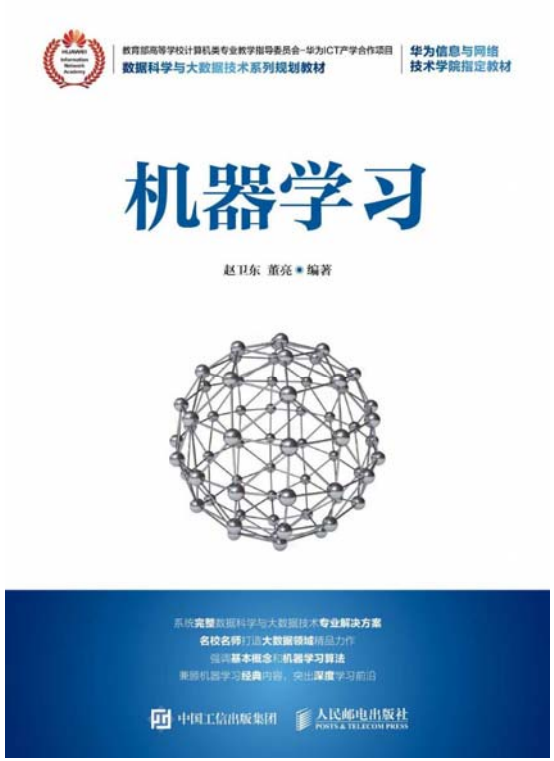


机器学习
电子推荐系统

复旦大学 **赵卫东** 博士

wdzhao@fudan.edu.cn

章节介绍

- 推荐系统根据用户的浏览记录、社交网络等信息进行个性化的计算，发现用户的兴趣，并应用推荐算法最终达到“千人千面”、“个性化”推荐的效果。本章首先结合推荐系统应用场景介绍推荐系统的通用模型，重点介绍基于内容的推荐、基于协同过滤的推荐、基于关联规则的推荐、推荐系统效果评测方法、推荐系统常见问题等，并结合实际案例介绍推荐系统设计和应用。

章节结构

- 推荐系统基础
  - 推荐系统的应用场景
  - 相似度计算
- 推荐系统通用模型
  - 推荐系统结构
  - 基于统计学的推荐
  - 基于内容的推荐
  - 基于协同过滤的推荐算法
  - 基于图的模型
  - 基于关联规则的推荐
  - 基于知识的推荐
  - 基于标签的推荐

章节结构

- 推荐系统评测
  - 评测方法
  - 评测指标
- 推荐系统常见问题

推荐系统基础

- 推荐系统是一种帮助用户快速发现有用信息的工具，通过分析用户的历史行为，研究用户偏好，对用户兴趣建模，从而主动给用户推荐能够满足他们感兴趣的信息。本质上，推荐系统是解决用户额外信息获取的问题。在海量冗余信息的情况下，用户容易迷失目标，推荐系统主动筛选信息，将基础数据与算法模型进行结合，帮助其确定目标，最终达到智能化推荐。推荐系统优点有：

  - 可提升用户体验。通过个性化推荐，帮助用户快速找到感兴趣的信息。
  - 提高产品销量。推荐系统帮助用户和产品建立精准连接，提高产品营销转化率。
  - 推荐系统可以挑战传统的2/8原则，使部热门的商品能够销售给特定人群。
  - 推荐系统是一种系统主动的行为，减少用户操作，主动帮助用户找到其感兴趣的内容。

推荐系统的应用场景

- 推荐系统的应用场景包括：
  - 电商平台。其中的“猜你喜欢”等部分、搜索结果中推荐商品排名靠前都是用了推荐系统。
  - 个性化电影网站。基于观看历史以及视频之间的联系分析用户兴趣，为用户做推荐。
  - 音乐歌单。基于用户收听历史、行为以及音乐风格等进行协同过滤推荐。
  - 社交网络。主要应用是好友推荐和资讯内容推荐。好友推荐是推荐有共同兴趣的用户成为好友，比如用户通过阅读、点赞、评论了相同的博文产生关系。便可以推荐互加好友。
  - 新闻网站。应用推荐方便用户获取个性化资讯，减少用户浏览、检索新闻的时间，增加用户粘性。
  - 个性化阅读。为每一个用户定制其感兴趣的个性化内容。获得用户兴趣，推送个性化的阅读内容，提供更优的阅读方式和更好的阅读体验。
  - 个性化广告。有针对性地向特定用户展示特定广告内容。对广告受众进行用户画像，基于用户行为做协同过滤，根据用户对广告的态度或反应改进推荐算法，减少用户对广告的负面体验。

相似度计算

- 在推荐系统中，涉及用户之间相似度、物品之间的相似度和用户与物品之间的相关性的计算。其中相似度计算是基于向量间距离，距离越近相似度越大。例如，在用户对物品偏好的二维矩阵中，一个用户对所有物品的偏好作为一个向量，可用于计算用户之间的相似度，即两个向量间的距离；将所有用户对一个物品的偏好作为表示此物品，可以用于计算物品之间的相似度。

皮尔逊相关系数

- 皮尔逊相关系数（Pearson Correlation Coefficient）一般用于计算两个变量间的相关性，它的取值是[-1,1]，当取值大于0时表示两个变量是正相关的；当取值小于0时表示两个变量是负相关的，取值为0表示不相关。在推荐系统中，常用于用户之间的相似度计算，计算公式如下：

$$sim(x,y) = \frac{\sum_{i=1}^{n}(x_i - \bar{x})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n}(x_i - \bar{x})^2}\sqrt{\sum_{i=1}^{n}(y_i - \bar{y})^2}}$$

- 其中，n为两个用户x、y共同评价过物品的总数；x_i表示用户x对物品i的评分，$\bar{x}$表示用户x所有评价过的物品的平均分；y_i表示用户y对物品i的评分，$\bar{y}$表示用户y所有评价过的物品的平均分；

