成功，但是还需要在不同领域研究人员的努力下进一步完善和发展[5]．一个典型的例子就是Netflix开出100万美元的奖金，奖励给能把他们网站的产品推荐精确度提高10%的人．Netflix的竞赛只是从推荐准确性的角度评价算法，事实上，还有很多的评价指标可以度量推荐算法的表现，因此也可以从多个角度对算法进行改进．当然，无论从哪个角度改进，都需要从整体入手对推荐系统的体系结构有一个完整的认识．

一个完整的推荐系统由3个部分组成：收集用户信息的行为记录模块，分析用户喜好的模型分析模块和推荐算法模块．行为记录模块负责记录用户的喜好行为，例如问答、评分、购买、下载、浏览等．问答和打分的信息相对好收集，然而有的用户不愿意向系统提供这些信息，那么就需要通过其他方式对用户的行为进行分析，例如购买、下载、浏览等行为．通过这些用户的行为记录分析用户的潜在喜好产品和喜欢程度．这就是模型分析模块要完成的工作．模型分析模块的功能能够对用户的行为记录进行分析，建立合适的模型来描述用户的喜好信息．最后是推荐算法模块，利用后台的推荐算法，实时地从产品集合中筛选出用户感兴趣的产品进行推荐．其中，推荐算法模块是推荐系统中最为核心的部分．

本文简要介绍一些文献和实际系统中采用的推荐算法和不同类型的推荐系统．根据推荐算法的不同，推荐系统可以分为如下几类：协同过滤(collaborative filtering)系统；基于内容(content-based)的推荐系统；混合(hybrid)推荐系统以及最近兴起的基于用户—产品二部图网络结构(network-based)的推荐系统．最后，我们将指出这些系统存在的缺陷和未来可能的若干研究方向．

1 协同过滤系统

协同过滤系统是第一代被提出并得到广泛应用的推荐系统．其核心思想可以分为两部分：首先，是利用用户的历史信息计算用户之间的相似性；然后，利用与目标用户相似性较高的邻居对其他产品的评价来预测目标用户对特定产品的喜好程度．系统根据这一喜好程度来对目标用户进行推荐．协同过滤推荐系统最大的优点是对推荐对象没有特殊的要求，能处理音乐、电影等难以进行文本结构化表示的对象．

协同过滤系统是目前应用最为广泛的个性化推荐系统，其中Grundy被认为是第一个投入应用的协同过滤系统[6]．Grundy系统可以建立用户兴趣模型，利用模型给每个用户推荐相关的书籍．Tapestry邮件[7]处理系统人工确定用户之间的相似度，随着用户数量的增加，其工作量将大大增加，而且准确度也会大打折扣．GroupLens[8]建立用户信息群，群内的用户可以发布自己的信息，依据社会信息过滤系统计算用户之间的相似性，进而向群内的其他用户进行协同推荐．Ringo[9]利用相同的社会信息过滤方法向用户进行音乐推荐．其他利用协同过滤方法进行推荐的系统还有Amazon.com的书籍推荐系统[10]，Jester的笑话推荐系统[11]，Phoaks的WWW信息推荐系统[12]，等等．

协同过滤推荐系统的算法可以分为两类：基于记忆(memory-based)[13—15]的和基于模型的(model-based)的算法[16—20]．基于记忆的算法根据系统中所有被打过分的产品信息进行预测．设$C=\{c_1, c_2, \cdots, c_N\}$为用户集合，$S=\{s_1, s_2, \cdots, s_M\}$为所有的产品集合．设$r_{c,s}$为用户$c$对产品$s$的打分，这个打分是不知道的，需要通过算法去预测．在协同过滤系统中，用户c对产品s的打分$r_{c,s}$通过其他用户对s的打分计算而得到．设$\hat{C}$为与用户c相似度比较高的用户集，预测$r_{c,s}$的函数形式有：

$$r_{c,s}=\frac{1}{N}\sum_{c\in\hat{C}} r_{c,s} \tag{1a}$$

$$r_{c,s}=k\sum_{c\in\hat{C}}\mathrm{sim}(c,\bar{c})\cdot r_{c,s} \tag{1b}$$

$$r_{c,s}=\bar{r}_c+k\sum_{c\in\hat{C}}\mathrm{sim}(c,\bar{c})\cdot(r_{c,s}-\bar{r}_c) \tag{1c}$$

其中k为一个标准化因子，通常$k=1/\sum_{c\in\hat{C}}|\mathrm{sim}(c,\bar{c})|$，$\mathrm{sim}(i,j)$表示用户$i$和$j$之间的相似性．用户$c$的平均

打分 $\bar{r}_c$ 定义为 $\bar{r}_c = (1/|s_c|)\sum_{s\in S_c} r_{a,s}$，其中 $S_c = \{s \in S \mid r_{a,s} \neq 0\}$. 如公式(1a)所示，最简单的计算方法就是直接计算邻居打分的平均值. 然而，最常用的计算方法却是加权平均(1b)，其中 $\mathrm{sim}(c, \bar{c})$ 为用户 c 和 $\bar{c}$ 的相似性. 如果用户 c 和 $\bar{c}$ 越像，$r_{a,s}$ 将有更大的权重用于预测 $r_{a,s}$. 由于(1b)应用了标准化因子 k，因此(1b)可以用于计算不同的推荐系统中的用户相似性度量. 应用(1b)时，尽管使用了加权和，但是并没有考虑不同用户的评价尺度不一样的问题. (1c)通过只考虑不同用户平均喜好程度的偏差，克服了评判尺度不一致的缺点，一般而言具有比(1b)更高的精确度. 偏好过滤[21—24]是另一个克服用户打分尺度不一的方法. 该方法注重于预测用户的相对偏好，而不是评分绝对值.

协同过滤系统中已经采用了很多种方法计算用户之间的相似度[1, 9, 13, 25—28]. 这些算法中，大部分都是基于用户对共同喜好产品的打分. 两个最常用的方法是 Pearson 相关性和夹角余弦，它们都定义用户 x 和 y 共同打过分的产品集合为：$S_{xy} = S_x \cap S_y$. 基于图论的方法[29]可以不用计算所有用户 y 的 S_{xy} 而直接确定 x 的最近邻. 用户 x 和 y 之间的 Pearson 相关性定义为[1, 9]

$$\mathrm{sim}(x, y) = \frac{\sum_{s\in S_{xy}} (r_{x,s} - \bar{r}_x)(r_{y,s} - \bar{r}_y)}{\sqrt{\sum_{s\in S_{xy}} (r_{x,s} - \bar{r}_x)^2 \sum_{s\in S_{xy}} (r_{y,s} - \bar{r}_y)^2}} \tag{2}$$

而在夹角余弦方法中[13, 25]，用户 x 和 y 都用 m 维向量表示，两个向量之间的相似性可以通过计算它们之间的余弦值得到

$$\mathrm{sim}(x, y) = \cos(\boldsymbol{x}, \boldsymbol{y}) = \frac{\boldsymbol{x} \cdot \boldsymbol{y}}{\|\boldsymbol{x}\|_2 \times \|\boldsymbol{y}\|_2} = \frac{\sum_{s\in S_{xy}} r_{x,s} r_{y,s}}{\sqrt{\sum_{s\in S_{xy}} r_{x,s}^2 \sum_{s\in S_{xy}} r_{y,s}^2}} \tag{3}$$

其中 $\boldsymbol{x} \cdot \boldsymbol{y}$ 表示两个向量的点积. 不同的系统可以采用不同的相似性计算方法以使得预测评分结果尽可能准确. 由于用户的兴趣和爱好是随时间变化的，所以一个普遍采用的策略就是提前计算所有用户的相似性 $\mathrm{sim}(x, y)$，隔一段时间进行一次更新. 用户需要推荐时，可以用事先计算好的相似性结果进行有效推荐.

