推荐-召回

姚凯飞

2017.7.30

什么是推荐召回

- 推荐系统中的一个模块
- 推荐系统中的一个流程
- 推荐数据流漏斗的顶部

- 海选/初选

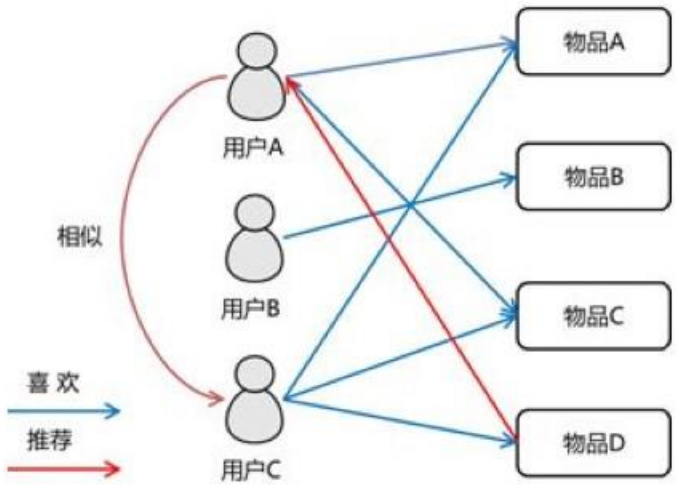
为什么做推荐召回

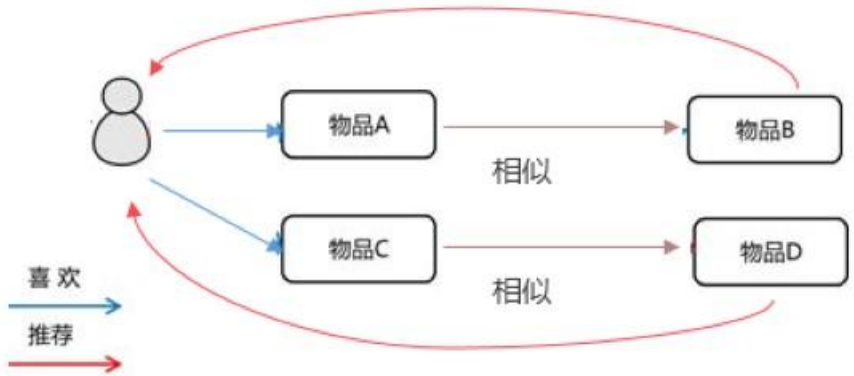
- 历史演化的存在
- 成本与收益的平衡
- 多样性问题的解决

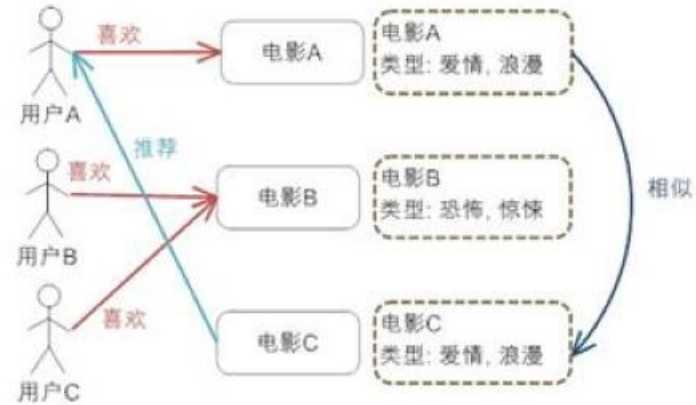
召回四象限

- 流行
- 多样
- 新鲜
- 相关

召回简图







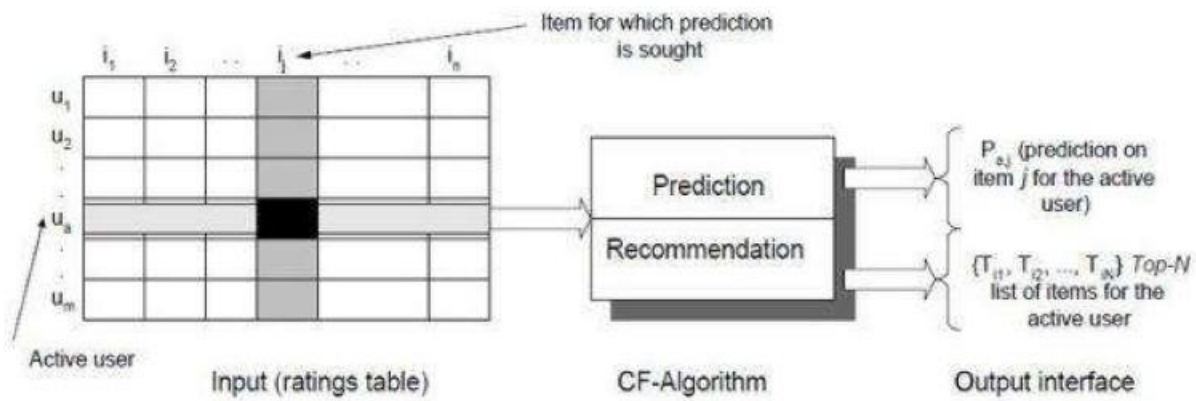
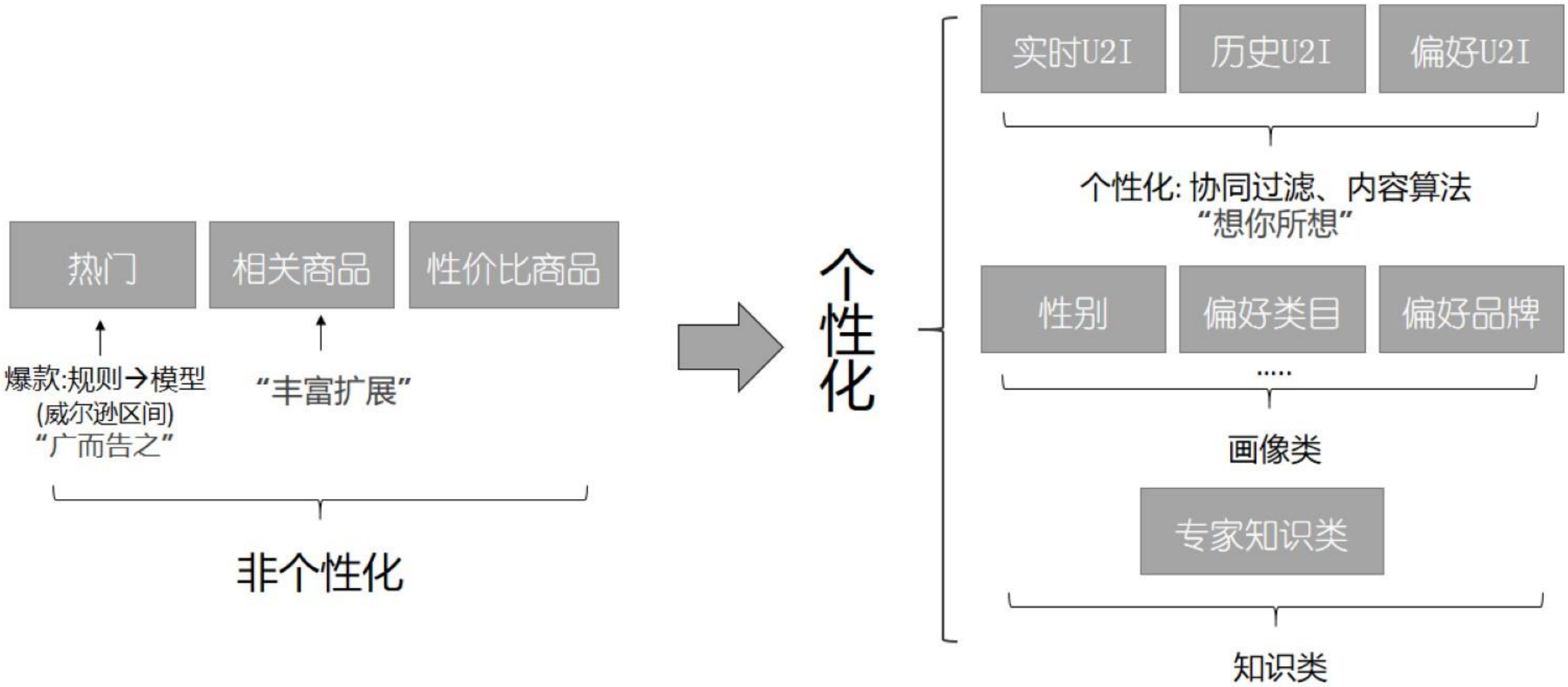


Figure 1: The Collaborative Filtering Process.

个性/非个性化召回-常用策略模型



相似度计算

$$d(x,y)=\sqrt{(\sum(x_i-y_i)^2)}$$

$$\cos(\theta)=\frac{a\cdot b}{|a|\,|b|}$$

$$J(A,B)=\frac{|A\cap B|}{|A\cup B|}$$

wbcosine

$$\mathrm{Sim}(I_i,I_j)=\frac{\sum_{u\in U_i\cap U_j} W_u^2/(1+\delta(abs(t_{ui}-t_{uj})))}{\sqrt{\sum_{u\in U(I_i)} W_u^2}\sqrt{\sum_{u\in U(I_j)} W_u^2}}$$

$$W_u=\frac{1}{\log_2(3+q_u)}$$

covariance

$$\rho_{XY}=\frac{Cov(X,Y)}{\sqrt{D(X)}\sqrt{D(Y)}}=\frac{E((X-EX)(Y-EY))}{\sqrt{D(X)}\sqrt{D(Y)}}$$

相似度计算注意点

- 在不丧失区分度的情况下，空间上尽量稠密
- 经验目标：稀疏度>1%
- 横向结合+纵向结合
- 相似度归一化(提高推荐的准确度)
- 时间因子(需要对历史共现的数据和历史频次的数据进行降权，要更加侧重于新数据的影响力)

画像过滤

Item/Content profile

- 构成:内容标签
- 静态数据:
  - 内容的keyword, 分类，热点标签，如标题党，图文的色情恶心标签，用户的情感极性等等
  - 对于content profile而言，就是尽可能的做好标签的准确度和覆盖率。

