基于内容的文章推荐系统实验报告

刘斌

清华大学软件学院  
17888825884

b\_liu14@163.com

李书昂

清华大学软件学院  
17888825866

lsa0924@163.com

熊世裕

清华大学软件学院  
18401653036

pxxsy@163.com

**摘要**

在这篇报告中，我们主要说明了我们在做这么一个文章推荐系统时采用的推荐方法以及使用到的由台湾国立大学开发的矩阵分解技术：LibMF。我们还会说明我们是如何利用这个推荐系统和已有的数据进行实验的。最后展示的是我们的实验结果和结论。

**关键词**

推荐系统，矩阵分解，LibMF

# 引言

互联网的出现和普及给用户带来了大量的信息，满足了用户在信息时代对信息的需求，但随着网络的迅速发展而带来的网上信息量的大幅增长，使得用户在面对大量信息时无法从中获得对自己真正有用的那部分信息，对信息的使用效率反而降低了，这就是所谓的信息超载问题。解决信息超载问题一个非常有潜力的办法是推荐系统，它是根据用户的信息需求、兴趣等，将用户感兴趣的信息、产品等推荐给用户的个性化信息推荐系统。和搜索引擎相比推荐系统通过研究用户的兴趣偏好，进行个性化计算，由系统发现用户的兴趣点，从而引导用户发现自己的信息需求。一个好的推荐系统不仅能为用户提供个性化的服务，还能和用户之间建立密切关系，让用户对推荐产生依赖。而文章推荐系统无疑可以帮助我们在这个每天都有成千上万论文发表的时代快速找到适合我们阅读的文章，提高我们的科研学习效率。

# 正文

# 2.1现有推荐方法[1]

**（1）基于内容推荐**

基于内容的推荐（Content-based Recommendation）是信息过滤技术的延续与发展，它是建立在项目的内容信息上作出推荐的，而不需要依据用户对项目的评价意见，更多地需要用机器学习的方法从关于内容的特征描述的事例中得到用户的兴趣资料。在基于内容的推荐系统中，项目或对象是通过相关的特征的属性来定义，系统基于用户评价对象的特征，学习用户的兴趣，考察用户资料与待预测项目的相匹配程度。用户的资料模型取决于所用学习方法，常用的有决策树、神经网络和基于向量的表示方法等。基于内容的用户资料是需要有用户的历史数据，用户资料模型可能随着用户的偏好改变而发生变化。

基于内容推荐方法的优点是：

1）不需要其它用户的数据，没有冷开始问题和稀疏问题。

2）能为具有特殊兴趣爱好的用户进行推荐。

3）能推荐新的或不是很流行的项目，没有新项目问题。

4）通过列出推荐项目的内容特征，可以解释为什么推荐那些项目。

5）已有比较好的技术，如关于分类学习方面的技术已相当成熟。

缺点是要求内容能容易抽取成有意义的特征，要求特征内容有良好的结构性，并且用户的口味必须能够用内容特征形式来表达，不能显式地得到其它用户的判断情况。

**（2）协同过滤推荐**

协同过滤推荐（Collaborative Filtering Recommendation）技术是推荐系统中应用最早和最为成功的技术之一。它一般采用最近邻技术，利用用户的历史喜好信息计算用户之间的距离，然后利用目标用户的最近邻居用户对商品评价的加权评价值来预测目标用户对特定商品的喜好程度，系统从而根据这一喜好程度来对目标用户进行推荐。协同过滤最大优点是对推荐对象没有特殊的要求，能处理非结构化的复杂对象，如音乐、电影。

协同过滤是基于这样的假设：为一用户找到他真正感兴趣的内容的好方法是首先找到与此用户有相似兴趣的其他用户，然后将他们感兴趣的内容推荐给此用户。其基本思想非常易于理解，在日常生活中，我们往往会利用好朋友的推荐来进行一些选择。协同过滤正是把这一思想运用到电子商务推荐系统中来，基于其他用户对某一内容的评价来向目标用户进行推荐。

协同过滤可分为：

1. item-based CF

基于item的协同过滤，通过用户对不同item的评分来评测item之间的相似性，基于item之间的相似性做出推荐；

1. user-based CF

基于user的协同过滤，通过不同用户对item的评分来评测用户之间的相似性，基于用户之间的相似性做出推荐；

基于协同过滤的推荐系统可以说是从用户的角度来进行相应推荐的，而且是自动的，即用户获得的推荐是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。

和基于内容的过滤方法相比，协同过滤具有如下的优点：

1） 能够过滤难以进行机器自动内容分析的信息，如艺术品，音乐等。

2） 共享其他人的经验，避免了内容分析的不完全和不精确，并且能够基于一些复杂的，难以表述的概念（如信息质量、个人品味）进行过滤。

3） 有推荐新信息的能力。可以发现内容上完全不相似的信息，用户对推荐信息的内容事先是预料不到的。这也是协同过滤和基于内容的过滤一个较大的差别，基于内容的过滤推荐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。

4） 能够有效的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）。

**（3）基于知识推荐**

基于知识的推荐（Knowledge-based Recommendation）在某种程度是可以看成是一种推理（Inference）技术，它不是建立在用户需要和偏好基础上推荐的。基于知识的方法因它们所用的功能知识不同而有明显区别。效用知识（Functional Knowledge）是一种关于一个项目如何满足某一特定用户的知识，因此能解释需要和推荐的关系，所以用户资料可以是任何能支持推理的知识结构，它可以是用户已经规范化的查询，也可以是一个更详细的用户需要的表示。