许多改进算法已经被广泛研究并且应用到标准的相关性计算和夹角余弦公式中. 例如缺席投票(default voting)，事例引申(case amplification)[13]和加权优势预测[14, 15]等. 其中，缺席投票是基于记忆方法的一种扩展. 如果用户明确评分的产品数目很少，上面提到的算法得到的用户相似度都不准确. 原因在于这种相似性的计算是基于用户 x 和 y 共同评过分的产品集合. 实证数据表明，如果给一些没有打分的产品赋予一些缺省的打分值，那么预测分数的准确性将大幅度提高[13, 26]. Sarwar 等[25]提出应用相关性和夹角余弦方法计算产品之间的相似性. 这个思想被 Deshpande 和 Karypis 推广到基于产品相似性的 top-N 推荐算法中[30]，即在进行推荐的时候只考虑相似度最高的 N 个产品，并非所有的产品. 实验证明这种方法不仅比传统的基于用户邻居的推荐算法快 1—2 个数量级，而且具有更好的推荐准确性. Chen 和 Cheng[27]利用用户产品列表中的先后次序计算用户之间的相似性，排名靠前的产品在计算用户相似性的时候具有较高的权重. 而 Yang 和 Gu[28]提出利用用户的行为信息构建用户的兴趣点，利用兴趣点计算用户之间的相似性. 实验证明，这种方法比经典的协同过滤算法的推荐结果要好. 文献 [25, 30] 的结果表明基于产品相似性的算法能够比基于用户相似性的算法得到更好的计算结果.

基于模型的算法收集打分数据进行学习并推断用户行为模型，进而对某个产品进行预测打分. 基于模型的协同过滤算法和基于记忆的算法的不同在于，基于模型的方法不是基于一些启发规则进行预测计算，而是基于对已有数据应用统计和机器学习得到的模型进行预测. Breese 等[13]提出一个基于概率的协同过滤算法，其计算打分的公式如下

$$r_{a,s} = E(r_{a,s}) = \sum_{i=0}^{n} i \times \Pr(r_{a,s} = i \mid r_{a,s\bar{s}} \in S_c) \tag{4}$$

上式假设打分值为0到 n 之间的整数值. 概率 Pr 表

示基于用户以前的打分，用户要给产品 s 打指定分数的概率. 为了估计概率，Breese 等[13] 提出了两个选择概率模型：聚类模型和 Bayes 网络. 在第一个模型中，假设用户的打分彼此独立，偏好相似的用户聚集成类，给定用户类的标号. 在 Bayes 网络中，类的数量和模型参数可以从已有数据中学习得到. Bayes 网络中的点由一个领域里的产品表示，点的状态对应着每个产品的打分值. 网络的结构和条件概率从已有数据中学习得到. 这个方法的缺陷就是每个用户只能属于一个类，而一些推荐系统中如果用户可以属于多个类或许会更好一些.

其他基于模型的协同过滤推荐系统有概率相关模型[17]，极大熵模型[20]，线性回归[25]，基于聚类的 Gibbs 抽样算法[31]，Bayes 模型[32]，等等. 最近，大量的研究试图从更复杂的概率模型中建立推荐过程模型. 例如，Shani 等[33] 把推荐过程看做基于 Markov 决策过程的序列决策过程，利用已有信息预测用户以后的喜好产品的概率，在找到相应的产品进行推荐. 其他的概率模型技术包括概率潜层语义分析[18,34,35]，语义生成模型(aspect model)[19]. 另外，Kumar 等[36] 用一个简单的概率模型说明每个用户相对小的数据在协同过滤中是非常重要的. Yu 等[37] 从输入数据处理的角度提出了改进协同过滤的其他方法，包括除噪音技术，选择用户打分集技术，冗余度分析和打分数据的稀疏性处理等. 数值结果显示这些方法可以提高基于模型的协同过滤算法的准确性和效率. Yu 等[37] 还提出了输入选择技术，可以解决基于模型的算法需要对大规模数据进行学习的问题. Manouselis 和 Costopoulou[38] 提出了多准则协同过滤推荐系统，可以对具有多重衡量指标的产品进行推荐. Chen 等[39] 提出了群体协同过滤推荐系统，该系统对群体而非个人进行推荐.

总结起来，协同过滤系统因为有以下的优点，在实际系统中得到了广泛的应用.

(1) 具有推荐新信息的能力，可以发现用户潜在的但自己尚未觉察的兴趣偏好.

(2) 能够推荐艺术品、音乐、电影等难以进行内容分析的产品.

虽然协同过滤推荐系统得到了广泛的应用，但是也面临很多问题，例如如何对新用户进行推荐或如何推荐新产品给用户(冷启动问题)，打分稀疏性问题，算法可扩展性问题等. 另外，基于用户的协同推荐算法随着用户数量的增多，计算量成线性加大，其性能会越来越差. 因此有的推荐系统采用基于产品相似性的协同过滤算法，在产品的数量相对稳定的系统中，这种方法是很有效的，例如 Amazon 的书籍推荐系统[10]. 但是对于产品数量不断增加的系统，例如 Del lici us 系统，这种方法是不适用的. 在 Web 应用中，响应速度是影响用户体验最重要因素之一，这极大地限制了基于用户的协同过滤技术在实际系统中的使用. Amazon 更多地使用了基于产品的协同过滤技术，而且随着 Amazon 的成功，基于产品的方法也大为流行起来.

2 基于内容的推荐系统

历史上，最初的基于内容的推荐(content-based recommendation)是协同过滤技术的延续与发展，它不需要依据用户对项目的评价意见，而是依据用户已经选择的产品内容信息计算用户之间的相似性，进而进行相应的推荐. 随着机器学习等技术的完善，当前的基于内容的推荐系统可以分别对用户和产品建立配置文件，通过分析已经购买(或浏览)过的内容，建立或更新用户的配置文件. 系统可以比较用户与产品配置文件的相似度，并直接向用户推荐与其配置文件最相似的产品. 例如，在电影推荐中，基于内容的系统首先分析用户已经看过的打分比较高的电影的共性(演员、导演、风格等)，再推荐与这些用户感兴趣的电影内容相似度高的其他电影. 基于内容的推荐算法的根本在于信息获取[40,41] 和信息过滤[42]. 因为在文本信息获取与过滤方面的研究较为成熟，现有很多基于内容的推荐系统都是通过分析产品的文本信息进行推荐.

在信息获取中，表征文本最常用的方法就 TF-IDF 方法[41]. 该方法的定义如下：设有 N 个文本文件，关键词 k_i 在 n_i 个文件中出现，设 f_{ij} 为关键词 k_i 在文件 d_j 中出现的次数，那么 k_i 在 d_j 中的词频 TF_{ij} 定义为

$$TF_{ij} = \frac{f_{ij}}{\max_z f_{zj}} \tag{5}$$

其中分母的最大值可以通过计算 d_j 中所有关键词

k_z 的频率得到. 在许多文件中同时出现的关键词对于表示文件的特性, 区分文件的关联性是没有贡献的. 因此 TF_{ij} 与这个关键词在文件中出现数的逆(IDF_i)一起使用, IDF_i 的定义为

$$IDF_i = \log \frac{N}{n_i} \tag{6}$$

那么, 一个文件 d_j 可以表示为向量 $\boldsymbol{d}_j = (w_{1j}, w_{2j}, \cdots, w_{kj})$, 其中

$$w_{ij} = \frac{f_{ij}}{\max_z f_{zj}} \log \frac{N}{n_i} \tag{7}$$

设 Content(s)为产品 s 的配置文件, 也就是一些描述产品 s 特性的词组集合. 通常 Content(s)可以从产品的特征描述中提取计算得到. 在大多数的基于内容的推荐系统中, 产品的内容常常被描述成关键词——Fab 系统[43] 就是一个典型的例子. Fab 是一个网页推荐系统, 系统中用一个网页中最重要的100个关键词来表征这个网页. Syskill 和 Webert 系统[44] 用128个信息量最多的词表示一个文件. 基于内容的系统推荐与用户过去喜欢的产品最为相似的产品[43—45], 即不同候选产品与用户已经选择的产品进行对比, 推荐匹配度最好的产品. 或者直接向用户推荐与用户配置文件最为相似的产品. 设 UserProfile(c)为用户 c 的配置文件, UserProfile(c)可以用向量(w_{c1}, w_{c1}, $\cdots$, w_{ck})表示, 其中每个分量 w_{ci} 表示关键词 k_i 对用户 c 的重要性. 用户和产品都可以利用 TF-IDF 公式表示为 $\boldsymbol{w}_c$ 和 $\boldsymbol{w}_s$, 在基于内容的系统中, $r_{c,s}$ 常被定义为:

$$r_{c,s} = \text{score}(\text{UserProfile}(c), \text{Content}(s)). \tag{8}$$

$r_{c,s}$ 可以利用向量 $\boldsymbol{w}_c$ 和 $\boldsymbol{w}_s$ 表示成一个值, 例如夹角余弦方法[40, 41]:

$$r_{c,s} = \cos(\boldsymbol{w}_c, \boldsymbol{w}_s) = \frac{\boldsymbol{w}_c \cdot \boldsymbol{w}_s}{\|\boldsymbol{w}_c\|_2 \times \|\boldsymbol{w}_s\|_2} \tag{9}$$

