一种改进的 Slope One 协同过滤算法

王 毅[1] 楼恒越[2]
(西南交通大学信息科学与技术学院 成都 610036)[1] (四川师范大学计算机科学学院 成都 610101)[2]

摘 要 相对传统的基于用户项目评分的协同过滤算法,Slope One 算法简单、高效。但该算法依赖于大量用户对待预测项目的评分,如果对预测项目评分的用户较少,没有考虑用户本身的喜好,将对评分预测的结果有影响。因此,引入描述关键字的语义相似度,利用关键字相似性度量项目间的相似程度,并结合该用户对其他项目的评分,提出一种基于项目语义相似度的改进 Slope One 算法,并在标准的 MovieLens 数据集上进行预测实验。实验数据表明,相对于原算法,改进的算法在一定程度上提高了预测的准确性。

关键词 协同过滤,Slope One 算法,用户推荐,语义相似

Improved Slope One Algorithm for Collaborative Filtering

WANG Yi[1] LOU Heng-yue[2]
(School of Information Science&Technology, Southwest Jiaotong University, Chengdu 610036, China)[1]
(Institute of Computer Science, Sichuan Normal University, Chengdu 610101, China)[2]

Abstract Compared to traditional rating-based collaborative filtering algorithm, the Slope One algorithm is simple and efficient. However, the Slope One algorithm relies on a large number of users' ratings to the item which should be predicted. The rating prediction is affected when users' ratings are not enough and it does not consider the users' habits. For the reason, the semantic similarity of the keywords to describe the items was introduced, which measures the degree of similarity between item-pairs, then a combination of items' semantic similarity and the Slope One algorithm was proposed. Finally, by the standard data set MovieLens, the data of the experiment's result show that the improved algorithm improvers the accuracy of the original algorithm.

Keywords Collaborative filtering, Slope One algorithm, User recommendation, Semantic similarity

1 引言

用户推荐系统主要通过分析大量用户的兴趣相似程度和用户本身的习惯,搜索用户可能感兴趣的项目(例如电影、文献、音乐、Web 页面等)推荐给用户[1-3]。用于构建用户推荐系统的关键技术之一是协同过滤算法,在文献[4]中提出了一种基于用户情景相似的协同过滤算法,将基于项目的协同过滤算法与用户聚类相结合,提高了原算法的预测准确度。而本文改进的 Slope One 算法是一种经典的基于用户-项目评分矩阵的协同过滤算法,该算法是一个增量算法,对评分较少的用户也可以产生推荐,同时准确度比传统的基于用户和项目的协同过滤算法要好[5]。现有的 Slope One 算法主要考虑同时对目标用户已评分项目和候选项目均有评分的用户,通过大量此类用户的评分估计目标用户对候选项目的评分。如果对候选项目评过分的用户很少或没有,则该算法可能导致生成的评分估计值误差较大,甚至无法估计评分。此外该算法仅考虑了不同用户间评分的相似性,而忽略了用户本身的喜好。

本文针对上述问题,将描述项目的关键字考虑进来,与 Slope One 算法相结合,提出一种改进的 Slope One 算法。主要思想是:首先根据同一用户已评分的项目与候选项目名称中的关键字,计算语义相似度,并以此作为项目间的相似度;然后根据该用户已评分项目估计一个结果;最后和 Slope One 算法的结果线性组合成预测的评分。实验结果表明改进的算法在推荐性能上有一定的提高。

2 Slope One 算法

Slope One 算法的基本思想是:设用户 u 对某两个项目的评分为 x,y,该算法假设所有用户对这两个项目的评分 x 与 y 之间符合线性关系 $y=x+b$。通过已经对这两个项目评过分的大量用户评分数据拟合该线性函数获得参数 b 的估计值 $\hat{b}$,最后将目标用户已有的项目评分 x 代入拟合公式 $y=x+\hat{b}$,从而获得 u 对项目的评分估计值 $\hat{y}$[5]。

2.1 符号表示

在一个典型的协同过滤算法中,输入的数据通常表示为一个 $m\times n$ 的用户-项目评分矩阵,如表 1 所列。

表 1 用户-项目评分矩阵

user/item	$item_1$	$item_2$	$item_3$	…	$item_n$
u_1	r_{11}	null	r_{13}	…	r_{1n}
u_2	r_{21}	r_{22}	null	…	r_{2n}
…	…	…	…	…	…
u_m	null	r_{m2}	null	…	r_{mn}

王 毅(1981—),男,硕士生,讲师,主要研究方向为数据挖掘,E-mail:yiwang@yeah.net;楼恒越(1990—),男,本科生,主要研究方向为网络信息系统。