欧几里德相似度

- 用于计算欧几里德空间中两个点的距离，以两个用户x和y为例子，看成是n维空间的两个向量x和y，x_i表示用户x对物品i的喜好值，y_i表示用户y对物品i的喜好值，他们之间的欧几里德距离（Euclidean Distance）计算公式如下：

$$d(x,y) = \sqrt{\sum_{i=1}^{n}(x_i - y_i)^2}$$

- 对应的欧几里德相似度，一般采用以下公式进行转换：

$$sim(x,y) = \frac{1}{1 + d(x,y)}$$

- 表示距离越小，相似度越大。在Mahout的Taste中，计算用户之间和物品之间的欧几里德相似度的类是EuclideanDistanceSimilarity。

余弦向量相似度

- 余弦向量相似度（Cosine Similarity）是计算两个向量的夹角余弦，被广泛应用于计算文档之间的相似度，其计算公式为如下：

$$sim(A,B) = \cos(\theta) = \frac{A \cdot B}{\|A\|\|B\|} = \frac{\sum_{i=1}^{n} A_i \times B_i}{\sqrt{\sum_{i=1}^{n}(A_i)^2} \times \sqrt{\sum_{i=1}^{n}(B_i)^2}}$$

- 其中，A_i和B_i分别表示两个文档的向量分量。在Mahout的Taste中，实现Cosine相似度的类是PearsonCorrelationSimilarity，此外一个类UncenteredCosineSimilarity实现了形式化以后的cosine向量夹角：

$$sim(x,y) = \frac{\sum_1^n x_i y_i - \bar{y}\sum_1^n x_i}{\sqrt{\sum_1^n {x_i}^2 - \bar{x}\sum_1^n x_i}\sqrt{\sum_1^n {y_i}^2 - \bar{y}\sum_1^n y_i}}$$

余弦向量相似度

- 余弦相似度更多的是从方向上区分差异，而对最后的结果数值不敏感，所以无法度量每个维数值的差异，在某些情况下会导致无法区分用户的评分。例如用户对内容进行评分，按照5分制进行打分，1分最差，5分最好，A和B两用户分别对两个物品进行评分，分值分别为(1,2)和(4,5)，使用余弦相似度得出的结果是0.98，两者相似度较高，但实际上A用户不喜欢这2个内容，而B用户比较喜欢，这说明结果产生了误差，调整余弦相似度是所有维度上的数值都减去均值，再用余弦相似度计算，例如A和B对两个物品的评分均值都是3，那么调整后为(-2,-1)和(1,2)得到相似度结果为-0.8，相似度为负值并且差异较大，这样更加符合事实。

曼哈顿相似度

- 曼哈顿距离在Mahout的Taste里的实现类是CityBlockSimilarity，采用了简化的计算方式，比欧式距离计算量少，性能相对高。比较适合用户的偏好数据是0或者1的情况。

对数似然相似度

- 对数似然相似度在Mahout的Taste中实现类名为LogLikelihoodSimilarity，比较适用于用户的偏好数据是0或者1的情况。

斯皮尔曼相似度

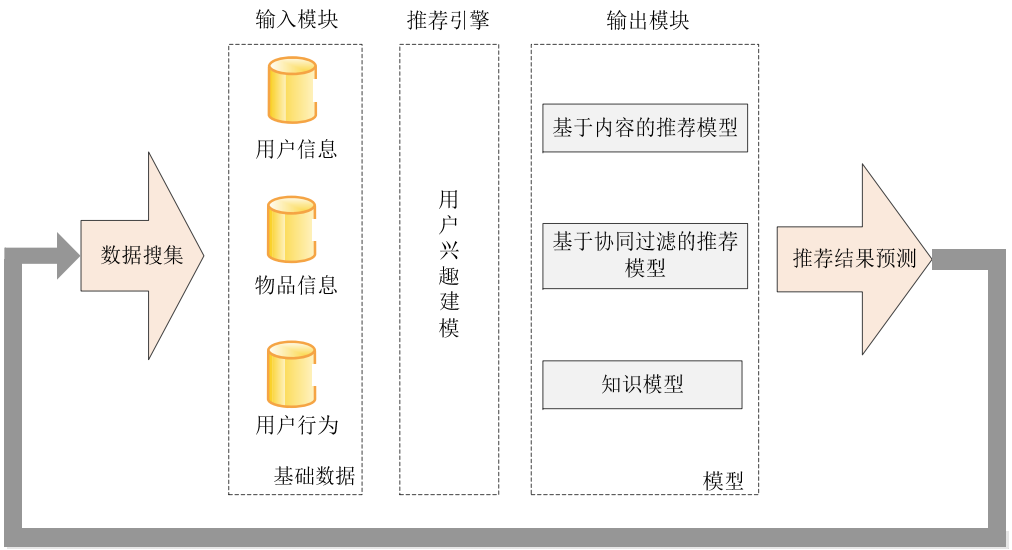
- 斯皮尔曼相似度在Mahout的Taste中实现类名为SpearmanCorrelationSimilarity，它舍弃了真实的评分值，将其转化为排序值，可以理解为是排列后用户喜好值之间的Pearson相关系统。例如，对于每个用户，重写其最不喜欢的评分值为1，次不喜欢的评分值为2，依此类推。对转换后的值求Pearson的相关系数，得到的结果就是斯皮尔曼相关系数。因为斯皮尔曼相关性计算需要花费时间对喜好值进行排序，效率并不高，所以一般用于学术研究或者小规模计算。

推荐系统通用模型

- 推荐系统应用较广，不同的业务场景用到的数据、算法和模型都不同。如果针对每个场景都从头开发，将会耗费较多时间和人力。推荐算法进行通用化设计，可以更好地将一类推荐算法复用到不同的推荐场景中，从而支持多种业务领域。