User profile

- 构成：用户标签
- 动态数据：
  - 用户的历史行为（用户过去一段时间的历史行为，用于对用户兴趣进行预判）
  - 用户的session行为（用户实时交互过程中产生的行为标签）
- 静态数据：
  - 用户的自身的标签（age、gender、职业、收入等等）。

lr

gbdt

svm

rnn

cnn

lstm

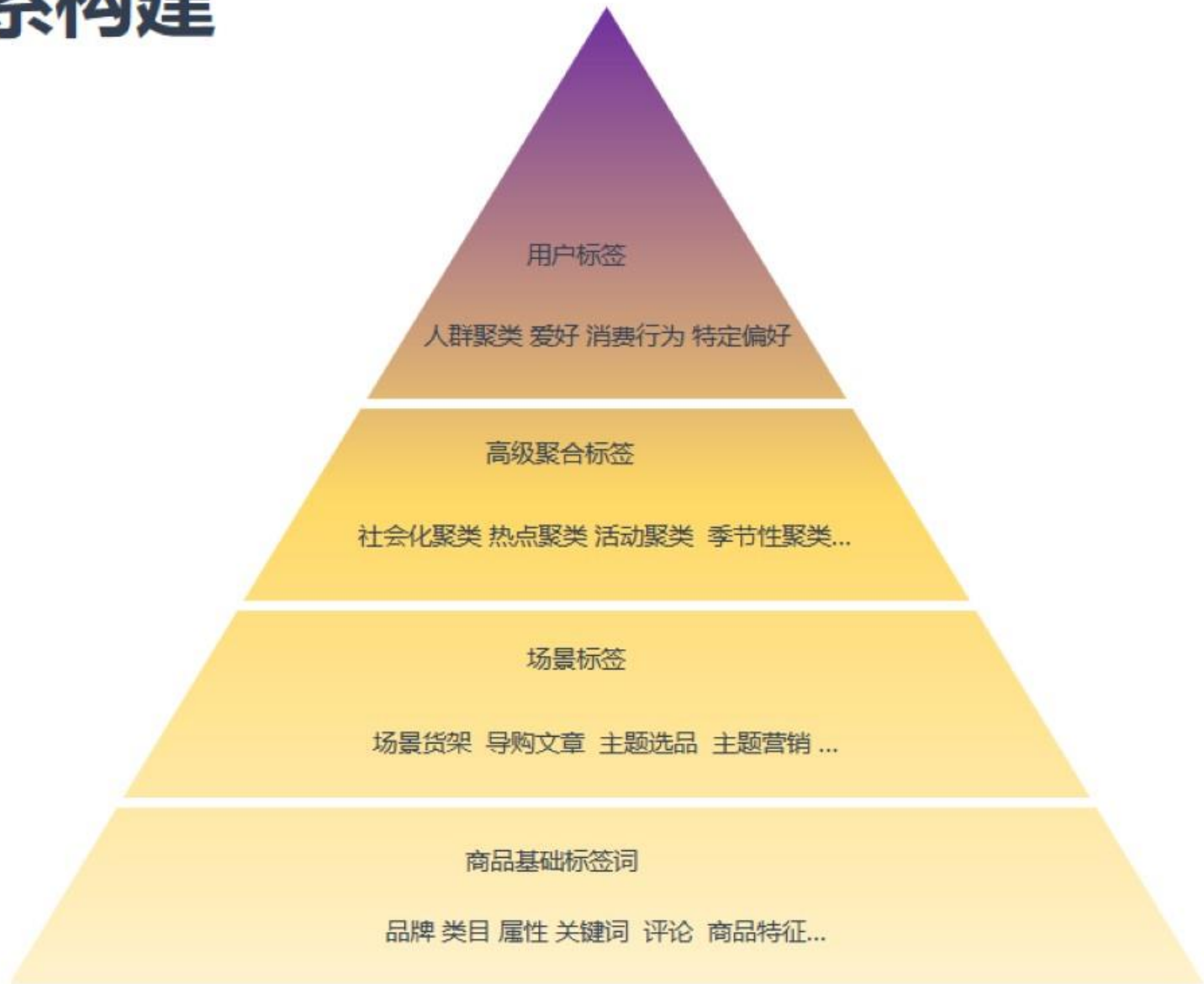
word2ve

c

fasttext

用户的行为 = 商品/内容-信息载体(明星、类目、年代) + 显性操作(购买、常看、关注、下载、收藏) + 隐形操作(时长、跳过)

标签体系构建


用户标签
人群聚类 爱好 消费行为 特定偏好
高级聚合标签
社会化聚类 热点聚类 活动聚类 季节性聚类...
场景标签
场景货架 导购文章 主题选品 主题营销 ...
商品基础标签词
品牌 类目 属性 关键词 评论 商品特征...

