第13讲 推荐系统

海量数据计算研究中心

杨东华

目录

目录

为什么需要推荐系统

什么是稀缺资源

货架空间是稀缺资源

传统零售商的货架空间是稀缺资源。

注意力成为了稀缺资源

然而网络使零成本产品信息传播成为可能，“货架空间”从稀缺变得丰富。人们逐渐从信息匮乏的时代走入了信息过载的时代。这时，注意力便成了稀缺资源。

为什么需要推荐系统

推荐系统

旨在向用户提供建议

推荐系统的目的向用户提供建议。

推荐系统的价值

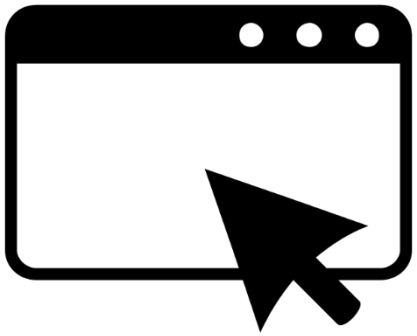
将正确的商品在正确的时间推荐给正确的人，这在商业上有巨大巨大价值。

在阅读网站上，一个合适的推荐远比首页上的图书更加吸引用户。

推荐系统的应用

在线商城

当用户进入商城首页后，就会看到系统根据用户的的历史行为推荐了丰富的商品。



个性化阅读

推荐在个性化阅读中也有广泛的应用，一个典型的例子就是豆瓣，其根据用户历史对书籍的打分为用户推荐可能喜欢的书籍。

电影推荐

推荐系统在电影的推荐中有着广泛的应用。国内的一些影视类网站大都有自己的推荐系统，比如爱奇艺、优酷、土豆等。

目录

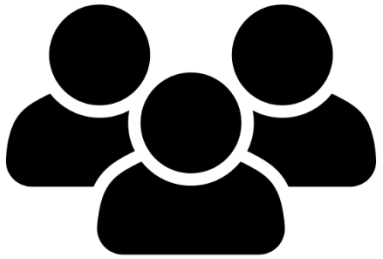
推荐算法概述

推荐结果是否因人而异

大众化推荐

在同样的外部条件下，不同用户获得的推荐是一样的。

如查询推荐，它往往只与当前的query有关，很少与用户直接相关。



推荐算法概述

推荐结果是否因人而异

大众化推荐

在同样的外部条件下，不同用户获得的推荐是一样的。

如查询推荐，它往往只与当前的query有关，很少与用户直接相关。

个性化推荐

不同的人在不同的外部条件下，可以获得与其本身兴趣爱好、历史记录等相匹配的推荐。

推荐算法概述

不同的推荐方法

1 基于人口统计学的推荐

Demographic-based Recommendation

2 基于内容的推荐

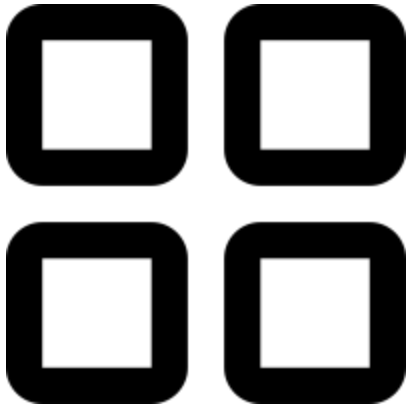
Content-based Recommendation

3 基于协同过滤的推荐

Collaborative Filtering-based Recommendation