除了传统的基于信息获取的推荐方法之外, 一些实际系统中还采用了其他技术, 例如 Bayes 分类[44—46]、聚类分析、决策树、人工神经网络[45] 等. 这些算法不同于基于信息获取方法的地方在于, 算法不是基于一个函数公式来进行推荐, 而是利用统计学习和机器学习技术从已有的数据中通过分析得到模型, 基于模型进行推荐. 例如, 利用 Bayes 分类器对网页进行分类[44—46]. 这种分类器可以用来估计一个网页 P_j 属于某个类 C_i 的概率. 给出这个网页中的关键词 k_{1j}, k_{2j}, $\cdots$, k_{nj}, 得到这些关键词属于 C_i 类的概率. 虽然关键词彼此相互独立的假设条件很不切实际, 但是这种分类方法在实际系统中仍然有高的分类准确率[45].

基于内容的推荐系统中, 用户的配置文件构建与更新是其中最为核心的部分之一, 也是目前研究人员关注的焦点. 例如 Somlo 和 Howe[47] 以及 Zhang 等[48] 提出了利用自适应过滤技术更新用户配置文件. 首先, 利用用户的喜好信息构建配置文件, 把用户的兴趣点归纳为几个主题文件. 进而在连续的 Web 文件流中依次对比 Web 的文本内容与主题文件的相似度, 选择性地把相似度较高的 Web 展示给用户并更新用户的配置文件. 进一步地, Robertson 和 Walker[49] 以及 Zhang 等[50] 在自适应过滤的基础上提出了最佳匹配度阈值设定算法. 首先还是在用户的配置文件中建立一些问题集, 系统利用已有数据与用户配置文件相似度的概率分布确定一个最佳阈值, 使得系统可以最大程度地区分与用户的配置文件相关和不相关的文件. 只有与用户配置文件的相似度大于最佳阈值的文件才能影响到用户配置文件的更新. 这种方法不仅可以进一步提高算法的精确性, 而且可以大大提高系统的运行效率.

通常用户的配置文件都是由一些关键词表示, 如果利用图论的索引方法可以节约存储空间. 然而, 当用户的兴趣爱好发生改变的时候, 配置文件更新的代价是很大的. Chang 等[51] 通过区分长期感兴趣与短期感兴趣的关键词, 赋予短期感兴趣的关键词更高的权重, 在此基础上建立新的关键词更新树, 从而大大减少了更新配置文件的代价. Degemmis 等[52] 代替传统的基于关键词的方法, 利用 WordNet 构建基于语义学(semasiology)的用户配置文件, 配置文件通过机器学习和文本分类算法得到, 里面包含了用户喜好的语义信息, 而不仅仅是一个个关键词. 在基于内容的协同过滤系统上的实验结果表明, 这种方法建立的配置文件可以大大提高推荐的准确性. AdROSA 广告推荐系统[53] 利用

用户注册信息构建配置文件，并且加入用户的IP地址，浏览习惯等信息．该配置文件与Web的内容信息进行匹配分析，相似性最高的Web被推荐给用户．

自动获取或更新用户配置文件的方法需要在配置文件的准确性和易更新性方面找到平衡．准确地捕捉用户喜好信息需要大量的计算资源，更新速度相应也慢很多．反过来，如果更新速度快，就要牺牲其准确性．人机交互的方法是解决这个问题的方法之一．Ricci等[54]设计了一个手机在线旅行推荐系统，通过简单的交互式问题获取用户的喜好信息，进而给用户推荐相应的旅游线路或旅行产品．用户在开始的时候可能对自己的喜好也不是很清楚，因此利用交互式提问的方式是获取用户喜好信息的便捷方法之一．

不同语言构成的配置文件无法兼容也是基于内容的推荐系统面临的又一个大问题．Martínez等[55]提出一个柔性语言表示方法，可以用多种语言的词语表示用户的配置文件．从而可以在多语种环境中进行推荐．

总结起来，基于内容推荐的优点有：

(1) 可以处理新用户和新产品问题(冷启动)．由于新用户没有选择信息，新产品没有被选信息，因此协同过滤推荐系统无法处理这类问题．但是基于内容的推荐系统可以根据用户和产品的配置文件进行相应的推荐．

(2) 实际系统中用户对产品的打分信息非常少，协同过滤系统由于打分稀疏性的问题，受到很大的限制．基于内容的推荐系统可以不受打分稀疏性问题的约束．

(3) 能推荐新出现的产品和非流行的产品，能够发现隐藏的“暗信息”．

(4) 通过列出推荐项目的内容特征，可以解释为什么推荐这些产品．使用户在使用系统的时候具有很好的用户体验．

基于内容的推荐系统不可避免地受到信息获取技术的约束，例如自动提取多媒体数据(图形、视频流、声音流等)的内容特征具有技术上的困难，这方面的相关应用受到了很大限制．下一节我们将介绍基于用户—产品二部图网络结构的推荐算法，该算法不仅可以不受信息挖掘技术的制约，而且可以解决协同过滤推荐系统中打分稀疏性和算法可扩展性等问题．但此类算法仍然难以从根本上解决冷启动问题．

3 基于网络结构的推荐算法

基于网络结构的推荐算法不考虑用户和产品的内容特征，而仅仅把它们看成抽象的节点，所有算法利用的信息都藏在用户和产品的选择关系之中．周涛等[56,57]和Huang等[58,59]分别利用用户—产品用二部分图(bipartite network)建立用户—产品关联关系，并据此提出了基于网络结构的推荐算法．其中，周涛等[56,57]提出了一种全新的基于资源分配的算法，Huang等[58]通过在协同过滤算法中引入二部分图上的扩散动力学，部分解决了数据稀疏性的问题，进一步地，Huang等[59]对两个实际推荐系统的用户—产品二部图进行了分析，发现这两个实证系统具有比随机图更大的平均距离和集聚系数．张翼成等[60,61]考虑用户对产品的打分信息，在更复杂的环境下实现了基于热传导[60]和物质扩散[61]的推荐算法，这些算法效果也明显好于经典的协同过滤．下面简要介绍基于网络结构的推荐算法最近的研究进展．

3.1 基于二部分图资源分配的推荐算法

考虑一个由 m 个用户和 n 个产品(例如书、电影、网页……)构成的推荐系统，其中如果用户 i 选择过产品 j，就在 i 和 j 之间连接一条边 $a_{ji}=1$($i=1,2,\cdots,m$；$j=1,2,\cdots,n$)，否则 $a_{ji}=0$．由此，这个系统可以用一个具有 $m+n$ 个节点的二部分图表示．对于任意目标用户 i，推荐算法的目的是把所有 i 没有选择过的产品按照 i 喜欢的程度进行排序，并且把排名靠前的那些产品推荐给 i．假设 i 选择过的所有产品，都具有某种向 i 推荐其他产品的能力．这个抽象的能力可以看做位于相关产品上的某种可分的资源——拥有资源的产品会把更多的资源交给自己更青睐的产品．对于有 m 个用户和 n 个产品的一般的推荐系统，如果用表示 w_{ij} 产品 j 愿意分配给产品 i 的资源配额，可以得到 w_{ij} 的一般表达式[56]

$$w_{ij}=\frac{1}{k_j}\sum_{l=1}^{m}\frac{a_{il}a_{jl}}{k_l} \tag{10}$$

其中 k_j 表示产品 j 的度(被多少用户选择过)，k_l 表示用户 l 的度(该用户选择过多少产品).

对于给定的一个目标用户，将他选择过的产品上的初始资源设为1，其他设为0. 这样得到一个 n 维的0/1矢量，代表针对该个体的初始资源分配构型. 显然，这个初始构型表达了个性化信息，对于不同用户是不一样的. 记这个矢量为 $\boldsymbol{f}$，通过上述过程得到的最终的资源分配矢量可以表示为

$$\boldsymbol{f}' = \boldsymbol{W}\boldsymbol{f} \tag{11}$$

把目标用户没有看过的所有产品，按照矢量 $\boldsymbol{f}'$ 中对应元素的大小进行排序——值越大就说明该用户越喜欢(这些产品在那些已经被选择过的产品心目中总的分量最重). 排序靠前的产品，可以推荐给目标用户.