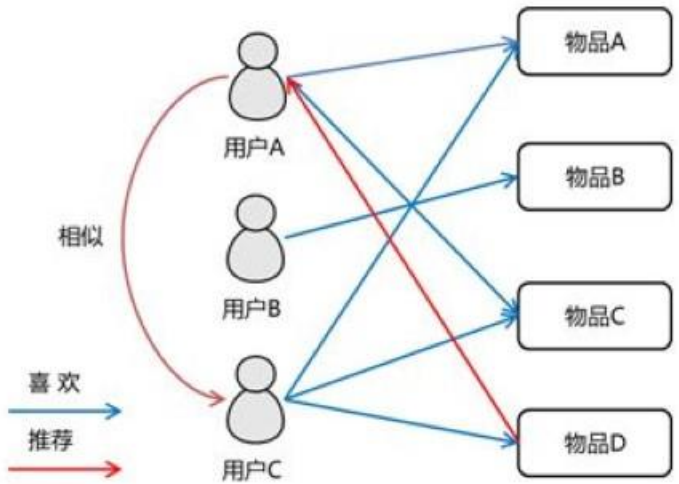
标签建模

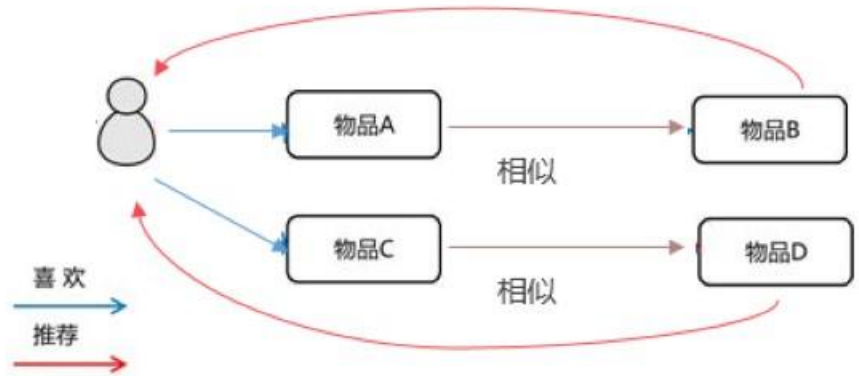
关键词偏好

- 通过用户点击，购买，成交，收藏的商品的标题挖掘用户所关注的语意单元的信息
- 方法：
  - 基于历史商品标题分词粒度的 TF-IDF 统计模式
    - 问题：
      - 粒度太细，用户在单个词上难有长期偏好
      - 词太多，存储空间大
      - 页面展示效果较凌乱
  - 建立基于<user-商品>原始统计为基础的topic model 的解决方案
    - PLSA
    - LDA
    - 人工review topic下的词

协同过滤

相似度计算-k近邻-偏好预测





协同过滤

- user base cf
  - 一些问题:如何度量相似性，考虑多少的邻居，如何从邻居中得到评分
  - 公式与解释(cosine/pearson)
  - 线上如何预测使用
  - 平衡模型中活跃用户的影响
  - common-items越多的用户可以得到越大的权重
  - 置信度问题
  - 近邻数量的选取

- item base cf
  - 由于user-cf可扩展性/计算空间与时间等问题，item-cf被使用
  - 用户间的交集的稀疏性问题
  - 挖掘商品间的关系来取代挖掘用户间关系
  - 比user-cf更稳定
  - 阈值以及近邻的选取
  - 举例:
    - 看了又看
    - 收藏了还收藏了
    - 评论了还评论

协同过滤

user-based CF

- user-similarity

$$s_{uv} = \frac{|N(u) \cap N(v)|}{|N(u) \cup N(v)|}$$

- prediction

$$p_{ui} = \sum_{v \in S(u,k) \cap N(i)} s_{uv} p_{vi}$$

item-based CF

- item-similarity

$$s_{ij} = \frac{|N(i) \cap N(j)|}{|N(i) \cup N(j)|}$$

- prediction

$$p_{ui} = \sum_{j \in S(i,k) \cap N(u)} s_{ij} p_{uj}$$

注：实际使用中，距离计算公式有大量调整和变形

协同过滤

- 优点
  - 利用群体智慧，无需依赖背景知识
  - 通用性高
  - 使用最广
  - 可解释强
- 缺点
  - 强依赖用户行为数据
  - 数据稀疏性问题-经验目标：稀疏度>1%
- 方法本质: 群体智慧
- 前提假设与条件:
  - 用户给予不同物品不同的评分(隐示/显示)
  - 用户在过去有某种品味，未来也将有相似的某种品味

协同过滤　User CF vs. Item CF　“集体智慧”

	群体/个体	计算代价	使用场景	冷启动	可解释性	实时性
User-based	更依赖与当前用户相似的用户群体的社会化行为	适用于用户数较少的场合（电商）	时效性强，用户个性化兴趣不太显著场景	新加入的物品能很快进入推荐列表中	弱	用户的新行为不一定导致推荐结果的变化
Item-based	更侧重与自身的个体行为	适用于物品数较少的场合(新闻)	长尾物品丰富，用户个性化需求强烈的场合	新加入的用户能很快得到推荐	强	用户的新行为一定导致推荐结果的变化

精度:

50% 一样，50% 不同；但相似的精度(互补)。

多样性:

单用户(Item CF < User CF)

系统(覆盖率)(Item CF > User CF)(Item CF 擅长长尾)