推荐系统结构

- 推荐系统有3个重要的模块，包括输入模块、推荐算法模块（推荐引擎）、推荐输出模块。推荐系统结构如下图所示。

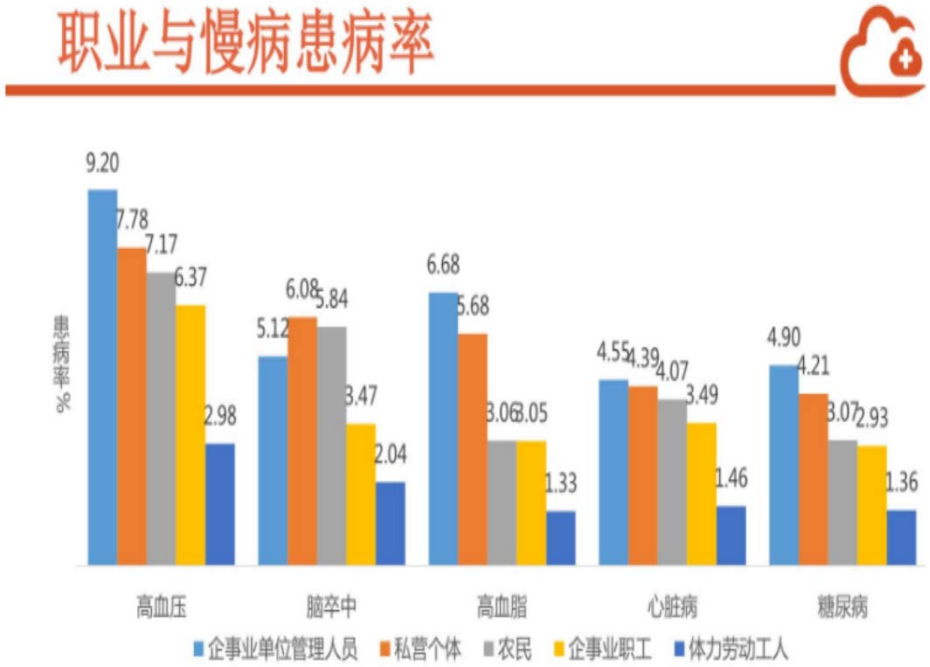


推荐系统结构

- 推荐系统首先通过分析用户行为数据，建立用户偏好模型。然后使用用户兴趣匹配物品的特征信息，再经过推荐算法进行筛选过滤，找到用户可能感兴趣的推荐对象，最后推荐给用户。上述过程经过训练和验证最终形成推荐模型，可用于在线或离线推荐。同时，推荐结果在用户端的响应也作为输入数据，用于模型的迭代优化。

基于人口统计学的推荐

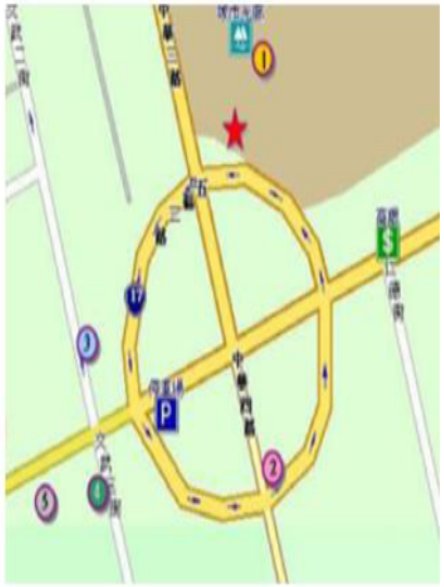
- 基于人口统计学的推荐机制根据用户的人口统计学信息发现用户间的相关程度向用户推荐与之相似的用户感兴趣的物品。基于人口统计学信息对用户画像，根据画像计算用户间相似度，形成用户群体，按照用户群的喜好推荐给当前用户一些物品。这种方法的好处是没有“冷启动”问题。不依赖于物品本身的特征，所以不同物品领域都可以使用，具有领域独立性。问题在于由于人口统计学信息未必准确，所以计算结果可信度较低，难以令人信服。



基于内容的推荐

- 基于内容推荐的原理是根据用户感兴趣的物品A，找到和A内容信息相近的物品B。提取用户偏好的物品特征是基于内容推荐算法的关键，基于内容的推荐过程是用户喜欢的物品和特征的描述，物品的特征有属性、描述等，图书的特征是一些文本内容，特征提取可能涉及文本处理相关技术，将文本内容转化为可计算的向量形式，实现对物品的特征建模，应用推荐算法进行内容推荐。除此之外，还有相似度计算。基于内容的推荐优点是简单有效，推荐结果直观，容易理解，不需要领域知识。不需要用户的历史行为数据，比如对物品的评价等。

基于位置感知的移动购物推荐



符合您需求的商店列表（前五名）：

Rank	店名
1	商务露天咖啡
2	麦富热一五福二店
3	德州炸鸡一五福分店
4	太立伊热丹
5	普吉梦都

注："↑"表示您目前的所在位置

首先建立用户的模型。通过分析用户的访问历史，可以用词汇来刻画用户的兴趣爱好。假设用户访问的网页集合为S，其兴趣模型表示为：

$$CP(c) = \sum_{w \in S} Vector(w)$$

式中Vector(w) 表示网页w的术语向量。

商户会向系统提供相关的网页，可以从这些网页建立商户的模型。通过向量的余弦相似性判断客户和商户之间匹配程度：

$$Similarity(c, w) = Cosin\left(CP(c), Vector(w)\right)$$

基于位置的推荐服务还要考虑商户与客户之间的位置远近关系：

$$DistanceDecay(c, w) = \frac{1}{e^{\lambda \times Distance(c, w)}}$$

式中λ是[0, ∞]范围内的一个参数，表示用户对位置敏感程度。Distance(c, w)表示顾客c和网页w代表的商户之前的欧几里德距离。

一个客户在他当前位置时对网页w代表的商户兴趣程度描述为：

$$Interest(c, w) = Similarity(c, w) * DistanceDecay(c, w)$$

基于协同过滤的推荐算法

- 基于用户行为数据设计的推荐算法，称为协同过滤算法。此方法主要根据用户的历史行为，寻找用户或物品的临近集合，以此计算用户对物品的偏好，包括基于领域、图、关联规则、知识的推荐算法，其中最广泛应用的是基于领域的方法，在实践中往往是上述几种方法的混合应用。

基于领域的推荐算法

- 基于领域的推荐算法主要包括两种算法：基于用户的协同过滤算法和基于物品的协同过滤算法，基于物品的协同过滤与计算用户兴趣相似度一致，基于物品的协同推荐需要计算与用户偏好的物品相似的物品。

基于用户的协同过滤算法

- 基于用户的协同过滤算法为用户推荐兴趣相似的其他用户喜欢的物品。算法的关键是计算两个用户的兴趣相似度。计算用户相似度的方法有3种：余弦相似性、皮尔森系数相关和修正的余弦相似性。算法步骤如下：
  - 找到与目标用户兴趣相似的用户集合
  - 找到这个集合中的用户喜欢的，且目标用户没有用过的物品，推荐给目标用户