为了量化算法的精确性，把真实数据随机划分为两个部分，一部分看做训练集，另外一部分是隐藏起来用于检测算法准确程度的测试集. 构造二部分图和计算 $\boldsymbol{W}$ 矩阵时，只有训练集可以使用. 在没有其他已经条件的前提下，只能假设用户已经选择过的产品是他喜欢的，因此，一个好的算法应该要把训练集中已知的用户喜欢的产品排在比较靠前的位置. 对于任意一个用户 i，假设他有 L_i 个产品是没有选择过的，那么算法会给出这 L_i 个产品一个按照喜好程度的排序(最终资源数量相同的产品被赋予一个随机的序号). 如果在测试集中 i 选择了产品 j(这同时意味着 j 不会出现在训练集中，因此是算法中 L_i 个没有选择的产品之一)，而 j 被算法排在第 R_{ij} 位，那么认为(i, j)的相对位置是

$$r_{ij} = \frac{R_{ij}}{L_i} \tag{12}$$

越精确的算法给出越靠前(r_{ij} 小)的相对位置.

MovieLens 作为标准数据库被用于测试算法的准确程度，该数据库包含了943个用户和1682部电影，由GroupLens研究小组收集(http://www.grouplens.org). 用户对自己看过的电影打1—5分，其中1分表示最不喜欢，5分表示最喜欢. 假设分数大于等于3表示用户喜欢这部电影)，并依此建立二部分图，该图共包含85250条边. 这些边被随机划分为两部分，其中90%归为训练集，10%归为测试集. 运行推荐算法后，测试集中每一组用户—电影对(i, j)都会对应一个相对位置值 r_{ij}. 将所有用户—电影对(i, j)的相对位置求平均，可以得到平均 $\langle r\rangle$，这个值可以量化评价算法的精确度——$\langle r\rangle$ 越小越精确. 通过计算得到，全局排序算法的 $\langle r\rangle = 0.136$，基于Pearson系数的协同过滤算法的 $\langle r\rangle = 0.120$，基于用户—产品二部图网络结构的算法的 $\langle r\rangle = 0.106$. 基于网络结构算法的 $\langle r\rangle$ 在三种算法中最小，说明准确程度最高.

3.2 产品的度信息对推荐准确性的影响

在上一小节介绍的算法中，如果一部电影被1000个用户看过，那么在对这1000个用户推荐时，初始条件中这个电影拥有的资源值都是1. 把这个1看做推荐能力，那么这部电影的总推荐能力就是1000. 也就是说流行的电影推荐的总能力也大！周涛等[57]通过适当地降低流行电影的推荐能力提高了算法的精确性. 对于任意目标用户 i，设定初始资源为[57]

$$f_j = a_{ji} k_j^{\beta} \tag{13}$$

其中 k_j 为第 j 部电影(第 j 个产品)的度，β 是可调参数，当它大于0的时候，大度电影的推荐能力得到提高；反过来，当它小于0的时候，大度电影的推荐能力被压制. $\beta = 0$ 的时候算法退化到上一节讨论过的情况.

MovieLens数据上的数值实验显示，算法精确度在 $\beta = -0.8$ 时得到很大提高，此时 $\langle r\rangle = 0.0972$，较文献[56]中的算法提高了8%. 注意到当 $\beta = -1$ 的时候，每部电影的总推荐能力是一样的，说明算法在推荐能力比较均匀的时候精确度较高.

需要特别强调的是，在同样的用户喜好程度下，推荐冷门的产品要比推荐热门的产品意义更大. 还是以电影为例，向用户推荐好莱坞的大片，如果用户喜欢，固然很好，但是即便没有这个推荐，用户自己通过广播、电视、网络等途径，也能够知道这部大片. 这就好比是互联网中的“明信息”. 但是还有很多信息，对应于那些冷门的产品，可能某个用户很喜欢，但是这些冷门产品数目庞大，又没有媒体宣传，如果没有推荐，用户

根本就无从得知——这些就是互联网中的“暗信息”. 从这个角度讲，挖掘暗信息意义要大很多. 回到算法本身，就是说推荐在同样精确度(用 〈r〉衡量)的情况下，推荐的产品度越小越好. 因为向用户推荐得太多，用户没有精力去看，所以一般而言推荐的产品数目都不会超过100. 举例来说，雅虎音乐(http：//new. music yahoo. com/)的个性化推荐包括40首歌，智能社会书签(smart social bookmarks)系统(http：//www. sesamr. com/)的个性化推荐包括20条书签. 给定了推荐列表的长度 L，系统会自动把排名最靠前的 L 个产品推荐给用户，考察这 L 个产品的平均度. 在三个典型的 L 值下(L=10，50，100)，推荐产品的平均度都随着 β 值单调上升，当 β=－0.8 的时候，不仅仅算法精确度明显比文献[56]好，而且算法能够推荐更加冷门的产品.

3.3 通过去除重复属性获得高精确度的算法

以电影为例，为了简单，我们假设一个用户是否喜欢一部电影只由两个因素决定：主角和导演. 特别地，假设一个目标用户喜欢主角 A 和导演 B. 假设该用户只看过两部电影，这两部电影一部是由 A 主演的(M1)，另一部是由 B 执导的(M2). 如果恰有一部电影 M3 由 B 执导 A 主演，那么这两部电影将分别对电影 M3 产生推荐，推荐的总强度是 2(M1 和 M3 因为 A 关联，M2 和 M3 因为 B 关联，关联的强度都是 1). 考虑另一种情况，该用户看过的两部电影都是由 B 执导的，但是主角都不是 A，那么对于另外一部也是由 B 执导的电影(记这 3 部电影为 M4，M5 和 M6，并假设 3 部电影主角各不相同，且都不是 A)，这两部电影推荐的总强度也是 2. 显然地，这里来自电影 M4 和 M5 的推荐包含了重复的属性(导演是 B)，因此虽然具有一样的强度，用户应该更喜欢电影 M3(既是 B 执导的，又是 A 主演的)而不是 M6(仅仅是 B 执导). 我们希望能够有一种简单的算法来降低这种重复属性的影响. 考虑到 M4 和 M5 自身也具有较强的关联(如果两个电影在对另一个电影的推荐中包含了重复的属性，自然这个属性会导致这两部电影自身的关联)，从 M4 经由 M5 到 M6 和从 M5 经由 M4 到 M6 的二阶关联也应该比较强，从原来的关联矩阵中适当减去二阶关联，有望提高算法的精确程度.

在文献[56]的算法中，最终的资源分配情况可以写成矩阵形式 $\boldsymbol{f}' = W\boldsymbol{f}$，周涛等[62]进一步考虑二阶的耦合，具体定义为

$$W' = W + aW^2 \tag{14}$$

对应的最终资源矢量为 $\boldsymbol{f}' = W'\boldsymbol{f}$，其中 a 是可调参数. MovieLens 数据上的实验结果显示在 a=－0.75 附近，算法表现最佳，对应的 〈r〉值为 0.082. 这里小于 0 的最优的 a 值支持了上面的分析. 特别要强调的是，从全局排序到协同过滤，到二部分图扩散，再到本算法，〈r〉值分别为 0.139，0.120，0.106 和 0.082，提高程度分别为 14%，12%和 23%. 注意到前面两个提高都是算法思想上本质的变化，而本算法与文献[56]的算法在思想上一脉相承，仅仅是从技术上考虑了可能的重复属性，但是提高程度却非常惊人. 高精度的推荐体现了本算法的重要应用价值. 一个自然的问题是，考虑更高阶的耦合是否是有价值的？将本算法进一步推广到三阶的情况

$$W' = W + aW^2 + bW^3 \tag{15}$$

其中 b 也是可调参数. 实验结果表明，综合考虑参数组(a，b)，也只能把 〈r〉再提高 1%—2%. 由于每提高 W 的一个阶次，算法的时间复杂性都会剧增，因此在实际应用中考虑到 2 阶运算就已经足够了.