用户对推荐算法的适应度:

前面我们大部分都是从推荐引擎的角度考虑哪个算法更优，但其实我们更多的应该考虑作为推荐引擎的最终使用者 -- 应用用户对推荐算法的适应度。

CF问题

$$w_{ij} = \frac{|N(i) \cap N(j)|}{|N(i)|^{1-\alpha} |N(j)|^{\alpha}}$$

$$w_{ij} = \frac{\sum_{u \in N(i) \cap N(j)} \frac{1}{\log 1 + |N(u)|}}{\sqrt{|N(i)||N(j)|}}$$

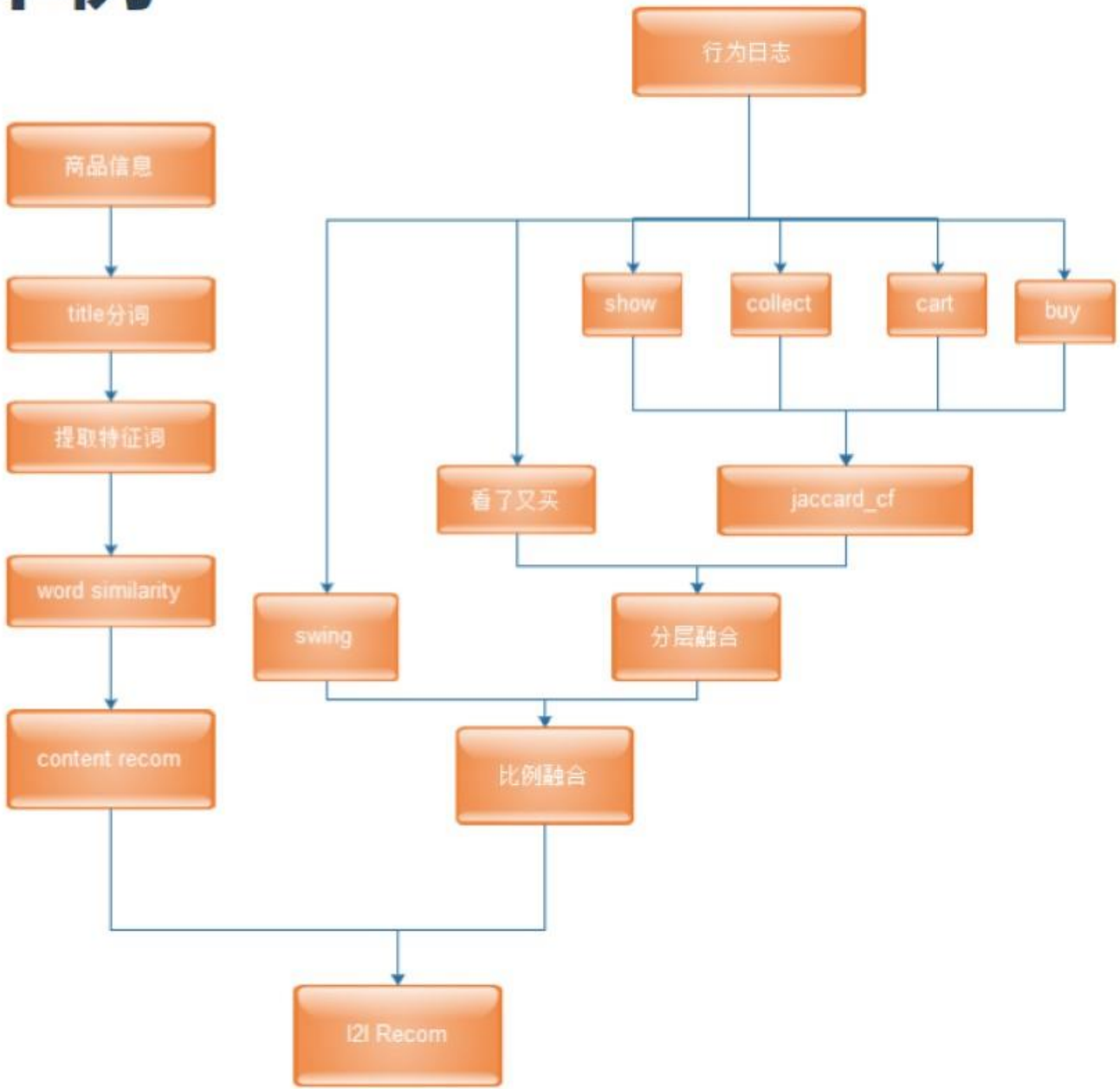
1.热门Item

2.过度活跃用户

3.User Base CF 的第一推动力问题(第一个用户从哪儿发现新的物品)

