大 连 理 工 大 学 本 科 外 文 翻 译

关于采用协同过滤算法的社交推荐系统的研究

A survey of collaborative filtering based social recommender systems

学 部（院）：软件学院

专　　业：软件工程（日语强化）

学 生 姓 名：李浩杰

学　　号：201493033

指 导 教 师：陆坤

完 成 日 期：2018/2/14

大连理工大学

Dalian University of Technology

关于采用协同过滤算法的社交推荐系统的研究

Xiwang Yang[a], Yang Guo[b], Yong Liu[a], Harald Steck[c]

[a]纽约大学理工学院

[b]阿尔卡特-朗讯贝尔实验室

[c]洛斯加托斯 Netflix 公司

摘要：

推荐在我们的日常生活中扮演着日益重要的角色。推荐系统会自动向用户推荐可能感兴趣的物品。最近的研究表明，可以利用来自社交网络的信息来提高推荐的准确性。在本文中，我们提出了基于协作过滤（CF）的社交推荐系统的调查。我们简要介绍推荐系统和不使用社交网络信息的传统方法的任务。然后，我们介绍推荐系统如何采用社交网络信息作为提高准确度的附加输入。我们将基于 CF 的社交推荐系统分为两类：基于矩阵分解的社交推荐方法和基于邻域的社交推荐方法。对于每个类别，我们会调查并比较几种有代表性的算法。

关键词： 社交网络，推荐系统，协同过滤算法

1.简介

通信网络为人们访问信息提供了便利。但同时，网络信息的丰富性也带来了“信息超载”问题。例如，如果有人想购买数码相机，在做出购买决定之前，阅读并比较有关数码相机的所有在线评论将是一件令人抓狂的体验。推荐系统通过自动向用户推荐可能符合其兴趣的商品来处理信息超载问题。准确的推荐使用户能够快速找到中意的物品，而不会被无关的信息淹没。供应商也很热衷于推荐那些符合他们网站访问者兴趣的产品，并希望他们满意以变成回头客。难怪，在 Netflix 竞赛[19]中，仅提高了 10％的推荐准确性就获得了 100 万美元的奖励。

推荐系统（RS）起源于认知科学，近似理论和信息检索等几个相关的学科。由于推荐的重要性日益增加,自 90 年代中期以来它已成为一个独立的研究领域[1]。一般而言，RS 向用户推荐那些可能对他有利的项目。一般而言，推荐方法有两种变体：基于内容的和基于协作过滤（CF）的方法[1,2]。CF 方法可以进一步分为基于模型的 CF 和基于邻域的 CF [2,3]。基于模型的方法使用用户-项目评级来学习得出预测模型。总体思路是用系统中表示用户与项目互动的因素来表示用户和项目的潜在特征，例如用户的偏好类别和项目的类别。相反，基于邻域的 CF 方法使用系统中存储的用户项目评级来直接预测新项目的评级。

在线社交网络（OSN）为进一步提高 RS 的准确性提供了新的契机。在现实生活中，人们通常会在购买产品或使用服务之前向社交网络中的朋友寻求推荐。社会学和心理学领域的发现表明，人类倾向于与类似的其他人构建关系，也称为 homophily [25]。由于稳定和持久的社交联系，人们更愿意与他们的朋友分享他们的个人意见，并且通常信任来自朋友的推荐，而不是来自陌生人和供应商的推荐。流行的在线社交网络，例如 Facebook [21]，Twitter [20]和 YouTube [17]，为人们交流和建立虚拟社区提供了新途径。在线社交网络不仅可以让用户更轻松地分享彼此的意见，而且还可以作为开发新的 RS 算法的平台，以自动化现实生活中的社交网络中的其他手动和轶事社交推荐。