	The Matrix	Titanic	Die Hard	Forrest Gump	Wall-E
John	5	1		2	2
Lucy	1	5	2	5	5
Eric	2	?	3	5	4
Diane	4	3	5	3	

预测Eric对电影Titanic的评分：

$$\hat{r}_{ui} = \frac{1}{|\mathcal{N}_i(u)|} \sum_{v \in \mathcal{N}_i(u)} r_{vi}$$

$$\hat{r}_{ui} = \frac{\sum_{v \in \mathcal{N}_i(u)} w_{uv} r_{vi}}{\sum_{v \in \mathcal{N}_i(u)} |w_{uv}|}.$$

基于用户协同过滤推荐示例

- 下表是基于用户协同推荐过程，可以看到用户A与用户C所喜欢的物品具有较多的交集，即两个用户具有相似性，那么用户C喜欢的物品很有可能用户A也会喜欢，而用户C喜欢物品D，则可以向用户A推荐物品D。

用户/物品	物品A	物品B	物品C	物品D
用户A	√		√	推荐
用户B		√		
用户C	√		√	√

- 计算用户兴趣相似度时，要避免热门物品自带马太效应的影响，即大部分用户可能都对热门的物品表现出喜欢的状况，但是这些用户之间并非一类人，因为所谓的热门物品区分度较弱。

基于用户协同过滤推荐示例

- 基于用户的协同过滤算法缺点是随着用户数目的增大，计算用户兴趣相似度越来越复杂，时间和空间复杂度与用户数接近于平方关系。所以一般采用离线方式推荐，当用户产生新行为时，不会立即进行计算，推荐结果不会立即变化。此外，这一算法是基于隐式群体的兴趣进行推荐，可解释性不强。这一算法适用于用户兴趣稳定且不明显的场景，即通过群体的兴趣来代表用户个体的兴趣，一旦群体的兴趣确立，就可以认为个体用户服从此兴趣，由此向其进行推荐，结果一般较准确。

基于物品的协同过滤算法

- 基于物品的协同过滤算法是给用户推荐跟他喜欢的物品相似的物品，是基础的推荐算法，集成在各类电商平台的推荐系统中。与基于内容的推荐算法相比，是通过用户的行为计算物品之间的相似度，而基于内容的推荐算法计算的是物品内容特征的相似度。例如，物品A、B有很大的相似度是因为喜欢物品A的用户也都喜欢物品B。

基于隐语义模型算法

- 隐语义模型最早出现在文本挖掘领域，用于找到文本的隐含语义，核心思想是通过隐含特征关联用户兴趣和物品，通过矩阵分解建立用户和隐类之间的关系、物品和隐类之间的关系，最终得到用户对物品的偏好关系，即对于某个用户，首先得到他的兴趣分类，然后从分类中挑选出他可能喜欢的物品。隐语义模型使用算法自动得到物品和用户的分类权数。不仅准确度更高，可以得到可靠的权重，还减少了标记物品所需要的人力。首先通过隐语义分析给物品分类，并计算出物品属于每个类的权重；然后，确定用户对哪类物品感兴趣，以及感兴趣程度。对于一个给定的分类，选择那些属于这个类的物品推荐给用户，以及确定这些物品在一个类中的权重。

基于隐语义模型算法

- 推荐系统用户的行为分为显性反馈和隐形反馈，隐语义模型在显性反馈数据上解决评分预测问题达到了很好的精度。对于隐形反馈数据是指数据集只有正样本，即用户喜欢什么物品，没有用户不感兴趣的样本。

$$Preference(u,i) = r_{ui} = p_u^T q_i = \sum_{k=1}^{n} p_{u,k} q_{i,k}$$

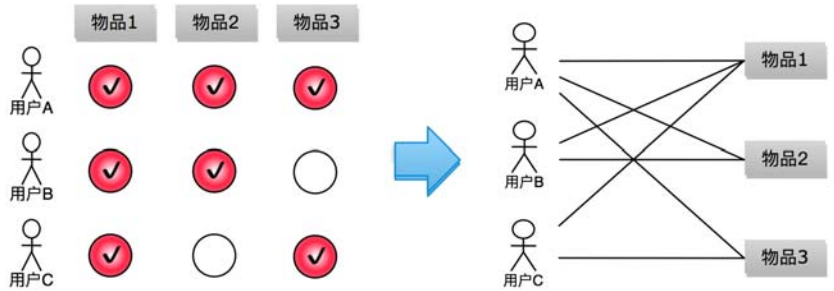
- 其中，k为隐类，$p_{u,k}$为用户u与隐类k之间的关系，$q_{i,k}$为物品i与隐类k之间的关系，其值越高越能代表隐类k。两者相乘为该用户与该物品之间的权重，即用户对物品的喜好程度。

基于图的模型

- 用户行为很容易用二分图表示，因此很多图的算法都可以用到推荐系统中，其中物品作为图中节点，节点之间连线是用户行为中共同购买或浏览，物品之间的相似性可以通过计算图中节点之间的强度来实现。

用户行为数据的二分图

- 基于图的模型基本思想是将用户行为数据表示为二分图。每个二元祖（u，i）代表用户u对i曾产生操作，这样便可以将这个数据表示为二分图。下图是一个简单的用户物品二分图模型，方形节点表示物品，用户节点和方形节点之间的边代表用户对物品的行为。下图可以说明A对1、2、3曾有操作。



PageRank算法

- PageRank 通过网页之间的链接关系计算网页权重，权重高的网页特点是：链接向它的网页数量多、链向它的网页权重也高。PageRank 通过这样的连接关系，一轮轮迭代计算后得出各网页的权重。算法迭代的公式如下：

$$\mathrm{PR}(page_i) = \frac{1-d}{N} + d\sum_{p_j \in M(p_i)} \frac{PR(page_j)}{L(page_j)}$$