其中 r_{ij} 表示用户 u_i 对项目 $item_j$ 的评分，其大小反映该用户对项目的喜好程度，一般为 1 到 5 之间的实数，u_i、$item_j$ 分别为用户和项目的 ID 号，不妨设 $u_i=i$，$item_j=j$。由于 n 常常很大，用户只可能对部分项目打分，因此该矩阵有大量值是 null，一般为稀疏矩阵。另外，记 $\hat{r}_{ij}$ 表示 u_i 对 $item_j$ 的评分预测值，S_i 表示 u_i 评过分的项目集合，U_j 表示所有对 $item_j$ 评过分的用户集合，$\| s \|$ 表示集合 S 中元素的个数。

2.2 Slope One 算法模型

通常，我们要拟合线性方程 $y=x+b$，需要一系列 (x_i, y_i)，$(i=1,2,\cdots,n)$ 值对的集合。然后根据最小二乘法进行线性拟合，可求得令目标函数 $\sum_i (x_i+b-y_i)^2$ 达到最小值的参数 b 的估计值 $\hat{b}=\sum_i \frac{(y_i-x_i)}{n}$。给定如表 1 所列的用户-项目评分矩阵，Slope One 算法就是利用该拟合公式对评分矩阵中 null 值进行估计填充。

假设现在要估算 r_{ij} 的值，Slope One 算法先利用式(1)计算 $item_i$ 与 $item_j$ 之间的相异性，然后再利用式(2)完成对 r_{ij} 的预测。

$$d_{ji}=\sum_{u\in U_i\cap U_j}\frac{r_{uj}-r_{ui}}{\| U_i\cap U_j \|} \tag{1}$$

$$\hat{r}_{ij}=\frac{1}{\| S_i \|}\sum_{k\in S_i}(r_{ik}+d_{jk}) \tag{2}$$

Slope One 算法有很多优点，但预测精度还有待提高，特别是当对候选项目评过分的用户数较少时，对预测的准确性影响较大。本文利用描述项目的关键字之间的语义相似度来衡量项目之间的相关程度，并将其和用户评分相结合，在一定程度上提高了推荐质量。

3 基于项目语义相似度的 Slope One 算法

在信息检索中，很多项目本身可以用一些关键字加以描述，例如电影标题中的关键词汇、文献中的关键字等。这些关键字构成了对项目的一种特征描述，因此可以利用其获得两个项目间的相关程度大小。

3.1 WordNet 语义相似度计算

WordNet 是一个覆盖范围宽广的英语词汇语义网络。名词、动词、形容词和副词各自组织成一个同义词集合的网络，每个同义词集合代表一个基本的语义概念，并且同义词集合之间也用“is-a”、“has-a”等多种关系连接[6]。在文献[7]中 Dekang Lin 利用信息论的知识提出一种基于 WordNet 的概念语义相似度的计算方法。

记 c_1 和 c_2 是两个概念，$lso(c_1,c_2)$ 表示在 WordNet 中 c_1 与 c_2 的最近公共上位词(lowest super-ordinate or most specific common subsumer)[8]。Lin 用两个概念的互信息量同它们信息量总和的比值度量概念语义相似度，计算方法如公式(3)所示。

$$\mathrm{sim}_{\mathrm{Lin}}(c_1,c_2)=\frac{2\times IC(lso(c_1,c_2))}{IC(c_1)+IC(c_2)} \tag{3}$$

式中，$IC(c)=-\log P(c)$ 表示概念 c 的信息量(information content)，而 $P(c)$ 表示概念 c 在指定的语料库中出现的概率，可以用 Resnik 在文献[9]中提出的式(4)计算。概念 c 一般包含多种意义，本文采用 Brown 语料库统计表示这些意义的所有词汇的集合 $W(c)$。N 是同时包含在 Brown 语料库中与 WordNet 中的词汇总数量。

$$P(c)=\frac{\sum_{\omega\in W(c)} count(\omega)}{N} \tag{4}$$

3.2 基于项目语义相似度的评分预测

根据 Lin 的语义相似度计算方法，本文利用数据挖掘中两个簇之间的“组平均”值[10]计算方法，提出一种将用户的评分与项目语义相似度相结合的方法，得到基于项目语义相似度的 null 值评分预测结果。具体步骤如下。

1)预处理。本文实验的项目为英文电影，描述电影的关键字从电影标题中获取。首先需要按非英文字符对标题中的单词进行分割，然后通过选定的停用词表进行过滤获得关键词表。而名词的复数形式，如 flowers 以及动词的各种时态，如 hating，WordNet 将这些词同其原形放在同义词集合中，即 flowers 与 flower，hating 与 hate 含义相同。而一个同义词集合中的单词在 Lin 的算法中视为同一个概念，因此基于 WordNet 编程时词形的因素对结果不会造成影响，无需专门处理。

