协同过滤推荐算法综述

李晓瑜
(安康学院 电子与信息工程学院,陕西 安康 725000)

摘 要:协同过滤推荐为个性化推荐技术中应用比较广泛的一种推荐,文章主要对协同过滤推荐中广泛应用的技术方法进行了归纳与总结,并对比分析各优缺点,主要包括常用的近邻选择算法和推荐算法以及推荐算法的评估策略进行了较为详细的介绍. 同时对协同过滤推荐技术的发展进行了展望.

关键词:推荐技术;协同过滤;算法
中图分类号:TP301.6 **文献标识码**:A **文章编号**:1672 - 3600(2018)09 - 0007 - 04

Survey of collaborative filtering algorithms

LI Xiaoyu
(Department of Electronic and Information Engineering, Ankang University, Ankang 725000, China)

Abstract: Collaborative filtering is widely used in personalized recommender system. This paper presents an overview of the field of collaborative filtering recommender systems and describes main techniques applied in the current generation of collaborative filtering algorithm. And analyzed that advantages and disadvantage. This paper also has a vision for the development of collaborative filtering technology.

Key words: recommender system; collaborative filtering; algorithms

0 引 言

协同过滤技术自1992年提出以来,发展迅速受到了学术界的广泛关注,特别是在个性化推荐技术中引入协同过滤算法是近几年比较热门的研究趋势. 协同过滤也称为社会过滤,它计算用户间偏好的相似性,在相似用户的基础上自动地为目标用户进行过滤和筛选,其基本思想为具有相同或相似的价值观、思想观、知识水平和兴趣偏好的用户,其对信息的需求也是相似的[1]. 协同过滤主要有两种类型一种是基于用户的,另一种是基于物品的. 基于用户的算法是将和目标用户有共同兴趣爱好的用户所喜欢的物品且目标用户没有购买的物品推荐给目标用户,基于物品的算法是将与目标用户喜欢的物品相似的物品推荐给目标用户. 协同过滤技术可以说是从用户的角度来进行相应推荐的,且推荐的过程是完全自动的,即用户获得的推荐其系统从购买模式或浏览行为等隐式获得的,不需要用户努力地找到适合自己兴趣的推荐信息,如填写一些调查表格等[2]. 使用协同过滤推荐算法进行推荐其主要步骤为建立用户评分表,寻找相似用户,推荐物品. 协同过滤算法研究与基于内容的推荐技术相比具有如下一些优点[3]:

1) 避免了传统基于内容过滤时,产品关键字提取不完全和不精确的问题,通过共享他人的经验,能够推荐一些难以进行内容分析的项目,比如信息质量、个人品味等难以表述的概念,以及视频,音乐和艺术品等商品;

2) 具备发现用户隐藏兴趣的能力. 基于内容的过滤推荐得到的结果很多都是用户本来就熟悉的内容,而协同过滤可以发现用户潜在的但自己尚未发现的兴趣偏好,推荐的结果在内容上可以是完全不相似的信息;

3) 能够有效地使用其他相似用户的反馈信息,较少用户的反馈量,加快个性化学习的速度. 虽然协同过滤作为一种典型的推荐技术有其相当的应用,但协同过滤仍有许多的问题需要解决. 最典型的问题有稀疏问题(Sparsity)和可扩展问题(Scalability).

本文主要对协同过滤推荐算法中的关键技术进行总结并分析了不同技术存在的问题,同时还对协同过滤技术的应用前景进行了展望.

1 近邻选择方法比较

计算用户或项目的相似度是协同过滤推荐算法中重要的一个环节. 在协同过滤推荐技术中计算相似度常用到的方法主

收稿日期:2018 - 01 - 12
基金项目:国家自然科学基金资助项目(50405029);陕西省教育厅2018年科学研究计划(自然科学专项)资助项目
作者简介:李晓瑜(1984—),女,山东菏泽人,安康学院讲师,主要从事网络通信和电子商务研究.