社交 RS 通过将 OSN 中的用户之间的社会兴趣和社会信任作为额外的输入来改进传统 RS 的准确性。例如，由于社交兴趣，用户可以阅读关于事件的特定新闻文章，仅仅是因为事件发生在她的家人住的地方; 由于社交信任，用户可能喜欢 Facebook 上她的好友推荐的歌曲。可以基于用户 u 关于用户 v 的明确反馈（例如通过投票）来建立一对朋友之间的社交信任，或者可以从隐式反馈（例如，交互/通信/交互的频率和数量） u 和 v 之间的电子邮件交换）。不同的社交 RS 算法以不同方式探索社交网络和嵌入式社交信息。在这项调查中，我们关注基于 CF 的社交 RS，因为大多数现有的社交推荐系统都是基于 CF。在对传统的基于 CF 的 RS 进行分类后[2,3]，我们将基于 CF 的社会 RS 划分为两大类：基于矩阵分解（MF）的社会推荐方法和基于邻域的社会推荐方法。在基于 MF 的社交推荐方法中，用户 - 用户社交信任信息与用户 - 项目反馈历史（例如，评级，点击，购买）集成以提高传统基于 MF 的 RS 的精确度，其仅将用户–项目反馈数据作为依据。基于邻域的社会推荐方法包括基于社交网络遍历（SNT）的方法和最近邻（NN）方法。基于 SNT 的算法在社交网络中遍历并查询她的邻居中的直接和间接朋友之后，为用户合成推荐。NN 方法将传统的 CF 邻域与社会邻域相结合，并预测项目的评级或推荐项目列表。

本调查的其余部分安排如下。我们在第 2 节中正式提出了 RS 的任务。第 3 节简要介绍了传统的基于 CF 的 RS。然后我们将在线社交网络作为第 4 节中的附加 RS 输入引入。基于 MF 的社会推荐方法在第 5 节中进行了调查。第 6 节对基于社区的社会推荐方法进行了调查。我们在第 7 节中对基于 CF 的社会推荐方法进行了比较。第 8 节对调查进行了总结。

2. 推荐系统的任务

推荐系统通常向用户提供她可能感兴趣的推荐项目的列表，或者预测她可能更喜欢每个项目的多少。这些系统可帮助用户决定适当的项目，并简化在集合中查找首选项目的任务。

文献的主体一直关注于预测评级值的准确性。为此，测试数据被表示为用户项目评级矩阵R ∈ $\mathbb{R}^{u_0 \times i_0}$，其中$u_0$表示用户数量，$i_0$表示项目数量。$R_{u,i}$是用户 u 对于物品 i 的评级。通常，用户项目评级矩阵 R 中有很多缺失值。在商业系统中 R 的稀疏度通常大于99%[42]。表 1 示出了关于六个用户（表示为u_1至u_6）和七个项目（表示为i_1至i_7）的玩具评级矩阵。每个用户对一些项目进行评级，以表达她对每个项目的兴趣。评级通常是采用数值五星级，其中一颗和两颗星代表负面评级，三颗星代表矛盾心理，而四颗和五颗星代表正面评级。RS 算法预测矩阵中缺失的评级，并且如果她对该项目的预测评级是例如四颗或五颗星则向用户推荐项目。

	i_1	i_2	i_3	i_4	i_5	i_6	i_7
u_1	5		1	5			2
u_2	4	1		5		4	1
u_3	5		1		5	5	1
u_4			5			3	
u_5	2						5
u_6		2				5	

表 1 用户-项目评级矩阵

通常，评级数据集被分成训练集和测试集，其中训练集用于模型拟合和参数调整，并且测试集用于评估 RS。让预测评级的矩阵表示为$\widehat{\mathrm{R}} \in \mathbb{R}^{u_0 \times i_0}$。为了评估 RS 的准确性，最流行的评估指标是均方根误差（RMSE）和平均绝对误差（MAE）：

$$RMSE = \sqrt{\frac{\sum_{(u,i)\in\mathcal{R}_{test}} \left(R_{u,i} - \widehat{R}_{u,i}\right)^2}{|\mathcal{R}_{test}|}}, \quad (1)$$

$$MAE = \frac{\sum_{(u,i)\in\mathcal{R}_{test}} |R_{u,i} - \widehat{R}_{u,i}|}{|\mathcal{R}_{test}|}, \quad (2)$$

