文章编号: 1003-5850(2010) 04-0012-03

基于领域知识的协同过滤推荐算法

Collaborative Filtering Recommendation Algorithm based on Domain Knowledge

闫祥雨 谢红薇 孙静宇

(太原理工大学计算机与软件学院 太原 030024)

【摘 要】传统协同过滤推荐算法中项目相似度的计算建立在用户评分项目交集之上,没有考虑不同项目之间所存在的语义关系,致使推荐准确率低。基于领域知识进行项目相似度计算的协同过滤算法在用户评分的共同项目很少的情况下仍能给出不错的推荐。实验结果表明,该算法可以有效地解决用户评分数据极端稀疏的问题,提高推荐系统的推荐质量。

【关键词】领域知识, 协同过滤, 稀疏性问题, 项目相似性

中图分类号: TP312 文献标识码: A

ABSTRACT Traditional collaborative filtering recommendation algorithm calculates items similarity using the intersection of different user rating items, does not consider the semantic relationship between different Items, results in a low accuracy rate. A novel collaborative filtering algorithms based on domain knowledge can give good results when user common rating items are sparse. The experimental results show that this method can efficiently improve the extreme sparsity of user rating data, and provide better recommendation results.

KEYWORDS domain knowledge, collaborative filtering, sparse problems, item similarity

随着推荐系统规模的扩大,用户数目和项目数目呈指数级增长,每个用户一般都只对很少的项目评分,这使整个用户-项目评分矩阵非常稀疏,一般都在 1% 之下[1]。由两个用户共同评分的项目则变得更少。基于项目协同过滤算法的提出虽然避免了传统的协同过滤算法计算用户之间相似性的瓶颈,但依然存在一些缺陷。当每个用户都只对很少的项目给出评分时,整个用户评分矩阵非常稀疏,这就导致用户之间的相似性计算不准确,产生的最邻近的邻居用户不可靠,从而难以推荐或预测一个新项目。另外,传统的协同过滤推荐算法中用户相似度的计算建立在用户评分项目交集之上,没有考虑不同项目之间存在的语义关系,从而导致推荐准确率低。本文提出了一种基于领域知识的协同过滤推荐算法(DB-based CF),该方法不仅考虑了项目间所存在的语义,并可在此基础上进行推理。这能很好地解决以上两个问题。

1 基于项目的协同过滤算法

传统的基于项目的协同过滤方法在于找到一组用户已经评分的项目,然后计算它们与目标项 i 的相似性,并从中选出 k 个最相似的项目 $\{i_1, i_2, \cdots, i_k\}$。同时计算它们相应的相似性 $\{s_{i1}, s_{i2}, \cdots, s_{ik}\}$。一旦最相似的项目被发现,通过计算目标用户对这些相似项目的加权平均数来产生预测。所以这里牵涉到两个问题,项目相似性计算和预测产生[2,3]。

① 项目相似性计算。在计算项目相似性时,常用的有三种方法: 余弦相似性、相关相似性、修正的余弦相似性。下面分别介绍这三种方法。

a. 余弦相似性。两个项目被看作是 m 维用户空间上的两个向量。它们间的相似性通过计算两个向量间的余弦夹角来求得。

$$sim(i,j) = \cos(\vec{i},\vec{j}) = \frac{\vec{i} \cdot \vec{j}}{\|\vec{i}\| * \|\vec{j}\|} \quad (1)$$

b. 相关相似性。选取在评分矩阵中对项目 i 和项目 j 都评过分的用户集合 U。则项目 i 和项目 j 之间的相似性 $sim(i,j)$ 通过 Pearson 相关系数度量:

$$sim(i,j) = \frac{\sum_{u \in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U}(R_{u,i} - \bar{R}_i)^2} \sqrt{\sum_{u \in U}(R_{u,j} - \bar{R}_j)^2}} \quad (2)$$

其中,$R_{u,i}$、$R_{u,j}$ 分别表示用户 u 对项目 i、j 的评分。$\bar{R}_i$、$\bar{R}_j$ 分别表示 i 或 j 个项目的平均评分。

c. 修正的余弦相似性。设 $S(i,j)$ 表示项目 i 与项目 j 之间的相似性。对项目 i 和项目 j 共同评过分的用户集合用 U 表示,则项目 i 和项目 j 之间的相似性 $S(i,j)$ 为

* 2009-11-24收到, 2010-02-24改回

** 基金项目: 山西省国际合作项目(2008081032)。

*** 闫祥雨,男,1984年生,硕士,研究方向: 人工智能。

$$sim(i,j)=\frac{\sum_{u\in U}(R_{u,i}-\bar{R}_u)(R_{u,j}-\bar{R}_u)}{\sqrt{\sum_{u\in U}(R_{u,i}-\bar{R}_u)^2}\sqrt{\sum_{u\in U}(R_{u,j}-\bar{R}_u)^2}} \quad (3)$$