4 混合型推荐方式

Hybrid Recommendation



基于人口统计学的推荐

主要思想

一个用户可能会喜欢与其相似的用户所喜欢的东西。

实施方法

记录每一用户的性别、年龄、活跃时间等元数据。

当我们需要对一个用户进行个性化推荐时，利用元数据计算与其他用户之间的相似度，并选出最相似的几个用户，进而利用这些用户的购买记录进行推荐。

优点

计算简单。

缺点

可信度较低。即便是性别、年龄等源数据属性相同的用户，也可能在物品上有着截然不同的爱好。

基于人口统计学的方法在实际推荐系统中很少作为一个特点的方法单独使用，而常常与其他方法结合，利用用户元数据对推荐结果进行进一步的优化。

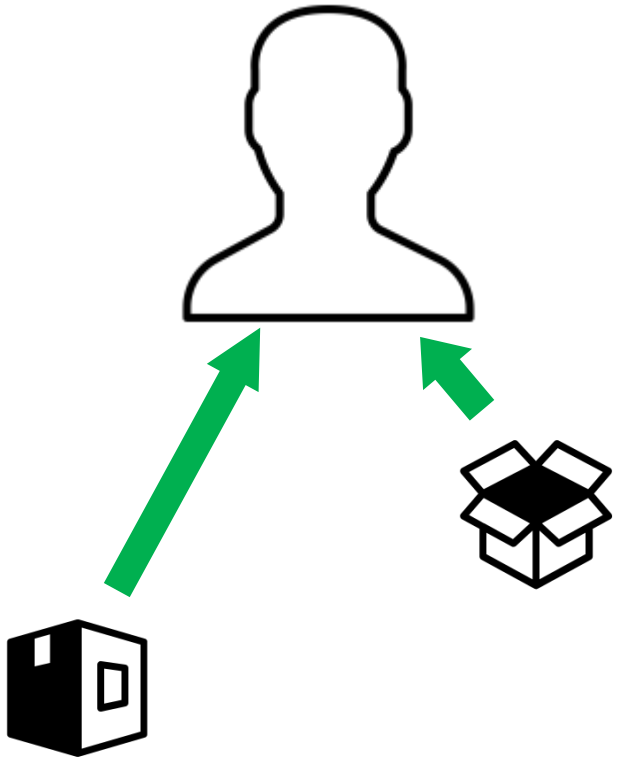
基于内容的推荐

主要思想

一个用户可能会喜欢和他曾经喜欢过的物品相似的物品。

用户画像

一种简单的推荐方法是，考虑该用户曾经购买或浏览过的所有物品，并将这些物品的内容信息加权整合作为对应用户的画像。然后计算用户画像和其他物品之间的相似度。



推荐算法概述

基于内容的推荐

优点

- ✓ 很好地解决了新物品的冷启动问题。
- ✓ 推荐结果有较好的可解释性。

推荐算法概述

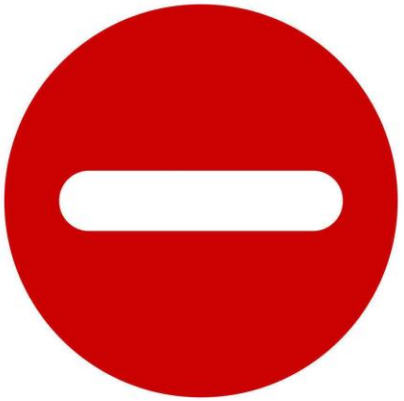
基于内容的推荐

优点

- ✓ 很好地解决了新物品的冷启动问题。
- ✓ 推荐结果有较好的可解释性。

缺点

- ◘ 为了得到物品的特征，系统需要复杂的模块甚至手工的方法来对物品信息进行预处理。
- ◘ 很难发现用户并不熟悉但是具有潜在兴趣的物品。

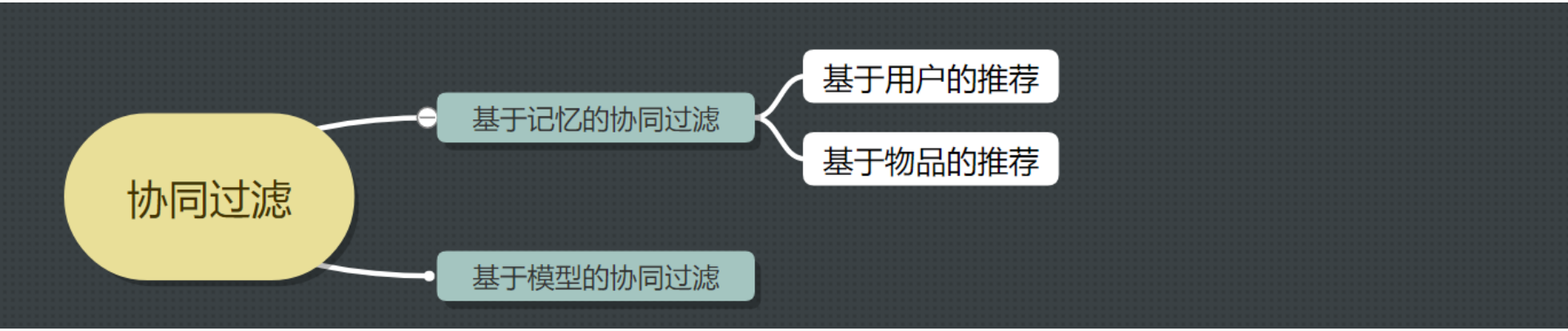


推荐算法概述

基于协同过滤的推荐

基于协同过滤的推荐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对特定商品的喜好程度，从而根据这一喜好程度对目标用户进行推荐。



混合型推荐方式

各有优缺点

由于各种推荐方法都有优缺点，所以在实际应用中，组合推荐经常被采用。

一个例子

研究和应用最多的是基于内容的推荐和协同过滤推荐的组合。