其中R_{test}是所有用户项对的集合；在测试集中。RMSE / MAE 越低，预测评级越接近实际评级。

商业 RS 算法通常为用户提供她可能更喜欢的 k 个推荐项目的列表，也被称为 Top-k 推荐，而不是向用户呈现预测的项目评级。除了 RMSE 和 MAE 之外，top-k RS 的直接准确性度量还有 top-k 命中率，精确度和标准化贴现累积增益（NDCG）[43]等。为了计算 top-k 命中率或召回率，对于每个用户 u，我们根据预测评级$\hat{R}_{u,i}$，或投票值对这些项目进行排序。这里我们以预测评级为例。如果预测评级是连续的，则排名列表是唯一的。否则，关系可能会被随机破坏。如果项目被发现有吸引力或有趣（例如，测试数据中指定的评级高于特定阈值），则该项目被定义为与测试集中的用户相关。例如，Netflix [19]数据中的评级值范围为 1 到 5 星，而[9]中的作者认为 5 星评级是相关的（即用户肯定喜欢这些项目），而其他评级值和缺失评级值被视为无关紧要。其他选择导致类似的结果。现在，top-k 命中率或召回率可以定义为测试集中位于排名列表 top-k 中的相关项目的分数，记为$N(k,u)$;从用户 u 的测试集合中的所有相关项目中，用$N(u)$表示。对于每个用户 u，top-k 命中率由下式给出：

$$\mathrm{H}(k,u)=\frac{N(k,u)}{N(u)}. \tag{3}$$

它可以聚合到所有用户，以获得平均 top-k,例如测试集的命中率或召回率为：

$$recall(k)=\frac{\sum_u N(k,u)}{\sum_u N(u)}, \tag{4}$$

这是用户的加权平均值，以及每个用户的体重与用户的相关项目数量$N(u)$成正比。越高 top-k 命中率或召回率表明 top-k 推荐越准确。Top-k 命中率和召回率是 k 的非递减函数。精度是 top-k 推荐的另一个流行指标。对于每个用户，精度由下式给出

$$precision(k,u)=\frac{N(k,u)}{k}, \tag{5}$$

这可以解释为其中相关项目的分数推荐给用户的 k 项。对于给定用户和固定 k，精确度与召回成正比。我们聚合了所有精度用户获得测试集的平均精度如下：

$$precision=\frac{1}{u_0}\sum_u\frac{N(k,u)}{k}=\frac{\sum_u N(k,u)}{u_0 k}, \tag{6}$$

其中 u0 是用户数量。NDCG 是信息检索的另一个准确性度量，其中推荐项目的收益相对于其在整个推荐清单中的位置/排名以对数形式打折[43]。具体来说，假设每个用户你有一个$g_{u,i}$的增益;当项目 i 被推荐时，用户 u 的 k 个项目列表的折扣累积增益（DCG）被定义为：

$$DCG@k(u) = \sum_{j=1}^{k} \frac{g_{u,i(j)}}{max(1, log_b j)}, \tag{7}$$

其中$i(j)$表示有序推荐列表中的第 j 个项目，对数基准 b 是一个自由参数，通常在 2 到 10 之间。以 2 为底的对数通常用于确保所有位置都打折扣。

用户 u 的 NDCG 是 DCG 的归一化版本，由以下公式给出：

$$NDCG@k(u) = \frac{DCG@k(u)}{DCG^{*}@k(u)}, \tag{8}$$

其中$DCG^{*}@k(u)$是理想的$DCG@k(u)$，即项目按降序排列，相对于实部$R_{u,i}$，并且列表在位置 k 被截断。

k 项列表的平均 DCG 定义如下：

$$DCG@k = \frac{1}{u_0} \sum_{u=1}^{u_0} DCG@k(u).$$

类似地，包含 k 个项目的列表的平均 NDC 被定义为：

$$NDCG@k = \frac{1}{u_0} \sum_{u=1}^{u_0} NDCG@k(u).$$

目前，大多数 RS 算法已经被评估并按其预测能力排列，即它们准确预测用户选择的能力。现在人们普遍认为预测的准确性是至关重要的，但真正对于现实世界而言的好 RS 仅具有高准确性并不足够[57,58]。在许多应用中，人们使用 RS 不仅仅是对他们口味的准确预测。用户也可能有兴趣发现新的和多样的项目，偏离他们的日常选择。在提出良好推荐时，RS 保留用户的隐私也很重要。如果要推荐的项目具有高度动态性，例如新闻文章，则 RS 的响应性至关重要。因此，识别可能影响 RS 在特定应用环境中的成功的相关属性集非常重要。