2)计算项目间的语义相似度矩阵。设描述项目 i 和 j 的关键字集合分别为 $\{a_1,a_2,\cdots,a_m\}$ 以及 $\{b_1,b_2,\cdots,b_n\}$，则项目 i 与 j 的语义相似度定义为

$$\mu_{ij}=\sum_{i=1}^{m}\sum_{j=1}^{n}\frac{\mathrm{sim}_{\mathrm{Lin}}(a_i,b_j)}{m\times n} \tag{5}$$

式中，设 $\mathrm{sim}_{\mathrm{Lin}}(a_i,b_j)>\xi$，$\xi\in(0,1)$。$\xi$ 为一阈值，将相似度较低的关键字组合去掉，以提高相似度计算的精度，本文实验中取为 0.5。根据式(5)可以产生项目间的相似度矩阵，如式(6)所示。

$$[\mu_{ij}]_n=\begin{pmatrix}1 & 0.5 & 0.75 & \cdots\\ 0.5 & 1 & 0.63 & \cdots\\ 0.75 & 0.63 & 1 & \cdots\\ \cdots & \cdots & \cdots & \cdots\end{pmatrix} \tag{6}$$

3)计算用户对预测项目的评分。设 j 为用户 u_i 的未评分项目，则计算基于项目相似度的预测评分如式(7)所示。

$$\hat{r}_{ij}=\frac{\sum_{k\in S_i} r_{ik}\times\mu_{kj}}{\sum_{k\in S_i}\mu_{kj}} \tag{7}$$

式(7)基于的假设是：同一个用户对相似的项目(例如同样题材或者一个系列的电影等)有同样的偏好，这样就以其他项目与预测项目间的语义相似度为权值，计算该用户已有评分的加权平均数作为预测项目的评分。此处只分析该用户自己的评分习惯，不考虑其他用户的影响。

3.3 改进的 Slope One 算法

将 Slope One 模型与项目语义相似度模型相融合，用户 i 对项目 j 的最终预测评分由两种模型的计算结果共同决定，如式(8)所示。

$$\hat{r}_{ij}=\begin{cases}\lambda r_{ij}^{(1)}+(1-\lambda)r_{ij}^{(2)}, & \min(r_{ij}^{(1)},r_{ij}^{(2)})>0\\ \max(r_{ij}^{(1)},r_{ij}^{(2)}), & \text{其他}\end{cases} \tag{8}$$

式中，$r_{ij}^{(k)}$，$k=1,2$，分别表示用 Slope One、项目语义相似度这两种模型计算出的用户 i 对项目 j 的预测评分。λ 表示两种方法在最终结果中所占的权重，可以根据对 $item_j$ 评过分的人数占全部用户人数的百分比进行调节，也可以直接指定一个固定值，本文根据实验结果的表现将其设定为 $\lambda=0.7$。

3.4 算法流程

输入：用户-项目评分矩阵(分为训练集和测试集)

输出：测试集中项目的预测评分

1)将训练集中的用户、项目以及评分数据和测试集中的

用户和项目数据结合，利用 Slope One 算法计算出测试集中项目的预测分数；

2)按照 3.2 节中的步骤 1)—步骤 2)计算出训练集和测试集中全部项目的语义相似度矩阵$[\mu_{ij}]_n$；

3)根据式(7)计算出测试集中项目基于项目语义相似的预测分数；

4)由式(8)计算出测试集中项目的最终预测分数。

4 实验结果及分析

4.1 实验数据集及检验标准

本文针对所设计的算法在 Hadoop 平台上进行了测试。实验数据采用了 MovieLens 提供的标准数据集，它包含了 943 个用户对 1682 部电影的 100000 次评分，其中在文件 movies.dat 中记录了这 1682 部电影的名称及类型[11]。为了对比本文所提算法的效果，首先将这 100000 次评分按 4:1 的比例随机抽取形成 3 组训练集和测试集，另外还有 2 组数据训练集仍然随机进行抽取，但对测试集中的候选项目限制对其评过分的用户数占总用户数的百分比不超过 10%。

采用的评估预测精度验证方法是广泛使用的衡量用户推荐算法精度的方法——平均绝对误差(Mean Absolute Error, MAE)：

$$MAE(f)=\frac{1}{R_{test}}\sum_{r_{ui}\in R_{test}}|f(u,i)-r_{ui}| \tag{9}$$

式中，$f(u,i)$是用户 u 对项目 i 的预测评分，r_{ui}是测试集 R_{test} 中项目的实际评分。通过计算项目的实际评分与预测评分之间的平均误差来度量算法推荐的精度，MAE 值越小，算法性能就越好。