最简单的做法就是分别基于内容的方法和基于协同过滤的方法产生一个预测解决，然后用某种方法组合其结果。

目录

基于协同过滤的算法

基于用户的协同过滤

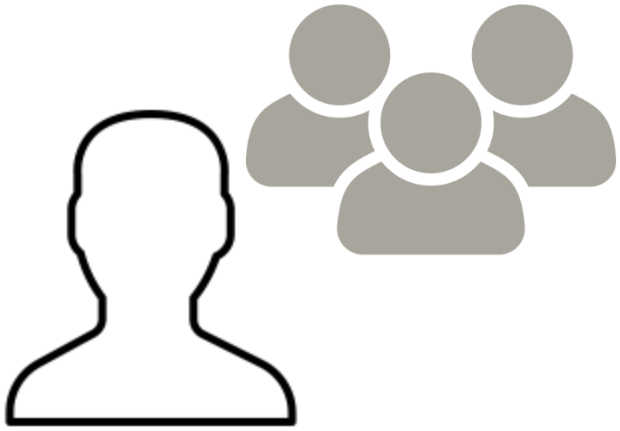
基本思想

用户可能喜欢和他具有相似爱好的用户所喜欢的物品。

推荐过程

对于特定用户A，当我们找到和他相似的用户，我们便把他们购买的物品推荐给用户A。

这个方法的核心是如何量化用户之间的相似度。



基于协同过滤的算法

基于用户的协同过滤

用户对商品的评分

采用一定的策略获取用户对商品的评分。如用户浏览一个商品，则分数+0.1；当他购买了这件商品，则分数+1。

计算用户相似度

使用$a_1 = [10,7,2,4,0]$作为用户00001的特征，$a_2 = [8,7,8,5,8]$作为用户00002的特征。

用户ID	白酒	红酒	女装	男装	运动鞋
00001	10	7	2	4	
00002	8	7	8	5	8
00003	8	6	4		5
00004		3	8	8	3
00005	2	5	4	4	
00006		2	9	7	3

基于协同过滤的算法

基于用户的协同过滤

计算用户相似度

使用$a_1 = [10,7,2,4,0]$作为用户00001的特征，$a_2 = [8,7,8,5,8]$作为用户00002的特征。

这里使用余弦相似度。那么用户00001和用户00002的相似度为

用户ID	白酒	红酒	女装	男装	运动鞋
00001	10	7	2	4	
00002	8	7	8	5	8
00003	8	6	4		5
00004		3	8	8	3
00005	2	5	4	4	
00006		2	9	7	3

$$\mathrm{sim}(00001, 00002) = \cos(a_1, a_2) = \frac{a_1 \cdot a_2}{|a_1| \cdot |a_2|} \approx 0.89$$

基于协同过滤的算法

基于用户的协同过滤

基于用户带来的问题

- ·**算法扩展性**· 随着用户数量的增加，其时间代价也显著增长。
- ·**数据稀疏性**· 往往用户所购买的不到一个商城所有物品的1%，因此用户之间的相似性可能不准确。