要有以下几种:

1.1 杰卡德相似

1.1.1 杰卡德系数[4]

Jaccard 系数用来度量二值型数据的重叠程度,其定义如下:

$$\mathrm{sim}(i,j) = \left|\frac{R_i \cap R_j}{R_i \cup R_j}\right| \tag{1}$$

其中,分子上是用户 i 和用户 j 的共有项目;分母上计算的是用户 i 和用户 j 的所有项目. 在电子商务中,Jacerad 系数通常可以用来对比不同用户的购物车数据,而这种仅适用于二值型的相似性度量方法也限制了其在推荐系统中的进一步应用[4].

1.1.2 杰卡德系数作为权重的相似性[4-5]

将杰卡德系数作为权重引入相似性计算. 修正后的杰卡德系数考虑到活跃用户和热门项目的评分数量应得到惩罚,修正后的杰卡德系数表示为:

$$J(A,B) = \begin{cases} \dfrac{|A \cap B|}{\sqrt{|A||B|}} & A \neq \varnothing \cup B \neq \varnothing \\ 1, & A = \varnothing \cap B = \varnothing \end{cases} \tag{2}$$

将修正后的 Jaccard 系数,作为原有的相似性度量方法的权重系数,以计算用户相似度为例,设传统的相似性度量方法用 $\mathrm{sim}(i,j)$ 表示,修正后的相似度用 $\mathrm{sim}^+(i,j)$ 表示,则 $\mathrm{sim}^*(i,j)$ 可表示为:

$$\mathrm{sim}^+(i,j) = J(i,j)\,\mathrm{sim}(i,j) \tag{3}$$

这种方式不仅保留了原有方法的易用性,而且克服了传统的方法面对稀疏数据的局限.

1.2 余弦相似性[6]

用户评分被看做是 n 维项目空间上的向量如果用户对项目没有进行评分,则将用户对该项目的评分设为 0,用户间的相似性通过向量间的余弦夹角度量设用户 i 和用户 j 在 n 维项目空间上的评分分别表示为向量 $\boldsymbol{i},\boldsymbol{j}$,则用户 i 和用户 j 之间的相似性 $\mathrm{sim}(i,j)$ 为:

$$\mathrm{sim}(i,j) = \cos(\boldsymbol{i},\boldsymbol{j}) = \frac{\boldsymbol{i}\cdot\boldsymbol{j}}{\|\boldsymbol{i}\|\|\boldsymbol{j}\|} = \frac{\sum_{c=1}^{n} R_{i,c}R_{j,c}}{\sqrt{\sum_{c=1}^{n} R_{i,c}^2}\sqrt{\sum_{c=1}^{n} R_{j,c}^2}} \tag{4}$$

分子为两个用户评分向量的内积,分母为两个用户向量模的乘积. 其中 $R_{i,c}$ 和 $R_{j,c}$ 分别代表用户 i 和用户 j 对项目 c 的评分. 然而在实际中不同的用户打分的尺度不一致,有些用户倾向于打高分,而有些用户倾向于打低分,此时余弦相似性就不能准确地度量用户间的相似性.

1.3 修正的余弦相似性[6]

余弦相似性度量方法中没有考虑不同用户的评分尺度问题,修正的余弦相似性度量方法通过减去用户对项目的平均评分来改善上述缺陷. 其公式表示为:

$$\mathrm{sim}(i,j) = \frac{\sum_{c=1}^{n}(R_{i,c}-\bar{R}_i)(R_{j,c}-\bar{R}_j)}{\sqrt{\sum_{c=1}^{n}(R_{i,c}-\bar{R}_i)^2}\sqrt{\sum_{c=1}^{n}(R_{j,c}-\bar{R}_j)^2}} \tag{5}$$

其中 $R_{i,c}$ 和 $R_{j,c}$ 分别代表用户 i 和用户 j 对项目 c 的评分,$\bar{R}_i$ 和 $\bar{R}_j$ 分别代表用户 i 和用户 j 对所有项目评分的平均值.

