推荐系统简介及其R语言实现

□ 中国联通成都市分公司　刘 可

摘 要：互联网的快速发展普及带来了信息过载，个性化推荐系统得到了快速发展。本文对推荐系统进行了分类简介，并对其R语言代码实现进行了介绍。

关键词：推荐；R语言

1 推荐系统出现的背景

互联网的出现和普及为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，也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，传统的搜索算法只能呈现给所有用户一样的排序结果，无法针对不同用户的兴趣爱好提供相应的信息反馈服务。信息的爆炸使得信息的利用率反而降低，这就是所谓的信息过载问题。

解决信息过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，个性化推荐系统通过建立用户与信息产品之间的二元关系，利用已有的选择过程或相似性关系挖掘每个用户潜在感兴趣的对象，进而进行个性化推荐，从而引导用户发现自己的信息需求。推荐问题从根本上说是代替用户评估它从未看过的产品，这些产品包括书、电影、CD、网页、甚至可以是饭店、音乐、绘画等等。一个好的推荐系统不仅能为用户提供个性化的服务，还能通过提供良好的服务，与用户之间建立密切关系，增加用户的粘性。

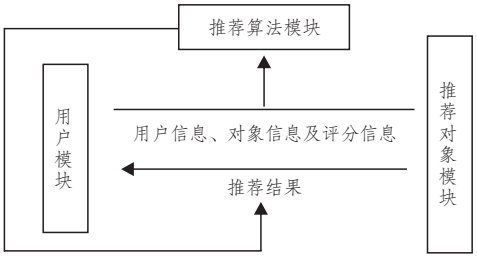

图1 推荐系统模型流程图

推荐系统有三个重要的模块：用户模块、推荐对象模块、推荐算法模块。通用的推荐系统模型流程如图1所示。推荐系统把用户模型中的兴趣需求信息和推荐对象模型中的特征信息匹配，同时使用相应的推荐算法进行计算筛选，找到用户可能感兴趣的推荐对象，然后推荐给用户。

2 推荐系统的分类

根据推荐系统中推荐算法的不同，推荐系统主要分为以下三种。

2.1 基于内容的推荐

基于内容的推荐（Content-based Recommendation）是信息过滤技术的延续与发展，它是在项目的内容信息上作出推荐，而不需要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（item）是通过相关特征的属性来定义，系统基于评价对象的特征，学习用户的兴趣，考察用户资料与待预测项目（item）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“冷启动”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；

6)对用户兴趣可以很好的建模，并通过对物品属性维度的增加，获得更好的推荐精度。

缺点是:

1）要求内容物品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；

2）物品的属性有限，很难有效的得到更多数据；

3）物品相似度的衡量标准只考虑到了物品本身，有一定的片面性；

4） 如果两个不同的产品恰好用相同的特征词表示，这两个产品就无法区分。

2.2 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，是第一代被提出并得到广泛应用的推荐系统。如Amazon的书籍推荐，Jester的笑话推荐等等。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或物品（item）之间的距离，然后利用目标用户的最近邻居用户对物品评价的加权评价值来预测目标用户对特定物品的喜好程度，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、电影等。

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互联网推荐系统中来，基于其他用户对某一内容的评价来向目标用户进行推荐。

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，不需要用户主动提供自己感兴趣的推荐信息，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）它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理解的，所以这种方法也是领域无关的，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；

5） 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。

协同过滤的缺点是：

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）在大部分的实现中，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些明显的问题，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等等；

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。

协同过滤系统的关键在于相似度的计算，根据相似度计算对象的不同，又分为基于用户的协同过滤系统和基于内容的协同过滤系统。在计算相似度时，大部分都是基于用户对内容的评分矩阵，最常用的相似度计算方法是Pearson相关性和夹角余弦。

2.3 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推荐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。

因为基于用户的协同过滤系统和基于内容的协同过滤系统的不同之处在于相似度计算的对象不同，体现在数学表达式上则是评分矩阵行列的转置，基于内容的推荐方法与基于内容的协同过滤系统区别主要在于前者的评分矩阵式人工标注，后者是基于用户行为的记录。因此，以下将以基于内容的协同过滤系统为例进行介绍。

