**《基于华中师范大学图书借阅记录的图书推荐系统》**

**目录：**

1. **推荐算法**

**1.1根据人口统计学的推荐**

**1.2基于内容的推荐**

**1.3基于关联规则的推荐**

**1.4基于协同过滤的推荐**

**1.4.1基于用户的推荐**

**1.4.2基于物品的推荐**

1. **结合算法进行分析**

**2.1优缺点**

**2.1.1冷启动问题**

**2.1.2领域无关**

**2.1.3系统性能**

**2.1.4潜在兴趣**

**2.1.5热度过度推荐**

**2.2实现所需数据要求**

1. **具体实现**

**1.推荐算法**

**推荐**：根据用户兴趣和行为特点，向用户推荐所需要的信息或者商品，帮助用户在海量信息中快速发现真正需要的商品，提高用户黏性，促进信息点击和商品销售。推荐系统是基于海量数据挖掘分析的商业智能平台，推荐主要基于以下信息：（推荐算法显然在用户方的利益是可以减少时间和精力的花费，从而达到服务大众的作用，嗯。）

1.热点信息

2.用户信息

3.用户历史浏览或者行为记录

4.社会化关系

【如果是基于借阅信息的推荐】：

1.借阅者的所属院系

2.借阅者的借阅记录

3.借阅者的身份

**1.1--根据人口统计学--**

简单的根据用户的基本信息发现用户的相关程度，然后将相似用户喜爱的其他物品推荐给当前用户。

class profile{

age

male\_or\_female

and\_so\_on

..

}

if A.profile ≈ B.profile

bind(A,B)

B.history-->A

**好处：**

1.因为不使用当前用户对于物品的喜好历史数据，所以对于新用户来讲没有“冷启动”的问题。

2.这个方法不依赖于物品本身的数据，所以这个方法在不同物品的领域都可以使用，他是领域独立的（domain-independent）。

**缺点：**

粗糙（图书等领域，得不到很好的推荐效果）可能涉及到一些与信息发现问题本身无关却比较敏感的信息。

**1.2--基于内容的推荐--**

**核心思想**：

根据推荐物品或内容的元数据，发现物品或者内容的相关性，然后基于用户以往的喜好记录，推荐给用户相似的物品

这种推荐系统多用于一些资讯类的应用上，针对文章本身抽取一些tag作为该文章的关键词，继而可以通过这些tag来评价两篇文章的相似度。

**优点**：

1.易于实现，不需要用户数据，因此不存在稀疏性和冷启动问题。

2.基于物品本身特征推荐，因此不存在过度推荐热门这个问题。

**缺点**:

对于tag的抽取：既要保证准确性又要具有实际意义，否则很难保证推荐结果的相关性。

**1.3--基于关联规则的推荐--**

**首要目标**：

挖掘出关联规则，也就是那些同时被很多用户购买的物品集合，这些集合内的物品可以相互进行推荐。

演变来源：Apriori和FP-Growth

基于关联规则的推荐系统一般转化率较高因为当用户已经购买了频繁集合中的若干项目后，购买该频繁集合中其他项目的可能性更高。

**缺点**：

1.计算量较大，但是可以离线计算，因此影响不大。

2.由于采用用户数据，不可避免的存在冷启动和稀疏性问题。

3.存在热门项目容易被过度推荐的问题。

**1.4--基于协同过滤的推荐--**

这种算法基于一个“物以类聚，人以群分”的假设，喜欢相同物品的用户更有可能具有相同的兴趣。

基于协同过滤的推荐系统一般应用于有用户评分的系统之中，通过分数去刻画用户对于物品的喜好。

通过用户建立物品与物品之间的联系。

**1.4.1-->①User-based**

**--基于用户的推荐--**

**基本原理：**

根据所有用户对物品或者信息的偏好，发现与当前用户口味和偏好相似的“邻居”用户群，在一般的应用中是采用计算“K-Nearest Neighboor”的算法；然后，基于这个K个邻居的历史偏好信息，为当前用户进行推荐。

**优点:**

推荐物品之间在内容上可能完全不相关，因此可以发现用户的潜在兴趣，并且针对每个用户生成其个性化的推荐结果。

**缺点：**

在于一般的Web系统中，用户的增长速度远远大于物品的增长速度，因此计算量的增长巨大，系统性能容易成为瓶颈。

**1.4.2-->②Item-based**

**--基于物品的推荐--**

**基本原理：**

使用所有用户对物品或者信息的偏好，发现物品和物品之间的相似度，然后根据用户的历史偏好信息，将类似的物品推荐给用户。

基于物品的协同过滤可以看做是关联规则推荐的一种退化，但由于协同过滤更多考虑了用户的实际评分，并且知识计算相似度而非寻找频繁集，因此可以认为基于物品的协同过滤准确率较高并且覆盖率较高。