4.2 实验结果的分析

本文将修改后的 Slope One 算法与原算法进行对比，实验结果如图 1 所示。

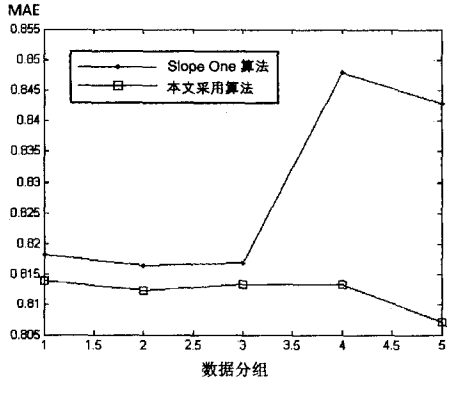

图 1 分组实验结果

图 1 中横坐标表示用于实验的数据分组，纵坐标表示每组数据分别用两种算法求的 MAE 值。

从图上来看，针对前 3 组随机抽取的数据，本文改进的算法 MAE 值比 Slope One 算法有一定程度的减小。相对于文献[5]中提出的一种改进的加权 Slope One 算法，本文提出的算法 MAE 值减小的幅度更大。而特意选取评分较少的项目构成测试集的第 4、5 组数据，两种算法的 MAE 值相差较大。因此，实验结果表明，Slope One 算法的精度受对候选项目评过分的用户数量影响较大，而本文采用的算法在不降低原来算法性能的前提下，减小了这种影响，从而在一定程度上提高了预测的精度。

结束语 本文分析了 Slope One 算法在对预测项目评分用户数目较少的情况下可能面临的预测精度降低的问题。因此，本文利用信息检索中常用的计算概念语义相似度的方法来度量项目之间的相似程度，对同一用户已有的评分数据做出预测，充分考虑了用户自身的喜好。实验结果也表明，将基于项目语义相似度的模型与 Slope One 算法结合后可以改善原算法预测评分的质量。

参 考 文 献

[1] Mahmood T, Ricci F. Improving recommender systems with adaptive conversational strategies[C]// HT '09 20th ACM Conference on Hypertext and Hypermedia. New York, USA: ACM Press, 2009: 73-82

[2] 刘浩淼，徐从富，何俊. 基于模糊聚类的歌曲智能推荐方法研究[J]. 计算机工程与设计，2009，30(10)：2423-2427

[3] 刘志昆，王卫平. 基于精确序列格式的网页个性化推荐[J]. 计算机系统应用，2006，(5)：32-35

[4] 周涛，李华. 基于用户情景的协调过滤推荐[J]. 计算机应用，2010，30(4)：1076-1082

[5] Lemire D, Maclachlan A. Slope One Predictors for Online Rating Based Collaborative Filtering[C]// SIAM Data Mining (SDM'05). 2005: 21-23

[6] Miller G A. WordNet A Lexical Database for English[C]// Communications of the ACM. New York, USA: ACM Press, 1995: 39-41

[7] Lin De-kang. An Information-Theoretic Definition of Similarity [C]// Proceedings of the 15th International Conf. on Machine Learning Morgan Kaufmann. San Francisco, CA, 1998: 296-304

[8] Budanitsky A, Hirst G. Evaluating WordNet-based Measures of Lexical Semantic Relatedness [J]. Association for Computational Linguistics, 2005, 32: 16-23

[9] Resnik. Using information content to evaluate semantic similarity[C]// Proc. 14th International Joint Conference on Artificial Intelligence. 1995: 448-453

[10] Soman K P, Diwakar S, Ajar V. 数据挖掘基础教程[M]. 范明，牛常勇，译. 北京：机械工业出版社，2009

[11] MovieLens Data Sets. 2011-5-8[EB/OL]. http://www.grouplens.org

(上接第 177 页)

[5] Lemire D, Maclachlan A. Slope one predictors for online rating-based collaborative filtering [C]// Proceedings of the S I2AM Data Mining Conference. Newport Beach, CA, USA: Society for Industrial Mathematics, 2005: 21-25

[6] Adomavicius G, Sankaranarayanan R, Sen S, et al. Incorporating contextual information in recommender systems using a multi dimensional approach [J]. ACM Transactions on Information Systems (TOIS), 2005: 23(1): 103-145

[7] 戴亚娥，龚松杰. 个性化服务中基于模糊聚类的协同过滤推荐[J]. 计算机工程与科学，2009，31(4)：1007