因此，研究人员提出了基于物品的协同过滤。

基于协同过滤的算法

基于物品的协同过滤

基本思想

基本思想与基于内容的推荐相似，都是计算物品之间的相似性。

但两者计算的角度不同。在基于物品的方法中，两个物品之间的相似性是由购买二者的用户群体之间的相似度决定的。

让我们举个例子说明一下。

基于协同过滤的算法

基于物品的协同过滤

物品间的相似性

右表表示一个系统中用户的消费情况。如第一行表示用户A购买了物品a,b,d。

用户	物品
A	a, b, d
B	b, c, e
C	c, d
D	b, c, d
E	a, d

基于协同过滤的算法

基于物品的协同过滤

物品间的相似性

右表表示一个系统中用户的消费情况。如第一行表示用户A购买了物品a,b,d。

为计算物品之间的相似性，我们重新组织一下表格。

	a	b	c	d	e
A	1	1		1	
B		1	1		1
C			1	1	
D		1	1	1	
E	1			1	

基于协同过滤的算法

基于物品的协同过滤

物品间的相似性

使用$b_1 = [1,0,0,0,1]$作为物品a的特征，$b_2 = [1,1,0,1,0]$作为物品b特征。

仍使用余弦相似度。那么物品a和物品b的相似度为

	a	b	c	d	e
A	1	1		1	
B		1	1		1
C			1	1	
D		1	1	1	
E	1			1	

$$\mathrm{sim}(a,b) = \cos(b_1, b_2) = \frac{b_1 \cdot b_2}{|b_1| \cdot |b_2|} \approx 0.41$$

基于协同过滤的算法

基于物品的协同过滤

相似性的调整

有些物品的销售量比较多，也就是平均评分比较高。这导致了这些物品与其他物品的相似度会偏高。

在计算相似性之前，将表中所有评分减去所在列的均值，使得每列的和为0。

	a	b	c	d	e
A	0.6	0.4	-0.6	0.2	-0.2
B	-0.4	0.4	0.4	-0.8	0.8
C	-0.4	-0.6	0.4	0.2	-0.2
D	-0.4	0.4	0.4	0.2	-0.2
E	0.6	-0.6	-0.6	0.2	-0.2

基于协同过滤的算法

基于物品的协同过滤

相似性的调整

用$c_1 = [0.6, -0.4, -0.4, -0.4, 0.6]$作为物品a的特征，$c_2 = [0.4, 0.4, -0.6, 0.4, -0.6]$作为物品b特征。

那么物品a和物品b调整后的相似度为

	a	b	c	d	e
A	0.6	0.4	-0.6	0.2	-0.2
B	-0.4	0.4	0.4	-0.8	0.8
C	-0.4	-0.6	0.4	0.2	-0.2
D	-0.4	0.4	0.4	0.2	-0.2
E	0.6	-0.6	-0.6	0.2	-0.2

$$\mathrm{sim}(a,b) = \cos(c_1, c_2) = \frac{c_1 \cdot c_2}{|c_1| \cdot |c_2|} \approx -0.17$$

基于协同过滤的算法

基于物品的协同过滤

预测用户是否喜欢某物品

得到物品之间的相似度后，我们使用加权和的方法，计算用户对他未购买的物品的喜好程度。

例如，用户A对物品c的喜好程度

	a	b	c	d	e
A	1	1		1	
B		1	1		1
C			1	1	
D		1	1	1	
E	1			1	

$$\text{fond}(A,c) = \frac{\text{sim}(a,c) * 1 + \text{sim}(b,c) * 1 + \text{sim}(d,c) * 1}{1+1+1}$$

当喜好程度$\text{fond}(A,c)$大于设定的阈值时，我们就将物品c推荐给用户A。

基于协同过滤的算法

基于物品的协同过滤

缺点

- ✓ **计算简单，容易实现实时响应**。物品被评分的变化往往比用户的变化低的多，因此物品相似度的计算一般可以采取离线完成、定期更新的方式，从而减少了线上计算。
- ✓ **可解释性比较好**。基于物品的推荐方法很容易让用户理解为什么推荐了某个商品。