- 上式中$\mathrm{PR}(page_i)$是网页$page_i$的访问概率，用来表示网页的重要度，d是用户继续访问网页的概率，d的值一般设为0.85，N是网页总数,$M(page_i)$表示指向网页$page_i$的网页集合，$L(page_j)$表示网页$page_j$指向别的网页的链接数量。从中可以看到某一网页$page_i$的重要性主要是所有指向它的网页来决定的。指向它的网页中重要性越高，且数量越多，网页$page_i$的重要性就越高。所以，也可以将这种思想应用到推荐系统中，用于用户与物品之间的相似度计算。

基于PageRank的推荐

- 对于推荐系统，需要计算的是物品节点相对于某一个用户节点u的相关性，PageRank算法能够为用户对所有物品进行排序。基于从不同点开始的概率不同，算法的执行过程如下：
  - 假设要给用户u进行个性化推荐，从其对应的节点开始在用户物品二分图上进行随机游走。
  - 又走到任何一个节点时，计算节点的访问概率，尤其决定是否继续游走。

基于关联规则的推荐

- 关联规则是反映物品与其他物品之间的关联性，常用于实体商店或在线电商的推荐系统：通过对顾客的购买记录数据进行关联规则挖掘，发现顾客群体的购买习惯的内在共性。早期的关联分析主要用于零售行业的购物行为分析，所以也称之为购物篮分析。需要注意的是关联关系并不意味着存在因果关系。关联规则分析中的关键概念包括支持度、置信度、提升度。在关联分析算法中，常见的有Apriori和FP增长算法。

支持度

- 支持度是指两件商品A和B在总销售笔数（N）中同时出现的概率，即A与B同时被购买的概率，计算公式如下：

$$Support(A \cap B) = \frac{\text{Freq}(A \cap B)}{N}$$

- 使用支持度的目标是找到在一次购物中一起被购买的两个商品，从而可以提高推荐的转换率。在使用支持度时需要结合业务特点确定一个最小值，只有高于此值的商品项集才能进行推荐，即关注出现频次高的商品组合，超过某一支持度最小值的项集称为频繁项集。

置信度

- 置信度是购买A商品后再购买B商品的条件概率，置信度大说明购买A的客户很大概率也会购买B。计算公式如下：

$$Confidence(A \rightarrow B) = \frac{Freq(A \cap B)}{Freq(\mathrm{A})} = \frac{Support(A \cap B)}{Support(\mathrm{A})}$$

- 例如，电商网站10月份订单中面包售出40万笔，一次购买面包牛奶的有30万笔，置信度75%，一次购买面包薯片的有10万笔，置信度25%，则发现用户购买面包后会向其推荐牛奶，或者使用组合搭配销售。

提升度

- 提升度用来判断规则是否有实际价值，描述的是对比不使用规则，使用规则可以提高多少。使用规则商品在购物车中出现的次数是否高于商品单独出现在购物车中的概率。大于1说明有效，小于1则无效。计算公式如下：

$$Lift(A \rightarrow B) = \frac{\text{Support}(A \cap B)}{Support(A) * Support(B)} \quad (1)$$

$$\text{Lift}(\text{A} \rightarrow \text{B}) = \frac{\text{Confidence}(A \rightarrow B)}{Support(B)} \quad (2)$$

- 例如，电商网站10月份有100万笔订单，购买面包30万笔，牛奶40万笔，同时购买两者的20万笔，面包、牛奶、面包和牛奶支持率依次为30%、40%、20%，所以提升度为1.667，大于1，所以牛奶面包规则是有提升效果的。

关联规则提取过程

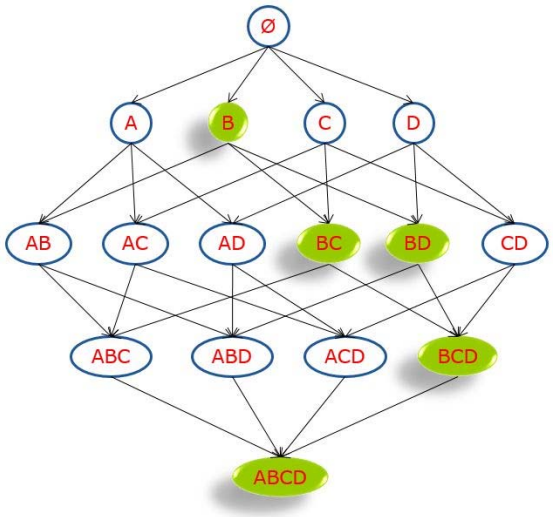
- 关联规则提取过程是找出所有支持度>=最小支持度、置信度>=最小置信度的关联规则。可以通过穷举项集的所有组合方式找出所需要的规则，但是时间复杂度较高，难以接受。需要找到快速挖出满足条件的关联规则的方法。关联规则挖掘分两步进行：首先生成频繁项集，即找出所有满足最小支持度的项集。然后生成规则，在频繁项集的基础上生成满足最小置信度的规则，产生的规则称为强规则。关联规则挖掘所花费时间主要消耗在生成频繁项集上，利用频繁项集生成规则时间复杂度不高，关键在于优化频繁项集的生成。

Apriori算法

- Apriori算法是最有影响力的基于关联规则频繁项集挖掘算法，算法分为两个步骤：第一步是通过迭代检索出所有事务中的频繁项集，即支持度不低于用户设定的阈值的项集。第二步利用频繁项集构造出满足用户最小置信度的规则。其中找出所有频繁项集是基于该算法的性质：
  - 频繁项集的子集也是频繁项集
  - 非频繁项集的超集一定是非频繁的

Apriori算法

- 具体的过程是先扫描所有订单记录，统计每个商品的频次和商品项集合，并计算每一个商品的支持度，将低于阈值的单项商品移除。如右图中B为非频繁项集，则将移除，基于Apriori的性质2，可以对B的分支进行剪枝，BC、BD、BCD、ABCD都被移除。然后对商品项集进行组合，形成二项商品集合，第二次扫描订单记录，计算每个二项商品集的支持度，将低于阈值的二项商品移除，依此类推，商品项集合无法继续进行组合为止，将所有频繁项集合进行集合连接，通过扫描订单记录剪枝，移除含非频繁项集的组合项，剩下的就是最小信任度的规则集合。