4.Item Base CF 物品冷启动

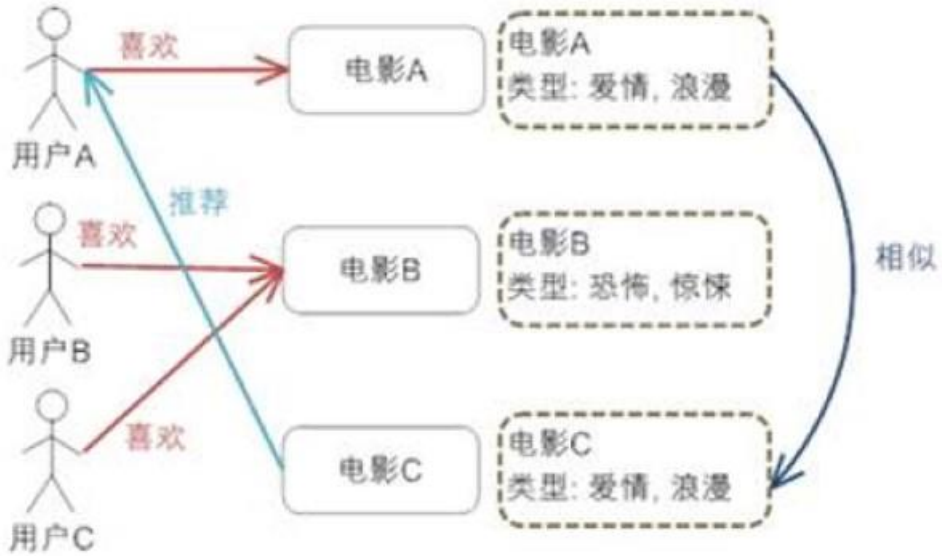
举例



- 单天频次、历史频次、单天共现、历史共现
- webcosine, adjust-cosine, swing
- 自动化评测工具和DEMO中心

内容过滤

基于行为的冷启动问题



- 信息检索领域最先出现的概念及应用
- 发现用户对文本(key-word)的兴趣
- 基于内容与基于知识有着很模糊的边界
- 用户偏好比显示行为更具有知识
- 商品的属性及相关内容信息

基于内容的工作就是总结出用户的偏好及定义商品内容与用户偏好的相似度

内容过滤

- 优点
  - 无需依赖用户数据，回避产品初期的用户不足集稀疏性问题(解决新商品的相似性)
  - 覆盖率高(提供更多的相似性)
  - 能够描述用户偏好的画像
- 缺点
  - 数据建设成本大(文本抽取等问题)
  - 人对内容理解的多样性，多层次
  - 内容相似性无法表现出质量及相关性
  - 时效性

- **图像**
- **视频**
- **音频**
- **文字**

内容过滤-常见问题

- 短文本
- 时效性
- 文本抽取算法

- 向量模型
  - tf/idf，tf-idf，cosine-similarity
  - 停用词
  - stem
  - 截断 topN
  - 特征选取

- 举例
  - 同主演，同导演
  - 同题材
  - 标题相似
  - 内容相似

模型过滤

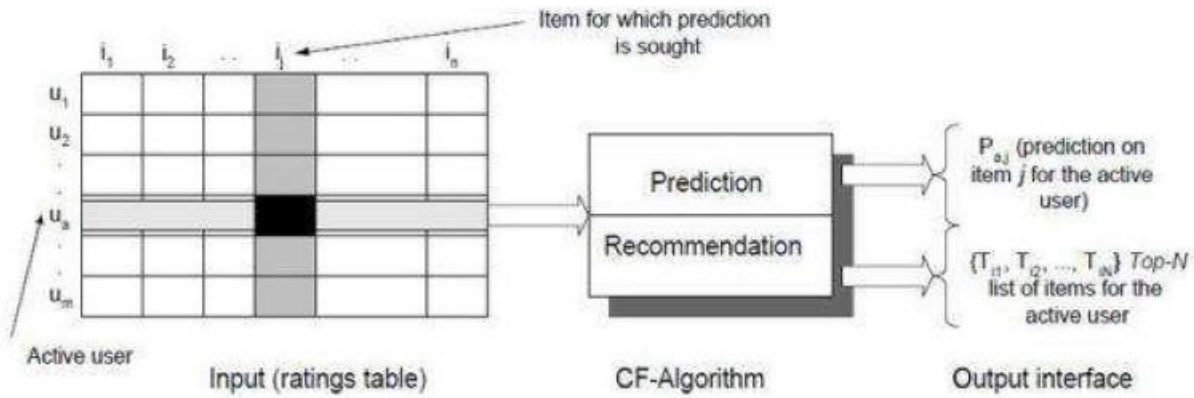


Figure 1: The Collaborative Filtering Process.

模型过滤

部分user-item关系-用隐变量刻画user-item关系-偏好预测

- offline处理和模型学习
- 在线使用离线学习到的模型进行预测
- 模型在一定周期后更新或重新训练
- 模型的更新和重新计算非常耗时及计算资源
- 概率模型:聚类模型贝叶斯网络/LDA

模型过滤-相关

- 假设：过去经常被一起频繁消费的商品，今后也会被一起消费
- 利用事先确定好的支持度/置信度/提升度，计算关联商品
- 适合长尾商品做有效预测，适合session/transaction数据
- 用户消费差异性被忽略，不是很适合个性化推荐

知识过滤

- 基于专家知识需要领域专家知识
- 转移产品知识（“教育用户”）
- 正确的建议
- 显性知识与社区数据合并
- 声明式知识库
- 举例医疗AI
- 需要专家知识
- 对于偏好的准确性
- 稳定与自由度假设