**优点：**

与基于用户的推荐相比，这种推荐方法应用更为广泛，扩展性和算法性能更好，由于项目的增长速度一般较为平缓，因此性能变化不大。

**缺点:**

无法提供个性化的推荐结果。

**1.4.3基于协同过滤的推荐机制**是现今应用最为广泛的推荐机制，**优点**如下：

1.不需要对物品或者用户进行严格的建模，而且不要求物品的描述是机器可以理解的，所以这种方法是领域无关的。

2.这种方法计算出来的推荐是开放的，可以共用他人的经验，很好的支持用户发现潜在的兴趣偏好。

**缺点:**

1.方法的核心是基于历史数据，所以对新物品和新用户都有冷启动的问题。

2.推荐的效果依赖于用户历史偏好数据的多少和准确性。

3.大部分的实现中，用户历史偏好是用稀疏矩阵进行存储的，而稀疏矩阵上的计算有明显的问题，包括可能少部分人的错误偏好会对推荐的准确度有很大的影响。

4.对于一些特殊品味的用户不能给予很好的推荐。

5.由于以历史数据为基础，抓取和建模用户的偏好后，很难修改或者根据用户的使用演变，从而导致这个方法不够灵活。

1. **结合算法进行分析**

**2.1优缺点：**

不同的推荐算法存在不同的问题：冷启动，领域无关，系统性能，潜在兴趣，热度物品被过度推荐等。

**2.1.1冷启动:**

**描述**：

如何在没有大量用户行为数据的情况下设计个性化推荐系统并且让用户对推 荐系统满意，从而愿意使用推荐系统，这就是冷启动问题。

**应对**：

这个问题在于，我们的用户群分为两类，一类是有一定借阅记录基础的用户（暂 且称之为老用户）一类是新生或者是没什么借阅记录的用户（暂且称之为新用户）。 对于老用户，冷启动问题几乎不会干扰算法的性能，但是对于新用户，冷启动就会 卡住一部分算法的实现。

因此我们需要区别对待，对于老用户使用存在冷启动问题的算法，对于新用户 使用不存在冷启动问题的算法。

**如何确定新老用户？**

--个人看法--：

问题可以转化为：确定一个常数C，使得：

·(0<=p<C) 新用户

·(C<=p) 老用户 p为借阅数。

对于C的确定，我们可以利用二分法，对区间(1,MAX)进行log2MAX次迭 代，每次都对一个示例用户进行推荐，并且进行人工的评分，最终确定一个评 分最高的C。.

**2.1.2领域无关：**

**描述：**

我是这么理解的，就是书籍能否跨领域推荐，比如能否根据一本心理学的书籍 而推荐出一本计算机科学的书籍。

**应对：**

对于这个问题，自然是能推荐更好。因此不作为主要因素去进行考虑。

**2.1.3系统性能：**

由于现在是理论阶段的讨论和研究，所以以有足够的系统性能为支撑。

并且，个人估计数据不会达到很高的量，像淘宝或者Amazon一样。因此暂时忽略。

**2.1.4潜在兴趣：**

在一个简单的系统中不需要涉及这方面的内容，因为这个对于系统增加的负担和实 际效果并不成正比。但是这个功能可以放在后续研发上。在初期阶段不予考虑。

**2.1.5热度书籍过度推荐：**

这个问题我们可以写一个热度榜单，另行放置，因为统计出热度最高的几本书并不 是一个难事，我们只需要把这些书列一个Top10的榜单，然后剔除这些书籍被推荐的机 会。就可以解决这个问题了。

**2.2实现所需的数据要求:**

**2.2.1基于人口统计学的推荐：**

·用户的基本信息。

**2.2.2基于内容的推荐：**

·书籍的标签(tag)。

**2.2.3基于关联规则的推荐：**

·书籍之间的关联规则，即经常被一群人借阅的书籍集合。

**2.2.4基于协同过滤的推荐：**

·用户对书籍的评分。

**总结：**

很显然，对于书籍的标签，如果数据中没有，获取tag是一件很繁重的工作，且不 容易实现，而用户对于书籍的评分也只能在新数据中获取。

所以因为数据的限制，只能选择**基于人口统计学的推荐**和**基于关联规则的推荐**。

**2.3结论:**

**综上所述：**

>·**对于新用户：**使用**基于人口统计学**的推荐。

>·**对于老用户：**使用**基于关联规则**的推荐。

1. **具体实现**

**（待续）**