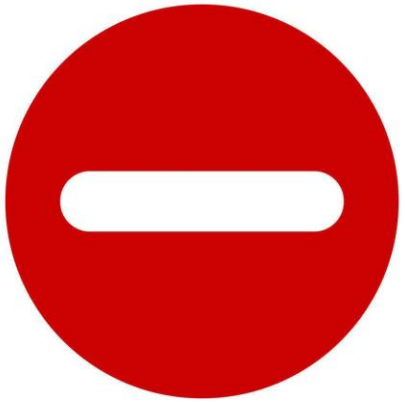
基于协同过滤的算法

基于物品的协同过滤

缺点

以物品为基础的信息过滤系统较少考虑用户之间的差别，因此精度与基于用户的方法往往稍微逊色一些。

除此之外，数据稀疏性和冷启动仍有待解决。。



基于协同过滤的算法

基于用户与基于物品的对比

计算复杂性

基于用户的方法往往时在线计算量大，难以实时响应。

✓ 用户数量大大超过物品数量

对于一个用户数量大大超过物品数量且物品数量相对稳定的应用，一般而言基于物品的方法从性能和复杂度上都比基于用户的方法更优。

这是因为物品相似度的计算不但计算量小，而且不必频繁更新。

✓ 物品数量巨大且频繁更新

而对于诸如新闻、博客或者微内容等物品数量巨大且频繁更新的应用，基于用户的方法往往更具优势。

基于用户的方法

OR

基于物品的方法

基于用户与基于物品的对比

适用场景

✓ 非社交网站

内容之间的内在联系是非社交网站中很重要的推荐原则，往往比基于相似用户的推荐原则更加有效。

如果向用户推荐图书并解释某个与该用户有相似兴趣的人也购买了被推荐的图书，是很难让目标用户信服的，因为该用户根本不认识那个“有相似兴趣的”用户；但如果解释为被推荐的图书与用户之前看过的图书相似，则更容易被用户接受，因为用户往往对自己的历史行为记录是非常熟悉和认可的。

✓ 社交网站

相反，在社交性网站中，基于用户的方法以及相关的基于用户网络的方法则是更不错的选择。因为基于用户的推荐方法加上社交网站中社会网络信息，可以大大增加用户对推荐解释的信服度。

基于协同过滤的算法

基于模型的协同过滤

计算规模庞大

基于用户和基于物品的方法共有的缺点就是计算规模庞大，并难以处理大规模数据量下的实时结果。

基于模型的协同过滤

基于模型的协同过滤则致力于改进该问题，首先利用历史数据训练得到一个模型，然后再用此模型进行预测。

基于协同过滤的算法

基于模型的协同过滤

使用的技术

基于模型的方法广泛使用的技术包括语义分析、贝叶斯网络、矩阵分解等。

思路

将用户属性和物品属性的各个特征作为输入，以用户打分作为输出来拟合模型，或者将打分作为类别转化为一个多分类器问题。

这种方式不是基于一些启发规则进行预测计算，而是对于已有数据应用统计和机器学习得到的模型进行预测。

基于协同过滤的算法

基于模型的协同过滤

优点

- ✓ ·**快速响应**· 只要训练出了模型就可以对新用户或新物品进行实时快速计算。
- ✓ ·**准确率高**· 由于可以直接以用户打分作为优化目标，往往可以获得较高的预测精度。

基于协同过滤的算法

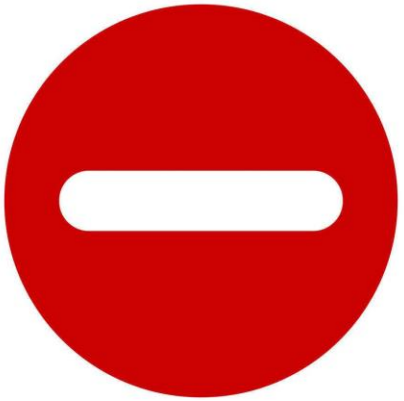
基于模型的协同过滤

优点

- ✓ ·**快速响应**· 只要训练出了模型就可以对新用户或新物品进行实时快速计算。
- ✓ ·**准确率高**· 由于可以直接以用户打分作为优化目标，往往可以获得较高的预测精度。