3. 基于反馈数据的推荐系统

社交网络信息仅在最近才可用来改进推荐系统。在概述使用社交网络信息的各种方法之前，我们简要回顾以下两个主要变体：基于内容的方法和协作过滤方法。

基于内容的方法的基本思想是使用项目的属性来预测用户对其的兴趣。例如，对于一本书，可以使用作者的名字，流派，关键字和标签。然后将这些属性与目标用户的口味相匹配。

协作过滤的关键思想是使用每个用户的反馈。关于用户的反馈，可以区分明确的反馈（例如，用户为项目分配评级）和隐式反馈（例如，用户点击链接，收听歌曲或购买项目）。当关于足够多的用户及其反馈的数据可用时，其可用于确定相似的用户（例如，收听同一首歌曲）;然后可以在类似用户中推荐项目：虽然类似的用户在他们的歌曲集合中有很大的重叠，但是每个用户可能已经收听了一些额外的歌曲;那些额外的歌曲可以推荐给其他类似的用户。作为基于其过去行为的相似性来识别类似用户的这种想法的补充，物品之间的相似性可以类推地推断，即，当它们是由相同用户购买时。发现这种基于协同过滤的基本思想可以在文献中提出非常准确的推荐[6,7,44,47]。由于大多数现有的社交推荐系统都是基于 CF 的，所以本节将重点讨论基于 CF 的传统 RS。

文献中的许多工作集中于使用明确的反馈数据，特别是分配给项目（例如，电影，歌曲，餐馆）的评分（例如，从一星到五星级）。其任务是预测用户评价的其他项目的评价值，推荐准确度通常以 RMSE 或 MAE 来衡量。遵循协作过滤方法的基本思想，已经提出了各种最近邻居方法，其可以分为用户 - 用户和项目 - 项邻近模型及其组合，例如参见[6]。发现最准确的方法之一是矩阵分解[24,5,6,13]。这种方法在以前被用于计算机视觉[16]和文本分析[15]。矩阵分解的最基本方法是奇异值分解，但已经开发了许多更复杂的方法，例如[24,5,6,13]。其基本思想是将用户和项目映射到低维空间，并确定这个潜在空间中用户和项目之间的相似性。例如，预测评级矩阵$\widehat{\mathrm{R}} \in \mathbb{R}^{u_0 \times i_0}$可以如下建模

$$\widehat{R} = r_m + QP^{\top}, \tag{9}$$

矩阵$\mathrm{P} \in \mathbb{R}^{i_0 \times j_0}$和$\mathrm{Q} \in \mathbb{R}^{u_0 \times j_0}$，其中$j_0 \ll i_0$; u_0确定（低）排名（例如 50）;$r_m \in \mathbb{R}$ 是（全局）偏移量。$\mathrm{Q} \in \mathbb{R}^{u_0 \times j_0}$本质上代表潜在的用户配置文件，而$\mathrm{P} \in \mathbb{R}^{i_0 \times j_0}$捕获潜在的项目配置文件。使用梯度下降方法，可以确定它们，例如，通过最小化给定评估值$\mathrm{R}_{u,i}$和模型为用户 u 和项目 i 预测的值$\widehat{\mathrm{R}}_{u,i}$之间的平方误差:

$$\sum_u \sum_i W_{u,i} \cdot \left(R_{u,i} - \widehat{R}_{u,i}\right)^2 + \lambda\left(||P||_F^2 + ||Q||_F^2\right), \tag{10}$$

为了防止过度拟合，将最后一项加入以规范学习矩阵 P 和 Q; λ >0 是正则化参数，Frobenius 范数由表示。有多种方式来指定训练权重$W_{u,i}$。一个简单而有效的选择是

$$W_{u,i} = \begin{cases} 1 & \text{if } R_{u,i} \text{ observed}, \\ w_m & \text{otherwise.} \end{cases} \tag{11}$$

当目标是优化观察评级的 RMSE 时，则 w_m= 0.如果目标是为了获得所有项目的良好排序（如通过精度，召回或 NDCG 测量），那么为未观测到的$R_{u,i}$输入一个低值的情况下一个小值w_m> 0 是有利的[9,53]。