混合过滤

混合的推荐机制

1.加权的混合（Weighted Hybridization）(不同策略不同的权重)

2.切换的混合（Switching Hybridization）(根据场景切换)

3.分区的混合（Mixed Hybridization）(各区分数可比)

4.分层的混合（Meta-Level Hybridization）(1234分策略)

算法融合

图过滤

- PersonalRank
- simrank /simrank++
- Adamic/Adar
- swing/wswing

热门过滤

- 爆款

规则过滤

- 运营BD需要

各种栏位场景适用推荐算法

	用户场景	业务意图	算法							
			短期意图	长期画像	协同过滤	关联规则	周期购	刚需爆款	冲动性爆款	主题推荐
首页	弱目的性闲逛	trade cross		✓	✓	✓	✓	✓	✓	✓
	强目的寻找相关促销或爆款	trade in	✓							
类目/搜索	寻找更适合自己的商品	trade in		✓						✓
	了解同类人群的购买选择	trade in			✓					
详情页	挑选、比较的需求	trade in		✓	✓					
	了解同类人群的购买选择	trade in			✓					
	经济节约的诉求	trade in								
	对相关商品的潜在购买提醒	trade cross				✓				
继续逛页	对相关商品的潜在购买提醒	trade cross				✓				✓
购物车	凑单免邮的需求	trade up		✓		✓	✓	✓	✓	
	对相关商品的潜在购买提醒	trade cross		✓		✓	✓	✓	✓	
	占便宜	trade up	✓	✓		✓	✓	✓	✓	
订单完成页	一次购物周期结束，顺便看看有没有其他购物心动点	trade cross	✓	✓		✓	✓	✓	✓	
消息触达	对于关注的商品、品牌，了解其有利的动态	trade in	✓	✓						✓

不同位置的推荐产品定位不同

- 单品页：购买意图
- 过渡页：提高客单价
- 购物车页：购物决策
- 无结果页：减少跳出率

- 订单完成页：交叉销售
- 关注推荐：提高转化
- 我的xxx推荐：提高忠诚度

.....

几个注意点

各类算法比较

	冷启动	可解释性	惊喜	时效性	鲁棒性
协同过滤	★	★		★	★★
图模型		★	★		★
矩阵分解	★	★	★	★	★
Topic Model	★	★★		★	
增强学习	★	★		★★	
决策树		★	★	★	★
Boosting	★	★	★	★	★

Item2vector

- 类似word2vector思想。
- Word2vec利用文本中每个词的前后依存关系来建模，并将word映射为一个向量。
- Item2vec前后item的依存关系是由每个用户在网站的交互item列表构建。

召回---trigger

- 上下文相关
  - 与上下文内容相关
  - 与上下文发布者相关
- 行为相关
  - 协同过滤
  - 实时反馈
- 热门
- 兴趣相关
  - 基于关系的兴趣
  - 基于消费行为的兴趣
  - 长期兴趣 & 短期兴趣
  - 人群兴趣