其中,$R_{u,i}$表示用户 u 对项目 i 的评分,$\bar{R}_u$ 表示用户 u 对项目的平均评分。

② 预测产生。在计算项目之间的相似性之后,要选择 k 个与目标项目最相似的项目,并产生目标项目的预测值

$$P_{u,i}=\sum_{j=1}^{k}(P_{u,j}\times S(i,j))\Big/\sum_{j=1}^{k}S(i,j) \quad (4)$$

$P_{u,i}$表示用户 u 对项目 i 的预测值。基于项目的协同过滤算法通过计算项目之间的相似性,选择与目标项目的最近邻居集合,避免了计算用户之间相似性的瓶颈,该算法比基于用户协同过滤算法的扩展性强,精确度高。但还是存在数据稀疏性和新项目预测的问题,大量的用户未评分的项目的评分被默认为 0,这使得项目相似性的计算相当不精确。为了解决这两个问题,本文提出基于领域知识的协同过滤推荐算法。

2 基于领域知识的协同过滤算法

为应用此算法,我们首先要建立如图 1所示层次结构的领域本体。领域本体中含有概念、概念间的包含关系以及概念间的其他关系。例如,在图 1中,Movie类的属性“Actor”可以看作 Actor类的一种引用,从关系的角度看,它又可看作 Movie关系的外键。

在此,我们应用领域本体来表示项目的属性信息。项目的属性中所含有的语义信息可以作为推理用户对某个特定项目喜好程度的额外的知识信息[4]。这样,系统就可以在更多的知识信息上来进行推理,从而来提高推荐的精度。另外,当没有或只有很少的信息可以使用时,该系统仍然可以使用语义相似性提供合理的推荐给用户。

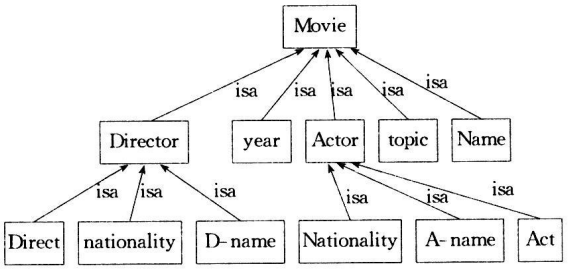

图 1 本体结构图

我们根据所建立的本体及其实例中的语义关系来进行推理,并完成对未评分项目的预测。如图 2所示:当卧虎藏龙被评为 5分时,英雄本色和喜宴也可能因被预测为高分而被推荐,因为他们的演员和导演分

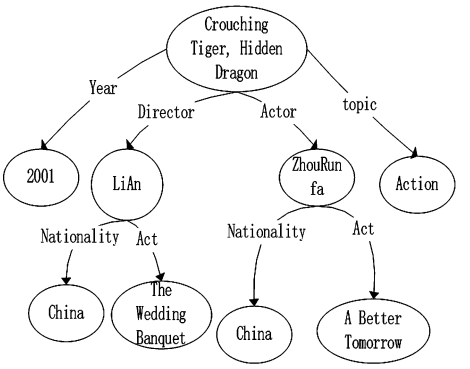

图 2 RDF数据模型

别与卧虎藏龙的一样。本文的算法就是在使用传统的基于项目的推荐算法之前先根据项目属性中的语义关系预测相关未评分项目的评分,来解决数据稀疏的问题,从而取得高质量的推荐。在这里借用文献[4]中所使用的拉普拉斯平滑方法来预测相关项目的评分,然后再用传统的基于项目的推荐算法完成推荐。下面我们来详细介绍此算法的步骤。

2.1 预测相关项目的评分

① 用户给出项目的评分,评分值在 1~5中取得。

② 在项目的语义关系基础上,推荐系统计算用户对其他项目和主题的兴趣值。对于直接相关主题的兴趣值计算,我们采用拉普拉斯平滑方法:

$$\theta_i=(N_i+\lambda)/(N_{presented}+N_{states}\times\lambda) \quad (5)$$

其中 θ_i 表示用户给一个主题评分 i 的可能性,N_i 表示被评分为 i 的主题出现在一个已评分项目集合中的次数,$N_{presented}$是该主题出现在已评分项目中的次数,λ 是平滑参数(常常设置为 1),N_{states}是指评分值。

应用这个公式,我们接下了可以计算主题的兴趣值:

$$Interest_{topic}=\sum_{i=1}^{5}\theta_i\times W_i \quad (6)$$

换句话说,可以根据对已有项目的评分来计算出与此类项目直接相关或语义相关的所有主题的兴趣值,然后把这个兴趣值作为属于此主题但未被评分的项目的评分。

③ 这样我们就可以根据已有评分来计算出与其语义相关的项目的评分。

基于本体中各个项目间的语义关系,我们可以根据项目的已有评分来计算与此项目语义相关的其他项目的分值。这对解决冷开始问题和数据稀疏问题有很大的帮助。

2.2 项目相似性计算

通过第一步,我们可以得到与已评分项目语义相关的未被评分的项目的评分。在这个基础上我们再利用式(3)来求项目相似度。

2.3 使用式(4)来产生预测

先使用拉普拉斯平滑方法预测用户未评分项的评分,然后再计算项目的相似度,最后完成推荐。由于第一步的加入,使得评分矩阵数据变得丰富,解决了数据稀疏性问题,从而能得到更准确的推荐结果。