基于知识的推荐

- 基于知识的推荐主要应用于知识型的产品中，在某种程度是一种推理技术，基于效用知识实现对某一特定用户推荐特定项目，因此推荐结果具有较强的可解释性。在一些涉及知识级别的场景中，需要用到基于知识的推荐，推荐系统依据用户目前所处的知识级别，同时根据所有的知识级别进行分析，为用户推荐合适进阶的信息。综合用户知识和产品知识，通过推理什么产品能满足用户需求来产生推荐。这种推荐系统不依赖于用户评分等关于用户偏好的历史数据，故其不存在冷启动方面的问题，可以响应用户的即时需求，当用户偏好发生变化时不需要任何训练。

基于知识的推荐案例

- 学生面对海量习题带来的信息过载时，容易出现学习过程中针对性不强、效率不高等问题，此时可基于知识点层次图进行个性化习题推荐。首先，借鉴课程知识点体系结构的特点，构建表征知识点层次关系的权重图，用以有效反映知识点间的层次关系。然后，根据学生对知识点的掌握情况在知识点层次图的基础上进行个性化习题推荐。通过更新学生与知识点对应的失分率矩阵，获取学生掌握薄弱的知识点，以此实现习题推荐。

基于实例的推荐

- 根据层次关系图的构建方法，基于知识推荐可以划分为基于约束推荐和基于实例推荐。早期的基于实例的推荐采用的是基于查询的方法，由用户指定需求，通过目录检索或搜索发现目标物品，用户对当前浏览的物品进行评价，然后基于其评价的结果进行导航，这是基于实例推荐系统的关键。评价的基本思想是用户以当前物品为满足的目标来指明他们的要求，推荐的过程就是商品筛选过滤的过程。

基于约束的推荐

- 基于约束的推荐系统强调推荐时的约束规则，基于约束的推荐是利用预先定义的推荐知识库显示地定义约束，把推荐任务看作是解决一个约束满足问题的过程，满足约束的候选项就推荐给用户。基于约束的推荐方法通常被用来为那些不经常被购买的产品领域构建推荐系统，而且产品非常复杂，很多顾客不能详细地了解其所有的技术特征，特别是在专业设备、金融服务或更复杂的产品等领域，其中基于约束的推荐系统一般会涉及用户属性、产品属性、过滤条件、物品约束条件、合取查询。

基于标签的推荐

- 标签是一种可以用来描述信息的关键词，可以作为物品的元信息来描述物品的特征，也可以用于标识用户的喜好。基于标签的推荐算法是通过统计每个用户最常用的标签，统计时往往计算权重值对标签进行排序，不是简单的使用出现次数，可加入时间因子等。对于每个标签，统计打过这个标签次数最多的物品列表，这样对于一个用户，就可以依据其常用标签找到对应的热门物品推荐给他。

标签生成算法优化

- 用户浏览热门标签对应的内容不能代表用户个性化的兴趣，需要应用标签权重来实现标签排序和选择，对于时间久的标签需要降低权值，最新的标签更能说明用户兴趣所在。一般情况下可以结合标签的出现次数和最后标记时间，假设标签i出现的次数为n，上一次访问此标签的时间为s秒，可以通过公式W=n/s来计算i对应的权重值。对于新用户或新物品，标签数量可能过少，需要对标签进行扩展，找到相似的标签；此外，还可以通过构建语料库对标签之间的共现次数进行统计，得到标签之间的概率相关性，构建标签相似性矩阵，或者通过第三方知识库构建向量空间模型，可快速对用户兴趣标签进行扩展。

标签清理

- 普通用户给物品所打的标签往往随意性较大，标签质量不稳定，可以结合信息熵对用户生成标签进行验证，判断用户生成标签的稳定程度，有针对性地过滤掉噪声标签；此外，由于不同用户生成的同一意义的标签可能会有多个，需要对标签进行相似度计算，清理掉同义词，使标签更加集中，有利于优化推荐结果；在使用LDA等主题提取算法时，由于部分算法是依据词频等因素提取主题关键词，容易提取到一些无意义的词汇，需要将无义词通过停用词来进行删除，方便做出推荐解释。此外，在中文主题提取中涉及中文分词，分词模块质量会影响生成的标签质量。

推荐系统评测

- 增加评测的目的是确定算法在什么情况下性能最好，一般评测维度分为用户维度、物品维度、时间维度。其中用户维度主要包括用户的人口统计学信息、活跃度以及是不是新用户等；物品维度包括物品的属性信息，流行度、平均分以及是不是新加入的物品等；时间维度包括季节、工作日还是周末、白天还是晚上等；不同纬度下的系统评测指标，能全面了解推荐系统性能。

评测方法

- 获得评测指标的实验方法，通常分为离线实验、用户调查、在线实验。一般说来，一个新的推荐算法最终上线，需要完成上述的三个实验。首先，通过离线实验证明它在很多离线指标上优于现有的算法；其次，通过用户调查确定用户满意度不低于现有的算法；最后，通过在线A/B测试确定关键指标上优于现有的算法。

离线实验

- 离线实验方法步骤如下：
  - 通过日志系统获得用户行为数据，并按照一定格式生成一个标准的数据集
  - 将数据集按照一定的规则分成训练集和测试集
  - 在训练集上训练用户兴趣模型，在训练集上进行预测
  - 通过事先定义的离线指标，评测算法在测试集上的预测结果

用户调查

- 用户调查需要一些真实的用户，让他们在需要测试的推荐系统上完成一些任务。在他们完成任务时，观察和记录用户的行为，并让他们回答一些问题。最后，分析他们的行为和答案，了解测试系统的性能。用户调查的优点是可以获得用户主观感受的指标，出错后容易弥补。缺点是招募测试用户代价较大，无法组织大规模的测试用户。

在线实验

- 在完成离线实验和用户调查之后，可以将系统上线做A/B测试，将它和旧算法进行比较。在线实验最常用的评测方法是A/B测试，通过一定的规则将用户随机分成几组，对不同组的用户采用不同的算法，然后通过统计不同组的评测指标，比较不同算法的好坏。A/B测试的核心思想是多个方案并行测试，每个方案只有一个变量不同，以某种规则优胜劣汰。在A/B测试中必须是单变量，A/B测试的优点是可以公平获得不同算法实际在线时的性能指标，包括商业上关注的指标。缺点是周期较长，必须进行长期的实验才能得到可靠的结果。