3 基于内容的协同过滤系统的R语言实现

3.1 用K近邻的方法自行编写R代码

假设用户—内容的评分矩阵为rating_matrix,矩阵的行对应于各个用户对各个内容item的评分值。为减少用户反馈难度，考虑最简单的情况，一般取值为二进制评分，用户喜欢为1，一般对应用户选用了该内容item；不喜欢为

0，一般对应用户没有选用该内容item。矩阵的列对应于各个内容item在各个用户处获得的评分。以一个电影推荐系统的评分矩阵为例，则其第一列是用户ID，第一行是各个电影item的ID，第i行j列的分x,则代表第i个用户对第j部电影的评分为x，如果第i个用户选取看了第j部电影，则对应的评分rating.matrix [i, j]=1，否则为0。

首先将用户ID和内容ID分别存入row.names和row.names向量：

```
row.names(rating.matrix) <- rating.matrix [, 1]
col.names(rating.matrix) <- rating.matrix [1, ]
```

将评分矩阵中的用户ID和内容ID去掉，得到仅仅包含评分值的用户—内容评分矩阵：

```
rating.matrix <- rating.matrix [, -1]
rating.matrix <- rating.matrix [-1, ]
```

相似度的计算采用Pearson相关性：

```
similarities <- cor(rating.matrix)
```

采用夹角余弦相似性：

```
z<-matrix(rep(sqrt(colSums(rating.matrix^2)),nrow(rating.matrix)),nrow = nrow(rating.matrix))
 xx<- rating.matrix /z
similarities <-t(xx)%*%xx
```

相似距离的计算：

```
 distances <- -log((similarities / 2) + 0.5)
```

K近邻的计算(考虑25个近邻点)

```
k.nearest.neighbors <- function(i, distances, k = 25)
 {
 return(order(distances[i, ])[2:(k + 1)])
}
```

针对用户user推荐item的概率计算：

```
recommend.probability <- function(user, item, rating.matrix, distances, k = 25)
{
neighbors <- k.nearest.neighbors(ietm, distances, k = k)
return(mean(sapply(neighbors, function (neighbor) { rating.matrix [user, neighbor]})))
}
```

针对用户user推荐item按照概率的排序的item ID：

```
most.probable.items <- function(user, rating.matrix, distances, k = 25)
{
return(order(sapply(1:ncol(rating.matrix),
function (item)
{
recommend.probability(user,
item,
rating.matrix,
distances,
k = k)
}),
decreasing = TRUE))
}
```

比如，推荐概率前10位的item：

```
colnames(rating.matrix)[listing[1:10]]
```

采用R语言自行编写推荐系统代码，优点是可以根据具体应用灵活调整推荐算法及参数，缺点是代码相对更复杂。

3.2 采用R语言中的推荐工具包Recommender

```
r<- Recommender(data, method, parameter)
```

生成推荐模型r, 其中data为评分矩阵，method为推荐算法，包括随机模型(random items)， 流行模型（popular items），基于用户的协同过滤（user-based CF）和基于内容的协同过滤（item-based CF）。

采用R语言中的推荐工具包Recommender的优点是代码简洁，缺点是算法调整不灵活。

4 结束语

个性化推荐是解决信息过载、提高用户使用满意度的有效手段，本文对推荐系统及其主要分类进行了简介，并对R语言环境下的代码实现进行了分析介绍。随着电信运营向互联网的转型逐渐深入，推荐系统在诸如IPTV、运营商电商网站等方面将会得到更大的发展。

参考文献

[1]Dietmar Jannach.Recommendation Systems.蒋凡等译.推荐系统[M].北京:人民邮电出版社,2013.

[2] 李明.R语言与网站分析[M].北京:机械工业出版社,2014.

[3] Drew Conway,John Myles White.陈开江,刘逸哲等译.机器学习:实用案例解析[M].北京:机械工业出版社,2014.

[4] recommenderlab: A Framework for Developing and Testing Recommendation Algorithms.http://cran.r-project.org/

[5] 推荐系统.百度百科.

作者简介

刘可（1976年—），现在成都联通负责信息化工作，主要研究方向是信息安全、系统构架设计以及通信网络。