在余弦相似性和修正的余弦相似度量方法中,对所有用户没有评分的项目都将评分假设为 0. 但事实上用户对未评分商品的喜好程度不可能完全相同对这些项目的评分也不可能完全相同(全部为 0) . 因此在用户评分数据极端稀疏的情况下,该方法就不能有效地计算用户之间的相似性.

1.4 相关相似性

1.4.1 皮尔逊相关

皮尔逊相关是一种度量两个变量间线性相关程度的方法. 它是一个介于 1 和 -1 之间的值,其中,1 表示变量完全正相关,0 表示无关, -1 表示完全负相关. 在协同过滤算法中,可以利用皮尔逊相关来计算两个用户或者两个项目之间的相关性大小[7];相关系数越高,则两者的相似性越大,反之,则相似性越小.

设经用户 i 和用户 j 共同评分的项目集合用 I_{ij} 表示,则用户 i 和用户 j 之间的相似性 $\mathrm{sim}(i,j)$ 通过 Pearson 相关系数度量两者的相似性可表示为[6]:

$$\mathrm{sim}(i,j) = \frac{\sum_{c\in I_{ij}}(R_{i,c}-\bar{R}_i)(R_{j,c}-\bar{R}_J)}{\sqrt{\sum_{c\in I_{ij}}(R_{i,c}-\bar{R}_i)^2}\sqrt{\sum_{c\in I_{ij}}(R_{j,c}-\bar{R}_j)^2}} \tag{6}$$

其中 $R_{i,c}$,和 $R_{j,c}$ 分别代表用户 i 和用户 j 对项目 c 的评分,R_i 和 R_j 分别代表用户 i 和用户 j 在所有项目的评分平均值.

由于皮尔逊相关系数是通过线性回归公式得到,需要数据之间满足线性关系以及残差相互独立且均值为 0 等假设. 当这些条件不满足时,其计算准确度将会降低.

1.4.2 pearman 秩相关

pearman 秩相关是利用评分的等级来代替评分值,无需满足 Pearson 相关的假设,比较适合于用户评分数据是离散的情况. 其计算公式如下:

$$\mathrm{sim}(i,j) = \frac{\sum_{c\in I_{ij}}(Rank_{i,c}-\overline{Rank_i})(Rank_{j,c}-\overline{Rank_j})}{\sqrt{\sum_{c\in I_{ij}}(Rank_{i,c}-\overline{Rank_i})^2}\sqrt{\sum_{c\in I_{ij}}(Rank_{j,c}-\overline{Rank_j})^2}} \tag{7}$$

由于通常在推荐系统中,项目的评分等级非常有限[8],影响了等级差异的显著性,从而影响了最终的秩相关系数大小.

1.5　基于项目聚类的用户最近邻全局相似性[9]

基于项目聚类的用户最近邻全局相似性,先计算局部最近邻用户相似性. 局部最近邻用户相似性是在 k 个项目聚类的基础上,引入重叠度因子,并将其融合到计算用户局部相似度的公式中. 用户 u 和用户 v 在聚类 C_j 上的局部最近邻用户相似性可表示为:

$$\mathrm{sim}_j(u,v) = \frac{\sum_{i\in I_{uv}^j}(r_{ui}-\bar{r}_u^j)(r_{vi}-\bar{r}_v^j)}{\sqrt{\sum_{i\in I_{uv}^k}(r_{ui}-\bar{r}_u^j)^2}\sqrt{\sum_{i\in I_{uv}^k}(r_{vi}-\bar{r}_v^j)^2}} \tag{8}$$

其中,I_{uv}^j 为用户 u 和用户 v 在聚类 C_j 上共同评分的项目集合. $\bar{r}_u^j$ 和 r_v^j 表示用户 u 和用户 v 对聚类 C_j 中所用项目评分的平均评分. 但是在实际应用中会出现用户共同评分的稀疏性,因此引入重叠度因子来对式(7)进行修正,修正后的公式可表示为:

$$\mathrm{sim}_j^+(u,v) = \frac{\min(|I_u\cap I_v\cap C_j|,\gamma)}{\gamma}\mathrm{sim}_j(u,v) \tag{9}$$

其中,$|I_u\cap I_v\cap C_j|$ 指用户 u 和用户 v 在聚类 C_j 上共同评分的项目数,设置参数 γ,当用户共同评分的项目数小于 γ,即数据相对稀疏时,共同评价的项目数越多,因子值越大,从而保证只有共同评分项目较多且评分相似的用户才有可能成为邻居用户.