3.4 通过引入耦合阈值提高算法精确性并降低算法复杂性

在实际应用的时候，降低算法的空间复杂性和时间复杂性是非常关键的问题. 事实上，在一个推荐系统中，一个用户与绝大多数的用户耦合很弱(他们共同选择过的产品非常少，甚至没有)，但是在计算推荐资源的两步扩散时，每一个用户都在考虑的范围中. 通过某种办法去除掉弱耦合的影响可以减少算法整体的计算量.

这里，介绍一种简单的方法[63]. 首先，计算任意两个用户之间的相似性，其中，对 i 而言，i 与 j 的相似性被定义为

$$d_{ij} = \frac{1}{k_i}\sum_{l=1}^{n} a_{li}a_{lj} \tag{16}$$

l 遍历所有的 n 个产品，整个求和符号表示 i 和 j 有多少个共同选择过的产品. 需要注意的是，$d_{ij} \neq d_{ji}$. 当针对目标用户 i 进行推荐的时候，只考虑与 i 的相似性大于或等于给定阈值 d_c 的用户(即所有满足 $d_{ij} > d_c$ 的用户 j)以及和这些用户(包括 i 自己)连接的产品. 这样得到一个子二部分图 $G_i(d_c)$，然后，将扩散算法局限在子图 $G_i(d_c)$中进行. 显然，d_c 越大，$G_i(d_c)$的规模越小，算法的计算量也越小.

数值实验显示，设定阈值 d_c，不仅可以降低算法的时间复杂性，同时还可以提高算法准确程度(考虑弱耦合的影响反而干扰和弱化了最重要的关联信息)，具有一举两得的效果.

3.5 基于传播的用户相似性度量

传统的协同过滤推荐算法中，用户的相似度用 Pearson 系数表征. 受到文献 [56] 的启发，刘建国和汪秉宏[64] 利用资源分配原理计算用户之间的相似性，进而利用协同过滤算法向用户进行个性化推荐，其中，用户的相似性用如下公式度量

$$s_{ij} = \frac{i}{k(u_j)}\sum_{l=1}^{n} \frac{a_{li}a_{lj}}{k(o_l)} \tag{17}$$

其中 $k(o_j)$为用户 o_j 的度. 在 MovieLens 上的数值实验显示，推荐结果的平均准确度可以从基于 Pearson 系数的 0.130 提高到 0.122. 在此算法中假设所有产品的贡献是一样的，即被 1000 个人选择的产品与被一个人选择的产品的影响力是一样的. 为了讨论产品的度信息对推荐结果的影响，文献 [64] 进一步给出包含参数 β 的新的相似性计算公式如下

$$s_{ij} = \frac{i}{k(u_j)}\sum_{l=1}^{n} \frac{a_{li}a_{lj}}{k^{\beta}(o_l)} \tag{18}$$

数值结果显示当 β=1.9 的时候算法的准确度还可以再提高 11.2%. 刘润然等[65] 利用用户和产品的度信息提出了另外的用户相似度计算方法，也得到了比经典的协同过滤算法更好的结果. 这再次验证了产品的度信息确实影响着推荐的准确度，降低大度节点的影响力有利于提高推荐的准确度.

为了节约存储空间，提高计算效率，刘建国和汪秉宏[64] 还提出了基于 top-N 用户相似性信息的协同过滤算法. 算法利用物质扩散原理计算好用户的相似度后，只利用相似度最高的 N 个邻居用户的信息进行相应的推荐. 数值结果显示，这种方法不仅仅可以节约存储空间，而且还存在一个最优的 N 值，在最优值附近的推荐准确度比考虑所有用户影响的情况还要好.

基于网络结构的算法开辟了推荐算法研究的新方向. 然而，该算法也面临着新用户，新产品等问题. 如下节所要介绍的，许多实际的推荐系统把上述几种推荐算法有机结合起来，并取得了不错的应用效果.

4 混合推荐算法

协同过滤，基于内容，以及基于网络结构的推荐算法在投入实际运营的时候都有各自的缺陷[31, 43, 66—70]，因此实际的推荐系统大多把不同的推荐算法进行结合，提出了混合推荐算法. 针对实际数据的研究显示这些混合推荐系统具有比上述独立的推荐系统更好的准确率[43, 70—75]. 目前，最常见的混合推荐系统是基于协同过滤和基于内容的，同时也发展出了其他类型的组合，下面简单进行介绍.

4.1 独立系统相互结合的推荐系统

建立混合推荐系统的方法之一即是独立地应用协同过滤，基于内容和基于网络结构的算法进行推荐. 然后将两种或多种系统的推荐结果结合起来，利用预测打分的线性组合进行推荐[67, 68]. 又或者，只推荐某一时刻在某一个评价指标下表现更好的算法的结果. 例如，Daily Learner 系统[16] 就选择在某一时刻更可信的结果进行推荐. 而文献 [71] 选择一个与用户过去的打分相一致的结果进行推荐.

4.2 在协同过滤系统中加入基于内容的算法

包括 Fab[44] 在内的一些混合推荐系统都是基于内容的协同过滤算法. 即利用用户的配置文件进行传统的协同过滤计算. 用户的相似度通过基于内容的配置文件计算而得，而非共同打过分的产品的信息[68]. 这样可以克服协同过滤系统中的稀疏性问题. 这个方法的另一个好处就是不仅仅当产品被配

置文件相似的用户打了分才能被推荐，如果产品与用户的配置文件很相似也会被直接推荐[43]. Good等[72]用不同过滤器(filterbots)的变化给出了一个相似性计算方法，应用一种特殊的内容分析代理作为协同过滤的一个补充. Melville等[73]利用基于文本分析的方法在协同过滤系统中用户的打分向量上增加一个附加打分. 附加分高的用户的信息优先推荐给其他用户. Yoshii等[74]利用协同过滤算法和音频分析技术进行音乐推荐. Girardi和Marinho[75]把领域本体(domain ontology)技术加入协同过滤系统中进行Web推荐. 另外，把内容分析结合到基于网络的推荐算法中，也是大有可为的. 例如大量的网站都通过标签(tags)和关键词(keywords)，因此研究如何把标签[76]或关键词[77]之间的关联关系与基于网络的推荐算法结合起来是很有意义的.

4.3 其他混合推荐系统研究进展

Basu等[66]以基于内容和协同过滤算法为工具建立〈用户，电影〉二维关联关系，其中用户数组利用协同过滤算法收集了共同喜欢某些电影的用户信息，而电影数组包括了这些电影共有的类型或流派特征. 把用户与电影的关系分类为喜欢和不喜欢. 通过这种分类，预测新用户对不同类型电影的喜好与否. Popescul等[78]和Schein等[69]基于概率浅层语义分析提出了一个结合基于内容和协同过滤算法的统一概率方法. 该方法把用户感兴趣的信息通过浅层语义分析表示成一些主题，利用全概率公式对用户的感兴趣主题进行预测. 实验证明，这种基于概率的方法对稀疏数据非常地有效. Condliff等[79]提出Bayes混合效用回归模型对未知产品进行估计和预测. 模型综合考虑用户的打分信息，用户和产品的配置文件. 建立用户模型之后，利用回归分析，研究用户对某个产品特性的喜好程度，进而把具有这些特性的产品推荐给用户. Christakou等[80]构建了基于神经网络的混合推荐系统. 还有一些混合推荐系统利用基于知识(knowledge-based)的方法进行推荐[77,81,82]，例如基于事例推理的推荐系统. 为了增加推荐准确性，Entrée系统[81]利用事例的领域知识，给饭店的客户推荐菜肴和食品，包括推荐其他饭店. Quickstep和Foxtrot系统[82]利用主体本体信息给用户推荐在线科技论文. Velasquez等[83]提出基于知识的Web推荐系统. 系统首先抽取Web的内容信息，利用用户浏览行为建立用户浏览规则，对用户下一步感兴趣的内容进行推荐. 在根据用户的反馈信息，进行规则的修正. Aciar等[84]利用文本挖掘技术分析用户对产品的评论信息，提出基于知识和协同过滤的混合推荐系统. Felfernig等[85]提出基于知识的自动问答系统CWAdvisor. 系统通过与用户的对话中自动抽取用户感兴趣的内容，把具有相关特性的产品推荐给用户. Mirzadeh等[86]利用交互式咨询管理进行个性化推荐. Wang等[87]构建了基于虚拟研究群体的知识推荐系统，利用基于内容和协同过滤推荐算法向用户推荐显性知识(有用的期刊文件)和隐性知识(可以讨论问题的领域专家). 基于知识的系统的主要缺点就是需要知识获取——这正是许多人工智能应用中最让人头痛的瓶颈. 然而，基于知识的系统已经在一些领域得到了很好的发展. 这些领域的领域知识可以从结构化的机器读取格式中获取，例如XML格式和本体.