矩阵分解方法也可以描述为一个概率图模型[8]，如图 1 所示。通过组合矩阵 Q 和 P 获得评分值$R_{u,i}$。对于矩阵 Q 中的条目的先验分布 P 用σ_Q和σ_P表示；这导致了上面等式中的 L2 正则化项，详见[8]。以σ_R表示的评级值之前的先验产生权重w_m。

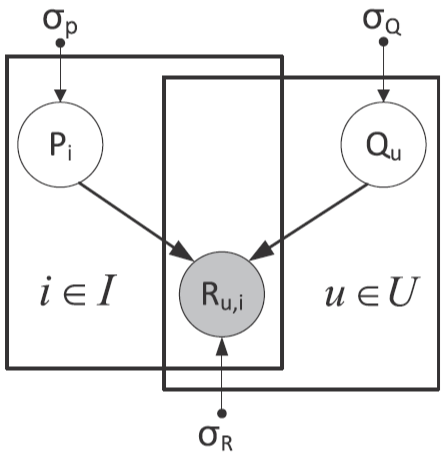

图 1 BaseMF 概率图模型

矩阵分解也与可以与邻域方法结合[6]。有条件限制的玻尔兹曼机[14]是另一个非常成功的模型。通过使用不同模型的集合可以进一步提高推荐的准确性，其预测在最终混合步骤中进行组合。已经开发了各种混合方法，例如，见[4]。

使用隐式反馈在文献中受到的关注较少。该领域著名的出版物包括[10,12,11]，其目的在于根据用户过去的观看行为向用户推荐电视节目，例如他们在每种类型的电视节

目上花费了多少时间。由于在实际应用中，像这样的隐式反馈通常比明确的反馈数据丰富得多，通常部署基于隐式反馈的 RS。

4 社交网络作为额外的 RS 输入

现在我们调查 RS 算法如何采用来自社交网络的信息。我们假设用户在底层社交网络中连接，无论是通用社交网络，如 Facebook [21]，还是特定领域的推荐社交网络，如用于电影推荐和 Epinions 的 Flixster [23]为广泛的产品推荐。我们将基础社交网络表示为有向图 G = (U, F），其中 U 是具有|U| = u_0的用户集合，并且 F 是友谊关系的集合。社交网络信息由矩阵$S \in \mathbb{R}^{u_0 \times u_0}$表示。每个用户都有一个他信任或跟随的直接邻居的$F_u^+$，同时，被一组用户$F_u^-$信任/跟随。用户 u 与用户 v（例如，用户 u 信任/知道/关注用户 v）之间的定向和加权的社交关系由正值$S_{u,v} \in (0,1]$表示; 1.一个缺席或不可观察的社会关系由$S_{u,v} = S_m$表示，其中典型的$S_m = 0$。社会权重$S_{u,v}$可以解释为用户在社交网络中信任或知道用户 v 的程度。它可以基于用户 u 关于用户 v 的明确反馈（例如通过投票）或者从隐式反馈（例如，交互/通信的程度）推断。通常，社会信任$S_{u,v}$是非负的。在特殊情况下，它也可能带有负值，明确地建模两个用户的冲突口味。图 2 示出了六个用户之间的社交网络的玩具示例，其中每个用户都有一组朋友。每个有向的友谊链接都由一个积极的信任值加权。所有用户对之间的社会信任被表 2 中所示的矩阵 S 捕获。在本文中，信任网络和社交网络可以互换地用作通用术语。

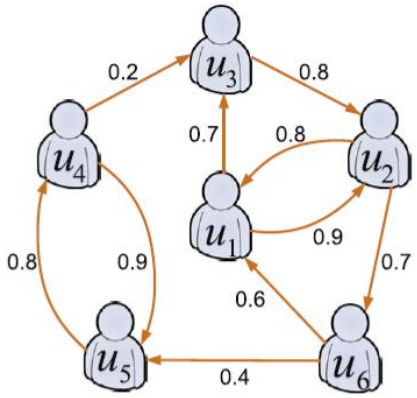

图 2 社交网络示意图

	u_1	u_2	u_3	u_4	u_5	u_6
u_1		0.9	0.7			
u_2	0.8					0.7
u_3		0.8				
u_4			0.2		0.9	
u_5				0.8		
u_6	0.6				0.4	

表 2 用户-用户信任矩阵

参考文献：略