全局最近邻用户相似性可以表示为:

$$\mathrm{sim}(u,v) = \sum_{j=1}^{k}\mathrm{sim}_j(u,v) \tag{10}$$

基于项目聚类的用户最近邻全局相似性协同过滤算法,根据用户共同评分的项目数量,引入重叠度因子,并将其融合到计算用户局部相似度的公式中,来进一步加强相似度的准确性.

2　推荐方法比较

2.1　平均加权策略

目前大多数协同过滤推荐系统都采用平均加权策略产生推荐[8],目标用户 u 对未评分项目 i 的预测评分为:

$$P_{ui} = \overline{R_u} + \frac{\sum \mathrm{sim}(u,v)\times(R_{vi}-\overline{R_v})}{\sum|\mathrm{sim}(u,v)|} \tag{11}$$

其中,$\mathrm{sim}(u,v)$ 为用户 u 和用户 v 的相似度,R_{vi} 为最近邻集中的用户 v 对项目 i 的评分,$\overline{R_u}$ 和 R_v 分别为用户 u 和用户 v 对项目的平均评分. 平均加权策略在产生推荐的时候综合考虑了用户对所有项目的评分情况. 这种方法适合于用户评分项目较多时,当用户评分项目较少时该方法就不能很好地反映用户对大多数项目的评分情况.

2.2　Top-N 推荐策略

Top-N 推荐策略是分别统计“最近邻居”集中的用户 i 对不同项的兴趣度的加权平均值,取其中 N 个排在最前面且不属于 Ii(Ii 表示用户 i 评分的项目集合)的项作为 Top-N 推荐集.

3　推荐质量评估方法比较

一个推荐系统的优劣是由其预测结果来衡量的,目前在协同过滤推荐算法中常用到的评估策略主要有以下几种.

3.1　平均绝对误差(MAE)

平均绝对误差是推荐系统中应用最为广泛的评估方法[10],它是通过计算预测值和实际值之间的绝对误差值得到的,计算公式为:

$$\mathrm{MAE} = \frac{\sum(i,j)|p_{ij}-r_{ij}|}{n} \tag{12}$$

其中 n 为评分的总数,p_{ij} 代表用户 i 对项目 j 的预测评分,r_{ij} 代表用户 i 对项目 j 的实际评分,MAE 值越小,推荐精度越高.

3.2　均方根误差(RMSE)[11]

均方根误差(RMSE)也称标准平方差,反映评分数据的离散程度,计算公式为:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{(i,j)}(p_{ij}-r_{ij})^2} \tag{13}$$

其中 n 为评分的总数,p_{ij} 代表用户 i 对项目 j 的预测评分,r_{ij} 代表用户 i 对项目 j 的实际评分,RMSE 值越小,推荐精度越高.

3.3　ROC 曲线[12]

对于一个二分问题,将实例分成正类(positive)或负类(negative),根据预测结果构造以下的二维列联表,其中 1 代表正

类,0 代表负类:

表 1 二分问题列联表

		Actual	Predict
		Positive 1	Negative 0
Positive	1	Truepositive	Falsenegative
Negative	0	Falsepositive	Turenegative

真正类率(true positive rate) TPR = TP/(TP + FN) ,表示用户所喜欢的项目被推荐的概率.

假正类率(false positive rate) FPR = FP/(FP + TN) ,表示用户不喜欢项目被推荐的概率. 还有一个真负类率(True Negative Rate,TNR) ,也称为 specificity,计算公式为 TNR = TN/(FP + TN) = 1 - FPR. 在绘制 ROC 曲线时,将 FPR 和 TPR 分别定义为 X 和 Y 轴,ROC 曲线下面的面积越大,预测的准确率越高.

