****

**数据仓库与数据挖掘推荐系统**

**--协同过滤实现**

**院（系）：**  软件学院

**专 业：**  10软件四班

**学生姓名 ：** 庄灿杰

蔡锐涛（组长）

戴颖毅

陶升奇

**指导教师** ： 蔡毅

**提交日期：** 2012/11/21

1. **问题分析**

数据来源：MovieLens

数据特性：

1.Ratings are made on a 5-star scale (whole-star ratings only).

2.Each user has at least 20 ratings.

3.100,000 ratings (1-5) from 943 users on 1682 movies.

4.Training set/test set = 80,000/20,000

5.Trainnig ratio (0.1-0.9).

6.Each record contain user Id、 item Id、a rating that user comment on the item、 and a time stamp.

问题描述：

根据所提供的MovieLens 的数据集合，使用数据仓库与数据挖掘课程上所学习的知识，设计一个推荐系统。

质量指标：

1. MAE (Mean Average Error)



1. RMSE (Root Mean Square Error)



1. TIME (Time costs)

【Where *m* is the number of user-movie pairs in the test set



is the rating obtained by calculation of using our method

is the rating given by user in the test set】



**二，分析设计**

推荐系统根据系统中已有的用户信息（Training Set），利用信息过滤技术来预测特定的用户对特定的商品的喜好，或者向特定的用户推荐最感兴趣的商品。常用的推荐方法有基于内容的推荐(CB)、协同过滤推荐(CF)以及组合推荐(Hybrid)等，其中协同过滤推荐是广为应用的个性化推荐技术之一，协同过滤推荐算法基本思想是通过计算目标用户与各个基本用户对项目评分之间的相似性(User-based) ,搜索目标用户的最近邻居(KNN),然后由最近邻居的评分数据向目标用户产生推荐 ,即目标用户对未评分项目的评分可以通过最近邻居对该项目评分的加权平均值进行逼近 ,从而产生推荐。

本次实验中采用的是协同过滤推荐，主要原因：源数据的特殊性，仅有(uID、iID、rating、timeStamp)，数据记录相对简单，数据中没提供资源标签，所以使用基于内容(CB)的推荐,并不合适。

实验中我们通过修改了协同过滤推荐算法中基于用户里的KNN算法的相关设置，增设阈值，减少了一定的计算量，并做了相应的测试，和详细的记录。

**三，算法流程**

1. 算法流程如下图所示
2. 在KNN算法中，目标用户u和邻居用户v之间的相似性是通过Peason相关系数来度量的。其计算公式为

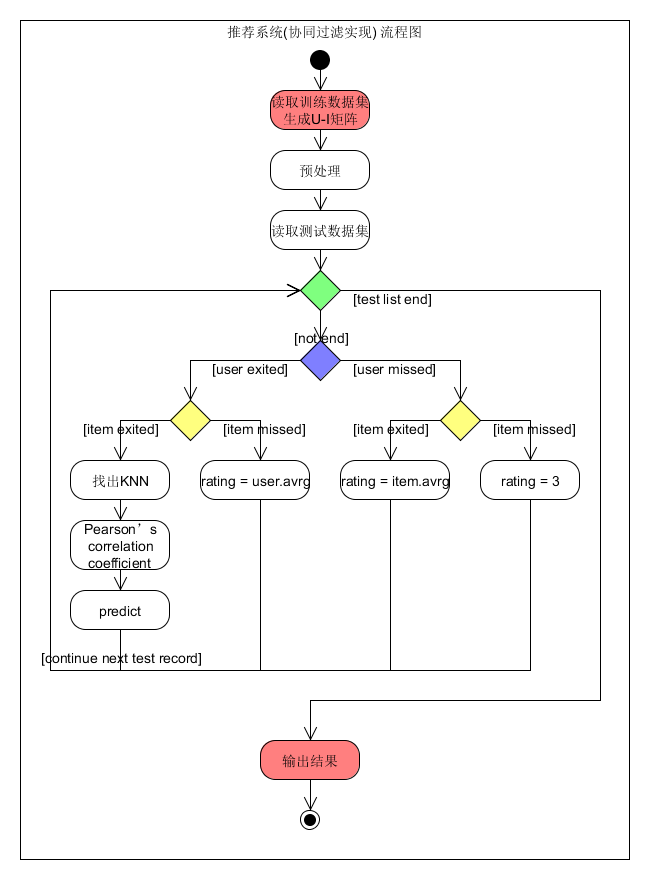


ru,i 表示用户u对物品i的评分，ru 和rv分别表示用户u和用户v对物品i的平均评分

1. 根据用户集预测评分的算法为



V是存在k个相似用户的用户集。

****

**四，参数设置分析**

//KNN –K的设置

USER\_KNEIGHBOURS: 20

ITEM\_KNEIGHBOURS: 10

//相似性阈值

USER\_SIMILARITY\_CUT: 0.2

ITEM\_SIMILARITY\_CUT: 0

USER\_SIGNIFICANCE\_WEIGHTING: 150

ITEM\_SIGNIFICANCE\_WEIGHTING: 100

//User-Based 与Item-Based 比重

RATIO(USER): 0.5

RATIO(ITEM): 0.5

本实验先实现User-Based CF，后为方便比较，在原来的基础上实现Item-Based CF。

**五，实验结果分析**

MAE: 0.6778

RUN TIME: 494.484 sec == 8.5分钟左右

（硬件：CPU ：Inter Core i5-460M 和 4G 内存，操作系统：Apple Mac 此系统无限制CPU使用，可以100%）