评测指标

- 评测指标用于评测推荐系统的性能，有些可以定量计算，有些只能定性描述。从经验上看，对于可以离线优化的指标，在给定覆盖率、多样性、新颖性等限制条件下，应尽量优化预测准确度。

用户满意度

- 用户满意度是评测推荐系统的重要指标，无法离线计算，只能通过用户调查或者在线实验获得。调查问卷，需要考虑到用户各方面的感受，用户才能针对问题给出准确的回答。在线系统中，用户满意度通过统计用户行为得到。例如用户如果购买了推荐的商品，就表示他们在一定程度上满意，可以用购买率度量用户满意度。一般情况，可以用用户点击率、停留时间、转化率等指标度量用户的满意度。

预测准确度

- 预测准确度度量的是推荐系统预测用户行为的能力，是推荐系统最重要的离线评测指标。准确度的指标可以分为预测评分准确度以及TopN推荐。

预测评分的准确度

- 预测评分的准确度指的是算法预测的评分与用户实际评分的贴近程度。准确度指标一般通过平均绝对误差、均方根误差实现。平均绝对误差因计算简单、通俗易懂得到广泛应用。但有局限性，因为对平均绝对误差贡献大的往往是很难预测准确的低分商品，计算公式如下：

$$\mathrm{MAE} = \frac{1}{|E^P|} \sum_{(u,a)\in E^P} |r_{ua} - r'_{ua}|$$

- 其中r_{ua}表示用户u对商品a的真实评分，r'_{ua}表示预测评分，E^P表示测试集。即便推荐系统A的MAE值低于系统B，很可能只是由于系统A更擅长预测这部分商品的评分，即系统A比系统B能更好地区分用户非常讨厌和一般讨厌的商品，显然这样区分的意义不大。

预测评分的准确度

- 均方根误差加大了对预测不准的用户物品评分的惩罚（平方项的惩罚），因而对系统的评测更加苛刻。如果评分系统是基于整数建立的（即用户给的评分都是整数），那么对预测结果取整数会降低平均绝对误差的误差。公式如下：

$$\mathrm{RMSE} = \sqrt{\frac{1}{|E^P|}\sum_{(u,a)\in E^P}(r_{ua} - r'_{ua})^2}$$

TopN推荐

- **TopN推荐**是指提供推荐服务时，一般给用户的是个性化的推荐列表，这种推荐叫做TopN推荐。TopN推荐的预测准确率，一般通过准确率（precision）和召回率（recall）两个指标度量，其计算公式如下：

$$Precision = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |R(u)|}$$

$$Recall = \frac{\sum_{u \in U} |R(u) \cap T(u)|}{\sum_{u \in U} |T(u)|}$$

- $R(u)$是根据用户在训练集上的行为给用户做出的推荐列表，$T(u)$是用户在测试集上的行为列表。TopN推荐更符合实际的应用需求，例如预测用户是否会看一部电影，比预测用户看了电影之后会给它什么评分更重要。

覆盖率

- 覆盖率（coverage）是描述一个推荐系统对物品长尾的发掘能力，用系统推荐的物品占总物品的比例来衡量。假设系统的用户集合为U，物品集合为I，推荐系统给每个用户推荐一个长度为N的物品列表$R(u)$，覆盖率公式为：

$$Coverage = \frac{|\cup_{u \in U} R(u)|}{|I|}$$

- 其中$|I|$表示物品列表的数量，覆盖率是内容提供者关心的指标，覆盖率为100%的推荐系统可以将每个物品都推荐给至少一个用户，而覆盖率只有10%，意味着只有很小一部分物品会推荐出来，推荐的内容过于狭窄。除了推荐物品的占比，还可以通过研究物品在推荐列表中出现的次数分布，更好地描述推荐系统的挖掘长尾的能力。如果分布比较平，说明推荐系统的覆盖率很高；如果分布陡峭，说明分布系统的覆盖率较低。

多样性

- 为了满足用户广泛的兴趣，推荐列表需要能够覆盖用户不同兴趣的领域，需要具有多样性。多样性描述了推荐列表中物品两两之间的不相似性，多样性的前提是用户的隐含兴趣是多样的，即用户当前的行为只是其兴趣的一部分，推荐系统很难将用户的所有真实兴趣提取出来，所以就要在推荐结果中加入一些与用户兴趣看起来不相符的内容，一方面减少审美疲劳，防止进入恶性循环推荐中。例如，给用户推荐内容单一，用户点击此类内容必然更多，后面推荐时更加推荐此类内容，另一方面，可以对用户潜在兴趣进行验证，收集用户的内容喜好。

新颖性

- 新颖性是影响用户体验的重要指标之一。它指向用户推荐非热门非流行物品的能力。评测新颖度最简单的方法，是利用推荐结果的平均流行度，因为越不热门的物品，越可能让用户觉得新颖。此计算比较粗糙，需要配合用户调查准确统计新颖度。

惊喜度

- 推荐结果和用户的历史兴趣不相似，但却让用户满意，这就是惊喜度很高。目前惊喜度还没有公认的指标来定义，主要靠用户的反馈和后续行为来验证。

信任度

- 如果用户信任推荐系统，就会增加用户和推荐系统的交互。增加系统透明度可以提高系统的信任度，提供推荐解释，让用户了解推荐系统的运行机制。或者利用社交网络，通过好友信息给用户推荐，由好友进行推荐解释。度量信任度的方式可以通过问卷调查，也可以通过对用户行为的不断累积进行分析，例如，用户对推荐的结果进行了阅读、购买或分享较多，说明其对推荐系统比较认可；相反，如果用户对推荐结果的后续行为中很少有正向习惯，则从侧面说明用户对于推荐系统的结果不认可。

实时性

- 推荐系统的实时性包括实时更新推荐列表满足用户新的行为变化，以及将新加入系统的物品推荐给用户。此外，由于大部分应用推荐系统的平台中往往物品数量和用户数量均较多，对系统的实时计算带来较大压力，如果系统设计时未考虑实时更新推荐结果的要求，很有可能会随着数据量的不断增加，推荐时间会不断增长，要么在硬件上进行大量投入，要么对推荐算法进行优化改进。实时性更多是对推荐系统的架构方面的评估。