5 其他推荐算法

除了上文介绍的几类推荐算法，实际系统中还存在其他推荐算法. 首先是关联规则分析. 关联规则关注用户行为的关联模式，例如，购买香烟的人大都会购买打火机，因此可以在香烟和打火机之间建立关联关系. 通过这种关联关系向用户推荐其他产品. Agrawal等[88,89]提出Apriori算法进行关联规则分析，Han等[90]提出FP-Growth算法大大改进了Apriori算法的运行效率. 另外值得关注的是基于社会网络分析的推荐算法. Wand等[91]利用社会网络分析方法推荐在线拍卖系统中可信赖的拍卖者. Moon等[92]利用用户的购买行为建立用户对产品的偏好相似性，并依此向用户推荐产品并预测产品的出售情况，从而增加用户的黏着性. 最近，任捷等[93]发现，只要预测用户评分的算法可以写成一个矩阵算符，就能够将原始算法改进为一种自适应迭代收敛的形式，从而明显提高算法的精确性.

表1列出了不同领域投入应用的主要的推荐系统.

6 结束语与展望

最近几年，随着互联网的高速发展，我们身处的信息世界的组织和结构有了很大的变化. 首先，

信息量的爆炸性增长使得一个普通用户搜寻自己感兴趣内容的难度和成本都提高了很多；其次，大量的信息被安静地放在网络的旮角，因为访问量小，因此不为人知. 这些“暗信息”中或许有一些是用户感兴趣的，但是没有外界的帮助，普通用户根本无法找到它们. 随着 Web2.0 技术的发展，很多服务型网站可以保留用户的历史记录，这些记录包括选择、评价、购买等等，通过这些记录，采用合适的推荐算法，可以在一定的准确程度上猜测用户喜好，并据此向用户推荐. 这些推荐不仅节省了用户浏览搜索的时间，更为关键的是，没有这些推荐，有些信息用户根本就不可能找到.

我们说对推荐系统的研究，既有重大的社会价值，又有重大的经济意义，就是指它既能作为信息过滤的工具帮助用户更好地利用互联网信息，又能作为网站营销的武器，提高网站的用户黏着度和推广相关产品或服务[94]. 同时，对推荐系统的研究，也有助于解决现代信息科学的中心问题之一：如何从具有极强噪音的稀疏关联矩阵中挖掘有用的信息. 事实上，早在20世纪70年代，就已经出现了相关的研究和应用——协同过滤算法的雏形便已开始形成. 到了90年代，推荐系统的理论框架就已经比较成熟了[1—3]. 遗憾的是，那个时候真正利用推荐算法进行商业化运营的网站并不是太多，可以用于研究的真实数据也比较缺乏，总体上发展比较缓慢，也没有得到信息科学以外的其他学科的关注. 如今，随着推荐系统科学价值和使用价值的凸现，对于推荐算法的研究受到了包括数学、物理、管理科学等多学科的关注.

表1 主要推荐系统表

领域	网站
电子商务	Amazon.com, eBay, Levis, Ski-europe.com, dangdang.com, douban.com
网页标签	Fab, del.icio.us, sesamr.com, Foxtrot, QuIC, Prof-Builder, Quickstep
新闻	GroupLens, PHOAKS
电影	MovieLens, Moviefinder.com
音乐	Pandora, Ringo, CDNOW

本文简单介绍了4类基于不同推荐算法的推荐系统. 虽然这些推荐系统都已经投入应用，并且取得了可观的经济效益，然而，这些系统都面临着许多问题，需要从理论和应用角度进行深入的研究.

(1) 协同过滤推荐系统面临新用户，新产品，打分稀疏性和算法可扩展性等问题. 协同过滤推荐系统必须从用户已经打分的产品中获得用户的偏好信息，进而计算用户之间的相似度. 当一个新用户加入系统的时候，没有选择过任何产品，这样系统就无法给新用户提供准确的推荐服务；新的产品地加入到推荐系统中，由于协同过滤系统完全靠用户已有的选择信息进行推荐，因此，直到新的产品被一部分用户打分，系统才可能推荐它. 已经有很多方法研究冷启动问题，例如 Lee 等[26] 利用伪打分信息，Ahn 等[95] 利用启发式算法度量用户之间的相似性，解决冷启动问题；在任何推荐系统中，已经打分的产品通常比起需要推荐的产品的数量要少很多. 从这些少数的例子中进行准确推荐就尤为重要. Huang 等[58] 利用辅助信息获取和信息扩散方法解决打分稀疏性问题，实验结果表明3种扩散方法在准确性，召回率，F-measure 等方面都比经典的协同过滤推荐算法要好. 解决打分稀疏性问题的另外一个方法就是在计算用户相似度的时候用到用户的配置文件，而不仅仅考虑对产品的打分信息；协同过滤推荐算法能及时利用最新的信息为用户产生相对准确的推荐，但是面对日益增多的用户，数据量的急剧增加，算法的扩展性问题(即适应系统规模不断扩大的问题)成为制约系统实现的重要因素. 虽然与基于模型的算法相比，协同过滤算法节约了为建立模型而花费的训练时间，但是用于识别“最近邻居”算法的计算量随着用户和项的增加而大大增加，对于拥有上百万用户的系统，通常的算法会遇到严重的扩展性瓶颈问题. 该问题直接影响着基于协同过滤的推荐系统的实时性和准确性. 目前，也有这方面的研究工作. 例如，Russell 和 Yoon[96] 利用 DWT 转换方法，提高了基于记忆的协同过滤算法的可扩展性. 基于模型的算法虽然可以在一定程度上解决算法的可扩展性问题，但是该类算法往往比较适于用户的兴趣爱好比较稳定的情况，因为它要考虑用户模型的学习过程以及模型的更新过程，对于最新信息的利用比协同过滤算法要差些.

(2) 基于内容的推荐系统受到信息获取技术，专业化程度过高等因素的制约. 在基于内容的推荐系统中，用户或产品特征集的内容都必须能被计算

机自动抽取出来，因此该类算法受到信息获取技术的严重制约. 例如多媒体数据(图形、视频流、声音流等)在特征自动提取上就存在很大困难. 内容分析的另外一个问题是，如果两个不同的产品恰巧用相同的特征词表示，这两个产品就无法区分. 这就需要加入本体(Ontology)和潜层语义分析(Latent Semantic Analysis)等工具来解决这类问题. 另外，如果一个系统只推荐与用户的配置文件高度相关的产品，那么推荐给用户的产品只能是与他之前购买过的产品非常相似的产品，推荐的多样性很难保证.

(3) 基于网络结构的推荐算法受到新用户、新产品等问题的制约. 利用网络结构的推荐算法的根本是要建立用户—产品二部图关联网络. 新用户或新产品刚进入系统时没有任何选择或被选信息，系统无法与其他用户或产品建立关联网络，因此也无法启动基于网络结构的推荐算法. 如果考虑新用户或产品的配置文件，可以在一定程度上解决新用户或新产品问题. 同时关联网络的建立还受到用户选择关系建立时间的影响. 如果把用户与产品的所有关联关系都考虑在内，就无法区分用户的长期兴趣点和短期兴趣点. 过多地考虑长期兴趣点会使得系统无法给出满足用户短期兴趣的产品，从而使得推荐的准确度大大降低. 如何在基于网络的推荐中加入历史时间的影响，是一个值得关注的问题.

除了上述讨论之外，目前所有的推荐系统都面临着一些具有共性的问题. 这些问题的解决可以从根本上极大地促进推荐系统的研究与应用. 例如，用户和产品的信息是动态改变的(新用户的加入，新产品的加入，用户选择或评价已经存在的产品……)，如果每次改变都需要完全重新计算，这个计算量是巨大的，比较可行的方案是设计某种近似的动态算法，每次都只是局部改变原来的算法结果，而不需要完全的重新的计算. 关于这方面算法的研究，现在基本上还是空白. 另外，现在已经有很多评价指标被提出来对现有的推荐系统结果进行评判. 例如准确率，召回率，F-measure，entropy，平均打分值，产品平均度，差异性等. 不同应用背景的系统在不同的评价指标下表现出来的效果是不同的，不同数据集的结果也不尽一样. 针对不同的推荐系统，如何选择合适的评价指标对推荐效果进行评判是推荐系统研究的重要问题.