3 结果及其分析

采用MovieLens站点提供的测试数据集对本算法进行验证。该数据集是美国明尼苏达州立大学计算机科学系Grouplens研究小组搜集的用于研究协同过滤算法的数据集,它包括大约6 040个用户对3 900部电影的近1 000 000个评分(评分值1~5)。每个用户至少评价了20部电影并且包含了用户的简单人口统计学信息和电影的分类信息,利用该数据集提供的电影分类作为领域知识。

采用平均绝对偏差MAE(Means Absolute Error)作为度量标准。MAE通过计算预测的用户评分与实际的用户评分之间的偏差度量预测的准确性[5]。设用户的预测评分和实际评分集合分别为$\{p_1, p_2, \cdots, p_N\}$、$\{q_1, q_2, \cdots, q_N\}$,则平均绝对误差$MAE$定义为

$$MAE=\left(\sum_{i=1}^{N}|p_i-q_i|\right)/N$$

比较本算法与传统的基于用户的、基于项目的协同过滤算法,得到如图3所示的MAE随N变化而变动的折线图。

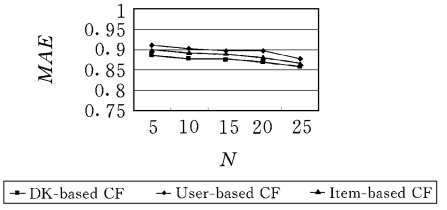

图3 三种推荐算法的精度比较

从图中可以看出,本算法在一定程度上提高了推荐精度。

4 结束语

本文深入分析了在用户-项目评分矩阵极端稀疏情况下,传统的协同过滤推荐算法所存在的问题,并针对该问题提出了基于领域知识的协同过滤推荐算法。此算法充分利用项目之间所存在的语义关系来计算项目相似度,然后再结合传统的基于项目的协同过滤算法求出目标项目的预测值。实验结果表明,基于领域知识的协同过滤推荐算法可以有效地解决用户-项目评分矩阵的极端稀疏问题,从而提高推荐质量。下一步工作我们将考虑再结合其他的推荐算法来提高推荐质量。

参 考 文 献

[1] 张光卫,李德毅,李 鹏等.基于云模型的协同过滤推荐算法[J].软件学报,2007,18(10):2 403-2 411.

[2] 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,2003,14(9):1 621-1 628.

[3] 罗耀明,聂规划.语义相似性与协同过滤集成推荐算法研究[J].武汉理工大学学报,2007,29(1):85-88.

[4] YiwenWanga, Natalia Stasha, Lora Aroyoa et al. Recommendations based on Semantically Enriched Museum Collections[J]. Web Semantics: Science, Services and Agents on the World Wide Web, 2008 283-290.

[5] 肖 敏,熊前兴.基于项目语义相似度的协同过滤推荐算法[J].武汉理工大学学报,2009,31(3):21-23,32.

(上接第2页)

滤测试,采用基于新闻标题的文本过滤方法进行过滤,实验结果查全率为81.7%。

随着因特网的不断普及和发展,各类新闻文本日益增多,文本过滤是处理海量文本的良好手段。文章给出的基于突发事件新闻标题的文本过滤模型,其特点是:该方法可操作性强,实现简单,过滤速度快,有一定的实际作用。下一步工作将选取更多类型的突发事件新闻语料进行大规模的试验并将查准率和查全率相结合,全面地评价此模型;另外,该模型只是完成第一步的粗略过滤,系统将在此模型的基础上结合统计方法对未过滤出的文本进行过滤,以便获得更好的效率,更进一步地对过滤模型进行完善。

参 考 文 献

[1] 黄萱菁,夏迎炬,吴立德.基于向量空间模型的文本过滤系统统[J].软件学报,2003,14(3):435-442.

[2] Robertson S, Hull DA. The TREC-9 Filtering Track Final Report. In Voorhees EM, Harman DK, eds. Proceedings of the 9th Text Retrieval Conference (TREC-9). Gaithersburg: NIST Special Publication, 2001: 25-40.

[3] 赵丰年,刘 林,商建云.基于概念的文本过滤模型[J].计算机工程与应用,2006,4:186-188.

[4] 林鸿飞.基于示例的文本标题分类机制[J].计算机研究与发展,2001,9:1 132-1 136.

[5] 卢伟伟.Web文本信息过滤方法研究[D].武汉:华中科技大学,2007.

[6] 战学刚,姚天顺.基于汉语分析的中文分类方法[A].中文信息处理国际会议论文集.北京:清华大学出版社,1998.412-417.