存在的问题

其问题在于如何将用户实时或者新增的喜好信息反馈给训练好的模型，从而在系统扩展的过程中维持推荐的准确度，也就是模型的增量训练问题。



目录

推荐方法的组合

组合方案

1. 加权融合（Weighted）

将多种推荐技术的结果加权混合产生推荐。

最简单的方式是基于感知器的线性混合，首先将协同过滤的推荐结果和基于内容的推荐结果赋予相同的权重值，然后比较用户对物品的评价与系统的预测是否相符，进而不断调整权值。

2. 变换（Switch）

根据问题背景和实际情况采取不同的推荐技术。例如，系统首先使用基于内容的推荐技术，如果不足以产生高可信度的推荐就转而尝试使用协同过滤技术。

因为需要针对各种可能的情况设计转换标准，所以这种方法会增加算法的复杂度。当然这么做的好处是对各种推荐技术的优点和弱点比较灵敏，可以根据特定场景充分发挥不同推荐算法的优势。

推荐方法的组合

组合方案

3. 混合（Mix）

将多种不同的推荐算法混合在一起，其难点是如何进行结果的重排序。

4. 特征混合（Feature Combination）

将来自不同推荐数据源的特征组合起来，由另一种推荐技术采用。

这种方法一般会将协同过滤的信息作为增加的特征向量，然后在这增加的数据集上采用基于内容的推荐技术。

特征组合的混合方式使得系统不再仅仅考虑协同过滤的数据源，所以它降低了用户对物品评分数量的敏感度。相反，它允许系统拥有物品的内部相似信息，对协同系统是不透明的。

推荐方法的组合

组合方案

5. 级联型（Cascade）

首先用一种推荐技术产生一个较为粗略的候选结果，在此基础上使用第二种推荐技术对其作出进一步精确的推荐。

6. 特征递增（Feature Augmentation）

将前一个推荐方法的输出作为后一个推荐方法的输入。

它与级联型的不同之处在于，这种方法上一级产生的并不是直接的推荐结果，而是为下一级的推荐提供某些特征。

一个典型的例子是将聚类分析环节作为关联规则挖掘环节的预处理，从而将聚类所提供的类别特征用于关联规则挖掘。

推荐方法的组合

组合方案

7. 元层次混合（Meta-level Hybrid）

将不同的推荐模型在模型层面上进行深度的融合，而不仅仅是将一个输出结果作为另一个的输入。

例如，基于用户的方法和基于物品的方法一种可能的组合方式为：先计算目标物品的相似物品集，然后删掉所有其他（不相似的）物品，进而在目标物品的相似物品集上采用基于用户的协同过滤算法。

这种基于相似物品计算近邻用户的协同推荐方法，能很好地处理用户多兴趣下的个性化推荐问题，尤其是在候选推荐物品的内容属性相差很大的时候，该方法可以获得较好的性能。

目录

推荐系统的评价

1. 用户满意度

- 描述用户对推荐结果的满意程度。
- 通过用户问卷或者监测用户线上行为数据获得。

2. 预测准确率

- 描述推荐系统预测用户行为的能力。
- 通过离线数据集上算法给出的推荐列表和用户行为的重合率来计算。

3. 惊喜度

- 如果推荐结果和用户的历史兴趣不相似，但让用户很满意，则这是一个惊喜的推荐。
- 定性地通过推荐结果与用户历史兴趣的相似度和用户满意度来衡量。

4. 新颖性

- 如果用户没有听说过推荐列表中的大部分物品，则说明该推荐系统的新颖性较好。
- 通过推荐物品的平均流行度和用户问卷来获得。

推荐系统的评价

5. 覆盖率

- 描述推荐系统对物品长尾的发掘能力。（在推荐系统中，长尾效应指的是用户对热门物品的兴趣相对较高，而对于冷门物品的兴趣相对较低的现象。也就是说，少数热门物品占据了绝大部分用户的关注，而大量冷门物品却很难得到用户的关注。）
- 通过推荐物品占总物品的比例和所有物品被推荐的概率分布来计算。

6. 多样性

- 描述推荐系统中推荐结果能否覆盖用户不同的兴趣领域。
- 通过推荐列表中物品两两之间不相似性来计算。

谢谢！

Thanks for your attention!