3.4 召回率(Recall) [8]

召回率用于反映待推荐项目被推荐的比率,计算公式为:

$$\text{tecall} = \frac{|\text{test} \cap \text{top-}N|}{|\text{test}|} \tag{14}$$

其中 test 表示测试数据集中的项目数量,top-N 表示系统推荐给用户的 N 个项目. Recall 值越大被推荐的机率越大.

4 结论与展望

本文主要介绍了采用协同过滤算法进行推荐时,常用的近邻选择算法和推荐算法及推荐算法的评估策略. 协同过滤推荐算法主要存在数据稀疏性、冷启动和鲁棒性问题还有在大数据环境下的推荐效率问题,针对这些问题一些研究者已提出了多种解决方法,最常见的是将其他领域的方法引入进来,协同过滤的跨学科研究也得到了进一步的发展. 随着互联网上信息的急剧增长,协同过滤推荐系统常需要处理海量的数据,如何存储以及如何依据大量的数据计算出推荐结果,是协同过滤推荐面临的一个挑战,可以将协同过滤技术与云计算技术相结合,这样可以使协同过滤推荐系统具有更高的容错能力,实时推荐能力和更强的并行计算能力. 为向用户提供个性化的商品或服务,协同过滤系统需了解用户的个人信息,这就涉及到用户的隐私保护问题. 对协同过滤推荐的隐私保护问题的研究还比较少,还需进一步深化.

参考文献:

[1]黄晓斌. 网络信息过滤原理与应用[M]. 北京: 北京图书馆出版社,2005.
[2]Ma H, King I, Lyu M R. Effective missing data prediction for collaborative filtering[C]. In roceedings of the 30th annual international ACM SIGIR conference on Research and development in information rotricval[A]. New York: ACM,2007: 39 - 46.
[3]Savasere A, Omiecinski E, Navathe S. An efficient algorithm for mining association rules in large databases[C]. Proceedings of the21st International Conference of Very Large Databases[A]. Zurich, Switzerland,1995: 432 - 444.
[4]张晓琳,付英姿,褚培肖. 杰卡德相似系数在推荐系统中的应用[J]. 计算机技术与发展,2015(04) : 158 - 165.
[5]占渊,肖蓉,缪仲凯,等. 基于改进的协同过滤相似性度量算法研究[J]. 计算机测量与控制,2017(09) : 287 - 290.
[6]邓爱林,朱扬勇,施伯乐. 基于项目评分预测的协同过滤推荐算法[J]. 软件学报,2003(09) : 1621 - 1628.
[7]MeLaughlin M R, Herlockor JL. A collaborative filtering algorithm and evaIua: tion metric that accurately model the user experience[C]. in Proceedings of 27th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'04) [A]. Sheffield, UK,2004: 329 - 336.
[8]马宏伟,张光卫,李鹏. 协同过滤推荐算法综述[J]. 小型微型计算机系统,2009,30(07) : 1282 - 1288.
[9]韦素云,业宁,朱健,等. 基于项目聚类的全局最近邻的协同过滤算法[J]. 计算机科学,2012,39(12) : 149 - 152
[10]张光卫,康建初,李鹤松,等. 面向场景的协同过滤推荐算法[J]. 系统仿真学报,2006(S2) : 595 - 601.
[11]DeLeo J. Receiver operating characteristic laboratory ROCLAB: software for developing decision strategies that account for uncertainty [C]. In: Proceedings of the 2nd International Symposium on Uncertainty Modeling and Analysis[A]. IEEE Computer Society Press, College Park, MD,1993,318 - 325.
[12]Chedrawy Z, Abidi SSR. An adaptive personalized recommendation strategy featuring context sensitive content adaptation[C]. In: Proceedings of Adaptive Hypermedia and Adaptive Web-based Systems,2006,4018: 61 - 70.

[责任编辑: 王军]