参 考 文 献

1 Resnick P, Iakovou N, Sushak M, et al. GroupLens: An open architecture for collaborative filtering of netnews. Proc 1994 Computer Supported Cooperative Work Conf, Chapel Hill, 1994: 175—186

2 Hill W, Stead L, Rosenstein M, et al. Recommending and evaluating choices in a virtual community of use. Proc Conf Human Factors in Computing Systems. Denver, 1995: 194—201

3 梅田望夫. 网络巨变元年——你必须参加的大未来. 先觉: 先觉出版社, 2006

4 Adomavicius G, Tuzhilin A. Expert-driven validation of Rule-Based User Models in personalization applications. Data Mining and Knowledge Discovery, 2001, 5(1—2): 33—58

5 Adomavicius G, Tuzhilin A. Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. IEEE Trans on Knowledge and Data Engineering, 2005, 17(6): 734—749

6 Rich E. User modeling via stereotypes. Cognitive Science, 1979, 3(4): 329—354

7 Goldberg D, Nichols D, Oki BM, et al. Using collaborative filtering to weave an information tapestry. Comm ACM, 1992, 35(12): 61—70

8 Konstan JA, Miller BN, Maltz D, et al. GroupLens: Applying collaborative filtering to usenet news. Comm ACM, 1997, 40(3): 77—87

9 Shardanand U, Maes P. Social information filtering: Algorithms for automating 'Word of Mouth'. Proc Conf Human Factors in Computing Systems Denver, 1995: 210—217

10 Linden G, Smith B, York J. Amazon. com recommendations: Item-to-item collaborative filtering. IEEE Internet Computing, 2003, 7(1): 76—80

11 Goldberg K, Roeder T, Gupta D, et al. Eigentaste: A constant time collaborative filtering algorithm. Information Retrieval J, 2001, 4(2): 133—151

12 Terveen L, Hill W, Amento B, et al. PHOAKS: A system for sharing recommendations. Comm ACM, 1997, 40(3): 59—62

13 Breese JS, Heckerman D, Kadie C. Empirical analysis of predictive algorithms for collaborative filtering. Proc 14th Conf Uncertainty in Artificial Intelligence Madison, 1998: 43—52

14 Delgado J, Ishii N. Memory-based weighted-majority prediction for recommender systems. Proc ACM SIGIR'99 Workshop Recommender Systems: Algorithms and Evaluation, 1999

15 Nakamura A, Abe N. Collaborative filtering using weighted majority prediction algorithms. Proc 15th Int'l Conf Machine Learning Madison, 1998: 395—403

16 Billsus D, Pazzani M. User modeling for adaptive news access. User Modeling and User-Adapted Interaction, 2000, 10(2—3):

147—180

17 Getoor L, Sahami M. Using probabilistic relational models for collaborative filtering. Proc Workshop Web Usage Analysis and User Profiling, San Diego, 1999

18 Hofmann T. Collaborative filtering via gaussian probabilistic latent semantic analysis. Proc 26th Ann Int'l ACM SIGIR Conf, Toronto, 2003: 259—266

19 Marlin B. Modeling user rating profiles for collaborative filtering. Proc 17th Ann Conf Neural Information Processing Systems, 2003

20 Pavlov D, Pennock D. A maximum entropy approach to collaborative filtering in dynamic, sparse, high-dimensional domains. Proc 16th Ann Conf Neural Information Processing Systems, 2002. (http://books nips cc/papers/files/nips15/AP06 pdf)

21 Cohen WW, Schapire RE, Singer Y. Learning to order things. J Artificial Intelligence Research, 1999, 10: 243—270

22 Freund Y, Iyer R, Schapire RE, et al. An efficient boosting algorithm for combining preferences. Journal of Machine Learning Research, 2003, 4: 933—969

23 Jin R, Si L, Zhai C. Preference-based graphic models for collaborative filtering. Proc 19th Conf. Uncertainty in Artificial Intelligence (UAI 2003), Acapulco, 2003: 329—336

24 Jin R, Si L, Zhai C, et al. Collaborative filtering with decoupled models for preferences and ratings. Proc 12th Int'l Conf. Information and Knowledge Management (CIKM 2003), New Orleans, 2003: 309—316

25 Sarwar B, Karypis G, Konstan J, et al. Item-based collaborative filtering recommendation algorithms. Proc 10th Int'l WWW Conf, Hong Kong, 2001: 1—5

26 Lee TQ, Park Y, Park YT. A time-based approach to effective recommender systems using implicit feedback. Expert Systems with Applications, 2008, 34(4): 3055—3062

27 Chen YL, Cheng LC. A novel collaborative filtering approach for recommending ranked items. Expert Systems with Applications, 2008, 34(4): 2396—2405

28 Yang MH, Gu ZM. Personalized recommendation based on partial similarity of interests. Advanced Data Mining and Applications Proceedings, 2006, 4093: 509—516

29 Aggarwal CC, Wolf JL, Wu KL, et al. Horting hatches an egg: A new graph-theoretic approach to collaborative filtering. Proc Fifth ACM SIGKDD Int'l Conf. Knowledge Discovery and Data Mining, San Diego, 1999: 201—212

30 Deshpande M, Karypis G. Item-based top-N recommendation algorithms. ACM Trans Information Systems, 2004, 22(1): 143—177

31 Ungar LH, Foster DP. Clustering methods for collaborative filtering. Proc Recommender Systems, Papers from 1998 Workshop, Technical Report WS-98-08, Menlo Park, 1998: 84—88

32 Chien YH, George EI. A Bayesian model for collaborative filtering. Proc Seventh Int'l Workshop Artificial Intelligence and Statistics, 1999

33 Shani G, Brafman R, Heckerman D, An MDP-based recommender system. The Journal of Machine Learning Research, 2005, 6: 1265—1295

34 Hofmann T. Latent semantic models for collaborative filtering. ACM Trans Information Systems, 2004, 22(1): 89—115

35 Si L, Jin R. Flexible mixture model for collaborative filtering. Proceedings of the Twentieth International Conference on Machine Learning (ICML-2003), Washington DC, 2003

36 Kumar R, Raghavan P, Rajagopalan S, et al. Recommendation systems: A probabilistic analysis. J Computer and System Sciences, 2001, 63(1): 42—61

37 Yu K, Xu X, Tao J, et al. Instance selection techniques for memory-based collaborative filtering. Proc Second SIAM Int'l Conf. Data Mining (SDM'02), 2002

38 Manouselis N, Costopoulou C. Experimental analysis of design choices in multiattribute utility collaborative filtering. International Journal of Pattern Recognition and Artificial Intelligence, 2007, 21(2): 311—331

39 Chen YL, Cheng LC, Chuang CN. A group recommendation system with consideration of interactions among group members. Expert Systems with Applications, 2008, 34(3): 2082—2090

40 Baeza-Yates R, Ribeiro-Neto B. Modern Information Retrieval. Addison-Wesley, Wesley Press, 1999

41 Salton G, Automatic Text Processing. Addison-Wesley, 1989

42 Belkin N, Croft B. Information filtering and information retrieval. Comm ACM, 1992, 35(12): 29—37

43 Balabanovic M, Shoham Y. Fab: Content-based, collaborative recommendation. Comm ACM, 1997, 40(3): 66—72

44 Pazzani M, Billsus D. Learning and revising user profiles: The identification of interesting Web sites. Machine Learning, 1997, 27: 313—331

45 Mooney RJ, Bennett PN, Roy L. Book recommending using text categorization with extracted information. Proc Recommender Systems Papers from 1998 Workshop, Technical Report WS-98-08, 1998

46 Park HS, Yoo JO, Cho SB, A context-aware music recommendation system using fuzzy Bayesian networks with utility theory. Fuzzy Systems and Knowledge Discovery, Proceedings, 2006, 4223: 970—979