**（4）基于矩阵分解的模型**

**（5）组合推荐**

由于各种推荐方法都有优缺点，所以在实际中，组合推荐（Hybrid Recommendation）经常被采用。研究和应用最多的是内容推荐和协同过滤推荐的组合。最简单的做法就是分别用基于内容的方法和协同过滤推荐方法去产生一个推荐预测结果，然后用某方法组合其结果。尽管从理论上有很多种推荐组合方法，但在某一具体问题中并不见得都有效，组合推荐一个最重要原则就是通过组合后要能避免或弥补各自推荐技术的弱点。

在组合方式上，有研究人员提出了七种组合思路：

1）加权（Weight）：加权多种推荐技术结果。

2）变换（Switch）：根据问题背景和实际情况或要求决定变换采用不同的推荐技术。

3）混合（Mixed）：同时采用多种推荐技术给出多种推荐结果为用户提供参考。

4）特征组合（Feature combination）：组合来自不同推荐数据源的特征被另一种推荐算法所采用。

5）层叠（Cascade）：先用一种推荐技术产生一种粗糙的推荐结果，第二种推荐技术在此推荐结果的基础上进一步作出更精确的推荐。

6）特征扩充（Feature augmentation）：一种技术产生附加的特征信息嵌入到另一种推荐技术的特征输入中。

7）元级别（Meta-level）：用一种推荐方法产生的模型作为另一种推荐方法的输入。

在综合比较上述方法以及考虑我们实验数据的规模之后，我们选择了。

**2.2 本实验采取的推荐算法-矩阵分解**

如果把用户和文章一一对应起来构建一个矩阵，用户读过某篇文章则把相应的位置标记为1，没有读过某篇文章则标记为“？”，那么我们的推荐系统要做的就是通过已知的被标记为1的位置预测出“？”的值，预测的值越大则说明这篇文章越值得推荐。

而矩阵分解便可实现这个预测的操作。矩阵分解的思想简单来说就是每一个用户和每一个物品都会有自己的一些特性，用矩阵分解的方法可以从评分矩阵中分解出用户——特性矩阵，特性——物品矩阵，这样做的好处一是得到了用户的偏好和每件物品的特性，二是降低了矩阵的维度。例如若评分矩阵为n\*m的一个矩阵，那么矩阵分解便可得到n\*k的用户矩阵以及k\*m的物品矩阵；然和再利用这两个矩阵相乘得到的矩阵应满足：若在评分矩阵中某个位置为已知的值，那么这个进行预测的矩阵在这个位置的值应接近这个已知的值。而在评分矩阵中被标记为“？”的位置在这个进行预测的矩阵中便是相应的推荐值。

在本次实验中，我们采用正是基于矩阵分解的推荐系统开源框架：libMF。

libMF是由台湾大学林智仁老师实验室开发的，一个用于推荐系统领域的矩阵分解开源库。LibMF在矩阵分解的并行化方面作出了很好的贡献，针对SDG优化方法在并行计算中存在的locking problem和memory discontinuity问题，提出了一种矩阵分解的高效算法，根据计算节点的个数来划分评分矩阵block，并分配计算节点。详细的算法介绍可以参见论文[3]。

# 实验及实验结果

由于对推荐的结果没有正确答案，无法进行precision和recall的计算。因此我们将train.txt中的数据集提取一部分（每个用户的60%）作为新的训练数据集train.txt，剩下的一部分（每个用户的40%），作为正确的结果进行检验，这就是answer.txt。此外，对于测试集，我们在上诉40%的数据的基础上，再随机选取一些不在train.txt出现过的数据，使得每个用用户对应251条测试数据。通过这样的方法，我们就得到了新的测试集test.txt。从而我们便可以通过train.txt训练模型，通过test.txt产生推荐结果然后通过answer.txt计算precison和recall。从而调整我们的参数，使得推荐结果达到最优。

在最开始时，我们采用了协同过滤的方法进行推荐，即itemBased CF和userBased CF。然而我们在通过测试发现这个方法得到的结果准确率较低，只有10%左右。

所以我们放弃了这个方法而采用了矩阵分解的方法进行推荐。我们使用了libMF框架，首先使用mf-train，通过输入的train.txt对模型进行训练，最终将得到的模型输出到model.txt中。接着使用mf-predict，在输入的test.txt中，对于每个用户对应的251本待推荐的书，通过model.txt载入模型后计算每一本书的推荐系数，然后按照推荐系数排序，将结果最高的5本书写入recommend.txt中。此外，在调参阶段，我们还使用了mf-test，输入recommend.txt和answer.txt，计算推荐结果的precision和recall,并不断调整参数以使得效果最优。在采用了矩阵分解的方法后，我们的推荐系统的准确率也提高到了42%左右。

# 致谢

感谢台湾国立大学提供的开源LibMF库帮助我们完成了矩阵分解的工作。

感谢靳晓明老师和助教辛苦工作，为我们提供了详细的文档介绍以及相关文献和代码包，对我们的工作有很大帮助！

# 引用

[1] 对推荐方法的介绍我们引用了博客：

http://blog.sina.com.cn/s/blog\_73de143c01014uy6.html

[2] http://blog.csdn.net/cserchen/article/details/14231153

[3] Y. Zhuang, W.-S. Chin, Y.-C. Juan, and C.-J. Lin. A Fast Parallel SGD for Matrix Factorization in Shared Memory Systems.