健壮性

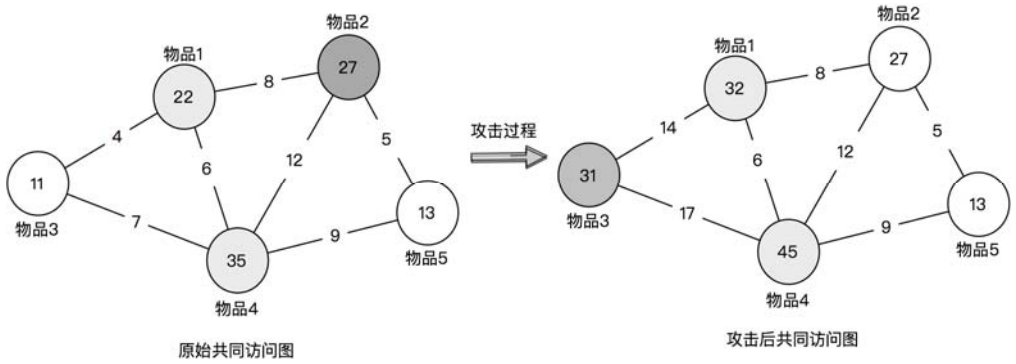
- 推荐系统通过共现的相关度来描述物品对于用户偏好的表现，从而实现物品推荐，但是物品共现并不代表偏好一致，如果通过攻击共现访问导致共现与用户偏好之间有偏差，就达到了攻击目的。攻击分为“推攻击”与“核攻击”，前者是使目标物品的推荐频率明显高于其他物品，从而实现该物品被更多地推荐给用户；后者是使目标物品的推荐频率明显低于其他物品，从而实现该物品尽可能不被系统推荐。

健壮性

- 推荐系统可能遭受到的攻击类型有：
  - 随机攻击。是对特定商品评最高分或最低分，对其他商品随机评分或给平均分，由于评分成本较高，这种攻击方法效果一般。
  - 蹭热销攻击。将目标商品与热销商品绑定在一起，随着热销商品推荐。
  - 反热销攻击。将目标商品与系统中不受欢迎的物品绑定在一起。在这种情况下，系统不容易推荐目标商品。
  - 大众化攻击。首先将目标商品评为最高分（如果是竞品则评为最低分），其它商品则根据商品的得分是否高于所有商品的平均分来给分，使得攻击者给出来的分值更加大众化，不容易被发现。
  - 探测攻击策略。首先伪造一个用户，系统就会给用户推荐一些商品，根据这些推荐商品的情况，探测相似用户选择商品的情况。然后依据获得的信息对他们选择的商品进行攻击。

基于共同访问图的攻击

- 基于共同访问的物品推荐系统是从用户和物品的角度出发，当发现用户喜欢物品i时，推荐系统会向用户推荐与物品i相似的N个其他物品，攻击者可以通过对物品之间相似度计算方法中人为注入攻击，如果物品间相似度计算是通过用户共同访问来实现的，那么，通过调整目标物品与锚点物品之间的相似度来达到攻击的目的，原理如下图所示。



基于共同访问图的攻击

- 攻击过程中，提升物品间的相似度是通过伪造用户实现的，把锚点物品与目标商品间的关联通过多个伪造用户行为进行增强，提高物品1、4、3的共现次数，三者间的相似度就会增加。具体步骤如下：
  - 为物品3选择锚点物品，因为物品1和4对应的推荐列表中不包含3，但都与3相连，所以选择1和4作为3的锚点物品。
  - 攻击者通过各种方法不断共同访问或购买锚点物品和目标物品，提高其共现度，本例中1、4与3的共现次数分别提升至14和17。
  - 根据生成推荐列表的计算方法，计算1和4的推荐列表中均含有3，从而实现攻击。

推荐系统常见问题

- 推荐系统实际应用中往往会遇到一些问题，例如用户数据很少，或用户行为较少的冷启动问题。冷启动问题有：系统冷启动、物品冷启动、用户冷启动。对于系统冷启动，先建立物品相关度，一旦用户展现出对物品的兴趣，即可推荐相关的物品。对于新上线的物品，利用物品内容相似性，推荐给喜欢类似物品的用户。物品冷启动对于时效性较强的网站非常重要，因为物品的价值会因为时间的推移而降低。针对用户冷启动，提供非个性化推荐，比如热门排行。积累一定数据之后再进行推荐，或者利用用户注册信息，利用用户社交网络账号，导入用户好友，推荐好友喜欢的物品。

利用上下文的信息

- 用户所处的上下文，包括用户访问推荐系统的时间、地点、心情等，用户兴趣时随着时间变化的，推荐算法需要平衡用户的近期行为与长期行为，使推荐列表既能反应用户近期行为表现的兴趣变化也能保证对用户兴趣预测的延续性。推荐系统推荐结果变化程度被定义为时间多样性，时间多样性高的系统中用户经常会看到不同的推荐结果。提高时间多样性一般分两步：首先系统在用户有新行为后及时调整推荐结果，其次，在用户没有新行为的时候也能经常调整结果，以提供一定的时间多样性。如果没有行为数据，需要在推荐时加入一定的随机性。记录用户每天看到的推荐结果，然后在每天推荐时对之前看到过的推荐结果适当降权，每天给用户使用不同的推荐算法，可以设计多种算法，每天随机选择一种算法做推荐。

利用上下文的信息

- 用户的兴趣还与地点相关，不同地方的用户兴趣存在着很大的差别，不同国家和地区的用户兴趣存在着一定的差异性。一个用户往往在附近的地区活动。因此，在基于位置的推荐中需要考虑推荐地点和用户当前地点的距离。不能给用户推荐太远的地方。在资讯、购物、旅游等应用中引入位置会使推荐结果更加本地化，容易引起用户注意。

利用社交网络数据

- 在一些主流社交网络中，人们之间可能存在亲属关系、工作关系和共同兴趣，一般的社交网络中虽然用户间可能没有明确的关系，但是包含了用户属于不同社区的数据。通过社会化的推荐可以解决冷启动问题，增加推荐的信任度。


THANK
YOU