47 Somlo G, Howe A. Adaptive lightweight text filtering. Proc Lecture Notes in Computer Science, 2001, 2189: 319—329

48 Zhang Y, Callan J, Minka T. Novelty and redundancy detection in adaptive filtering. Proc 25th Ann Int'l ACM SIGIR Conf Tampere, 2002: 81—88

49 Robertson S, Walker S. Threshold setting in adaptive filtering.

J Documentation, 2000, 56: 312—331
50 Zhang Y, Callan J. Maximum likelihood estimation for filtering thresholds. Proc 24th Ann Int' l ACM SIGIR Conf, New Orleans, 2001: 294—302
51 Chang YI, Shen JH, Chen TI. A data mining-based method for the incremental update of supporting personalized information filtering. Journal of Information Science and Engineering, 2008, 24(1): 129—142
52 Degemmis M, Lops P, Semeraro G. A content-collaborative recommender that exploits WordNet-based user profiles for neighborhood formation. User Modeling and User-Adapted Interaction, 2007, 17(3): 217—255
53 Kazienko P, Adamski M. AdROSA-Adaptive personalization of web advertising. Information Sciences, 2007, 177(11): 2269—2295
54 Ricci F, Nguyen QN. Acquiring and revising preferences in a critique-based mobile recommender system. IEEE Intelligent Systems, 2007, 22(3): 22—29
55 Martínez L, Pérez LG, Barranco M. A multigranular linguistic content-based recommendation model: Research articles. International Journal of Intelligent Systems, 2007, 22(5): 419—434
56 Zhou T, Ren J, Medo M, et al. Bipartite network projection and personal recommendation. Phys Rev E, 2007, 76: 046115
57 Zhou T, Jiang LL, Su RQ, et al. Effect of initial configuration on network-based recommendation. Europhys Lett, 2008, 81: 58004
58 Huang Z, Chen H, Zeng D. Applying associative retrieval techniques to alleviate the sparsity problem in collaborative filtering, IEEE Trans Information Systems, 2004, 22(1): 116—142
59 Huang Z, Zeng D, Chen H. Analyzing consumer-product graphs: Empirical findings and applications in recommender systems. Management Science, 2007, 53(7): 1146—1164
60 Zhang YC, Blattner M, Yu YK. Heat conduction process on community networks as a recommendation model. Phys Rev Lett, 2007, 99: 154301
61 Zhang YC, Medo M, Ren J, et al. Recommendation model based on opinion diffusion. Europhys Lett, 2007, 80: 68003
62 Zhou T, Su RQ, Liu RR, et al. Ultra accurate personal recommendation via eliminating redundant correlations. arXiv: 0805.4127
63 Kuscsik Z, Zhang YC, Zhou T. Improved recommendation algorithm with similarity threshold. submitted to Phys Rev E
64 Liu JG, Wang BH. A spreading activation approach for collaborative filtering. arXiv: 0712.3807
65 Liu RR, Jia CX, Zhou T, et al. Personal recommendation via modified collaborative filtering. arXiv: 0801.1333
66 Basu C, Hirsh H, Cohen W. Recommendation as classification: using social and content-based information in recommendation. Papers from 1998 Workshop, Technical Report WS-98-08, AAAI Press 1998: 714—720
67 Claypool M, Gokhale A, Miranda T, et al. Combining content-based and collaborative filters in an online newspaper. Proc ACM SIGIR' 99 Workshop Recommender Systems: Algorithms and Evaluation, Berkeley 1999
68 Pazzani M. A framework for collaborative, content-based, and demographic filtering. Artificial Intelligence Rev, 1999, 13(5—6): 393—408
69 Schein AI, Popescul A, Ungar LH, et al. Methods and metrics for cold-start recommendations. Proc 25th Ann Int' l ACM SIGIR Conf, Tampere, 2002: 253—260
70 Soboroff I, Nicholas C. Combining content and collaboration in text filtering. Proc Int' l Joint Conf Artificial Intelligence Workshop: Machine Learning for Information Filtering, Aug. 1999
71 Tran T, Cohen R. Hybrid recommender systems for electronic commerce. Proc. Knowledge-Based Electronic Markets, Papers from the AAAI Workshop, Technical Report WS-00-04, AAAI Press, Menlo Park, 2000: 78—83
72 Good N, Schafer JB, Konstan JA, et al. Combining collaborative filtering with personal agents for better recommendations. Proc Conf Am Assoc Artificial Intelligence (AAAI-99), 1999, 439—446
73 Melville P, Mooney RJ, Nagarajan R. Content-boosted collaborative filtering for improved recommendations. Proc 18th Nat' l Conf Artificial Intelligence, Edmonton, 2002, 187—192
74 Yoshii K, Goto M, Komatani K, et al. An efficient hybrid music recommender system using an incrementally trainable probabilistic generative model. IEEE Transactions on Audio Speech and Language Processing, 2008, 16(2): 435—447
75 Girardi R, Marinho LB. A domain model of Web recommender systems based on usage mining and collaborative filtering. Requirements Engineering, 2007, 12(1): 23—40
76 Cattuto C, Loreto V, Pietronero L. Semiotic dynamics and collaborative tagging. PNAS, 2007, 104(5): 1461—1464
77 Zhang Z, Lü L, Liu JG, et al. Empirical analysis on a keyword-based semantic system. arXiv: 0801.4163
78 Popescul A, Ungar LH, Pennock DH, et al. Probabilistic models for unified collaborative and content-based recommendation in sparse-data environments. Proc 17th Conf Uncertainty in Artificial Intelligence, 2001, 437—444
79 Condliff M, Lewis D, Madigan D, et al. Bayesian mixed-effects models for recommender systems. Proc ACM SIGIR' 99 Workshop Recommender Systems: Algorithms and Evaluation, Aug. 1999
80 Christakou C, Vrettos S, Stafylopatis A. A hybrid movie recommender system based on neural networks. International Journal on Artificial Intelligence Tools, 2007, 16(5): 771—792

81 Burke R. Knowledge-based recommender systems. Encyclopedia of Library and Information Systems, Kent A, ed., vol. 69, Supplement 32, Marcel Dekker; 2000

82 Middleton SE, Shadbolt NR, de Roure DC. Ontological user profiling in recommender systems. ACM Trans Information Systems, 2004, 22(1): 54—88

83 Velasquez JD, Palade V. Building a knowledge base for implementing a web-based computerized recommendation system. International Journal on Artificial Intelligence Tools, 2007, 16(5): 793—828

84 Aciar S, Zhang D, Simoff S, et al. Informed recommender: Basing recommendations on consumer product reviews. IEEE Intelligent Systems, 2007, 22(3): 39—47

85 Felfernig A, Friedrich G, Jannach D, et al. An integrated environment for the development of knowledge-based recommender applications. International Journal of Electronic Commerce, 2006, 11(2): 11—34

86 Mirzadeh N, Ricci F. Cooperative query rewriting for decision making support and recommender systems. Applied Artificial Intelligence, 2007, 21(10): 895—932

87 Wang HC, Chang YL. PKR: A personalized knowledge recommendation system for virtual research communities. Journal of Computer Information Systems, 2007, 48(1): 31—41

88 Agrawal R, Imielinski T, Swami A. Mining association rules between sets of items in large databases. Proceedings of the ACM SIGMOD Conference on Management of Data, 1993: 207—216

89 Agrawal R, Srikant R. Fast algorithms for mining association rules. In: Proc 1994 Int Conf Very Large Databases (VLDB' 94), Santiago, 1994, 487—499

90 Han J, Pei J, Yin Y, et al. Mining frequent patterns without candidate generation. Data Mining and Knowledge Discovery, 2004, 8: 53—87

91 Wand JC, Chiu CC. Recommending trusted online auction sellers using social network analysis. Expert Systems with Applications, 2008, 34(3): 1666—1679

92 Moon S, Russell GJ. Predicting product purchase from inferred customer similarity: An autologistic model approach. Management Science, 2008, 54(1): 71—82

93 Ren J, Zhou T, Zhang YC. Information filtering via self-consistent refinement. Europhys Lett, 2008, 80: 68003

94 Schafer JB, Konstan JA, Riedl J. E-commerce recommendation applications. Data Mining and Knowledge Discovery, 2001, 5(1—2): 115—153

95 Ahn HJ. A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem. Information Sciences, 2008, 178(1): 37—51

96 Russell S, Yoon V. Applications of wavelet data reduction in a recommender system. Expert Systems with Application, 2008, 34(4): 2316