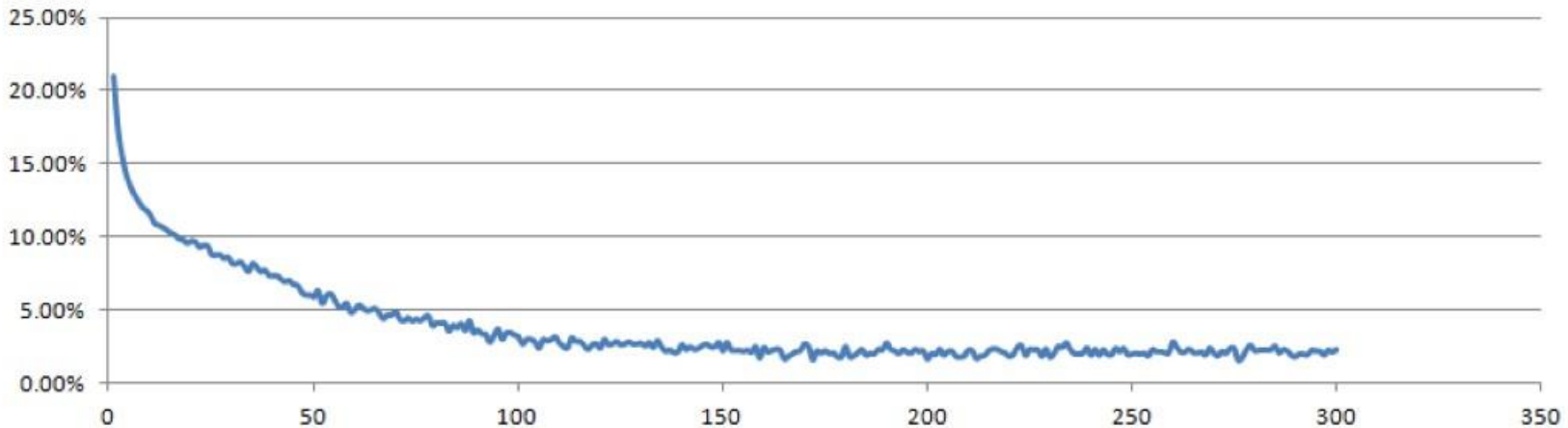
召回问题

- 多样性-人类视觉窄化
- 稀疏性
- 马太效应

离线与实时召回

$$S_{adjusted}(user,item) = e^{-\lambda * show(user,item)} S(user,item)$$

当前点击“甜美”的人，未来点击“甜美”的概率



横轴是未来点击与当前的点击次数间隔，即未来第n次点击
纵轴是点击“甜美”的概率

反馈数除以热门程度 $interest(u,c_i) = \frac{D(u,c_i)}{D(c_i)}$

$$f(x) = Eval\{\sum_{t=T0}^{T0+\Delta T} X(t), \Delta T\}$$

召回演进1.0—统计方法推动

- 规则
- 热销，爆款模型

召回演进2.0--算法推动

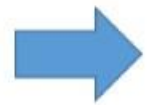
- 关联规则
- 协同过滤

召回演进3.0—架构通用化

- 自适应召回context/profile/parameter
- cf
- 情境(节日/季节)
- 反向(复购时间)
- 主题(商品聚合)
- 画像

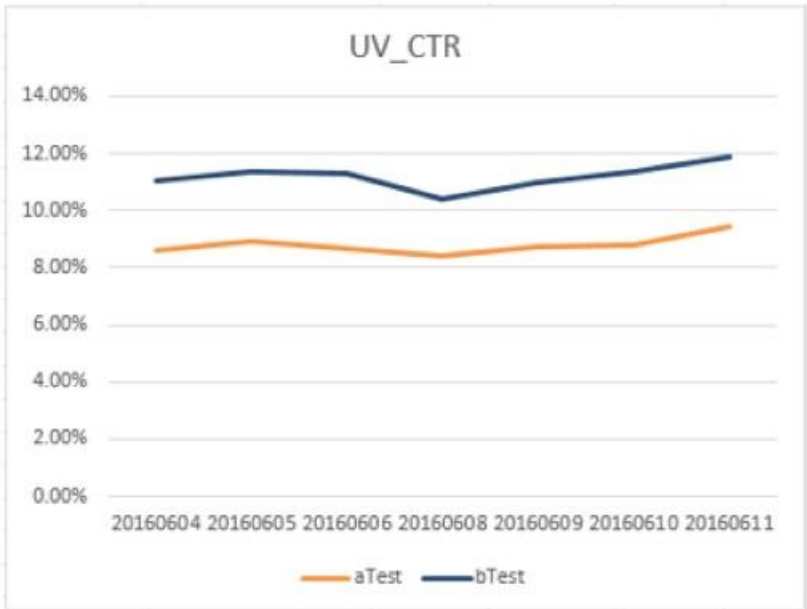
召回评估

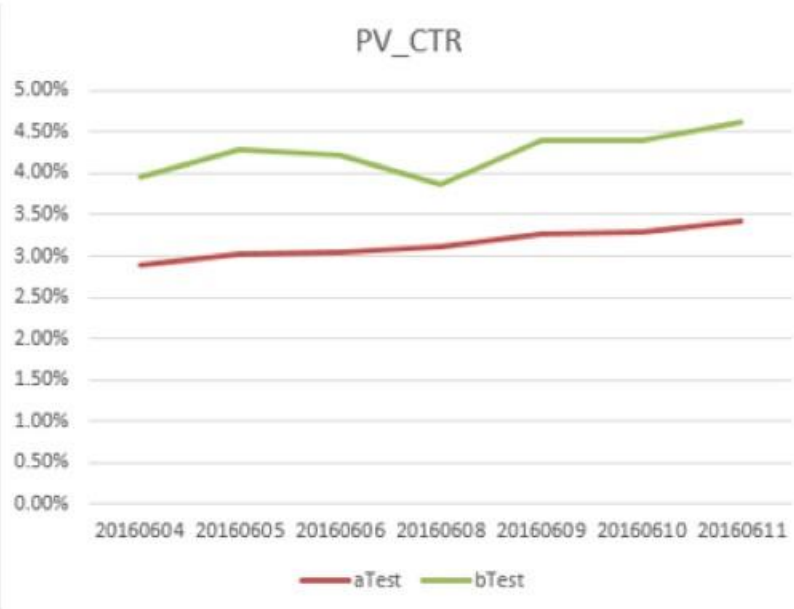
- 命中率
  = 推荐命中的i2i_pair/点击i2i_pair *100%



- 准确率
- 召回率
- F1
- 覆盖率
- 零结果率
- 深度

召回评估


UV_CTR
14.00%
12.00%
10.00%
8.00%
6.00%
4.00%
2.00%
0.00%
20160604 20160605 20160606 20160608 20160609 20160610 20160611
aTest
bTest


PV_CTR
5.00%
4.50%
4.00%
3.50%
3.00%
2.50%
2.00%
1.50%
1.00%
0.50%
0.00%
20160604 20160605 20160606 20160608 20160609 20160610 20160611
aTest
bTest


GMV
50000
45000
40000
35000
30000
25000
20000
15000
10000
5000
0
20160604 20160605 20160606 20160608 20160609 20160610 20160611
aTest
bTest