

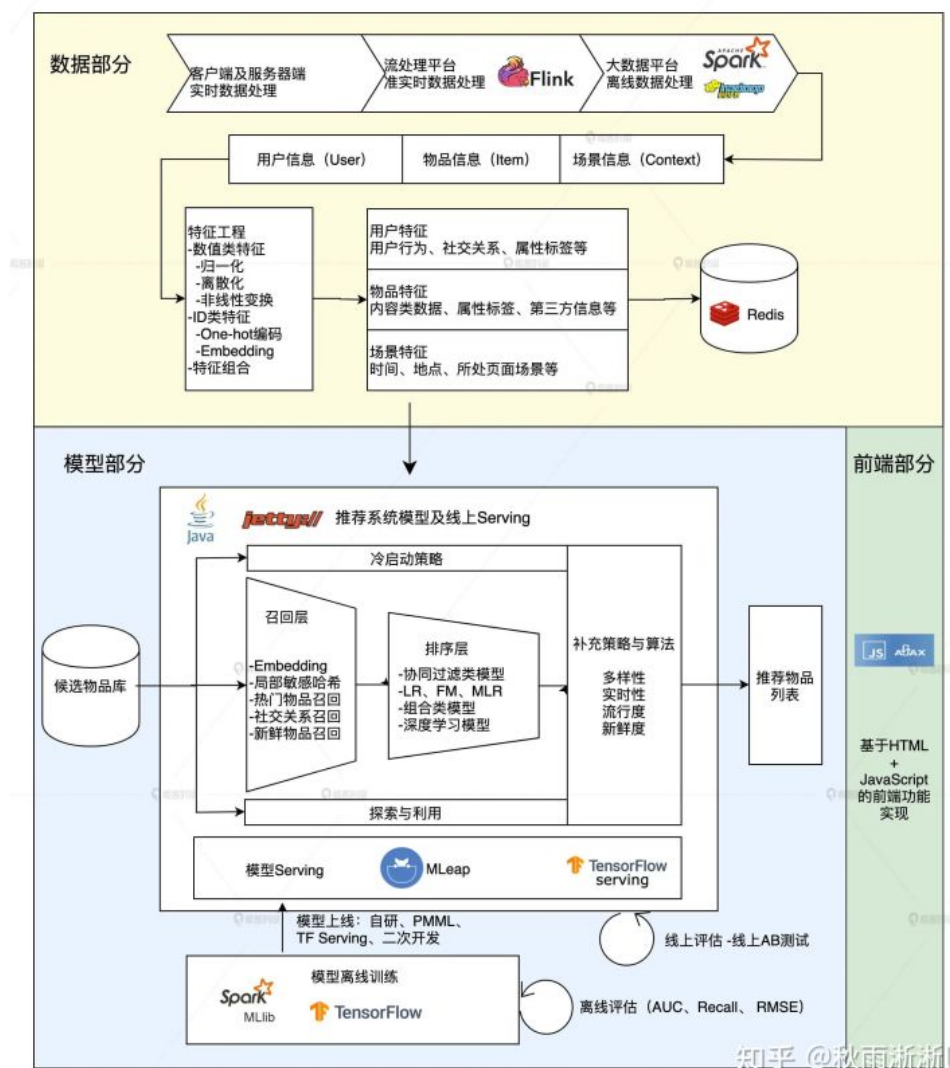
推荐系统

[[TOC]]

0.资料网址:

- 《推荐系统实践》-- 项亮

推荐系统流程:



三大模块

- 召回
- 粗排
- 精排

1. 基于邻域的算法

1.1 基于用户的协同过滤

- 找和用户兴趣相似的别的用户
- 把别的用户感兴趣的物品(行为相似度)也推给他
- 使用 物品-用户倒排表避免相似度计算
- 用户对热门商品的共同兴趣的权重应该比对冷门商品的低

1.2 基于物品的协同过滤

- 基于物品的相似度进行推荐
- 相似度并不完全取决于物品，而是取决于用户的行为，如果多位用户购买A的同时也买B，认为A和B相似
- 也需要惩罚不热门的物品

2. 推荐系统的冷启动问题

2.1 物品的冷启动

- 利用物品的内容信息衡量物品的相似度

2.2 用户的冷启动

- 让新注册的用户填写一些信息
- 如果用户用关联账户注册，可以分享用户画像

3. 多样性

Recommendation Goals

Exploitation

- **Relevant** - users are likely to take the actions which are shown (browse series, watch content, add to watch list, ...)
- **Transparent** - it's clear why the content selected is being recommended to the user
- **Contextual** - the content selected is timely and relevant to a person's profile, device, and location
- **Adaptive** - recommendation results adapt quickly to new users, new behavior, and newly available content

Exploration

- **Coverage** - new content has an opportunity to surface for a user (not churning between same actions over and over). A significant portion of available content are recommended
- **Serendipitous** - recommending content not obviously linked to previously expressed preferences
- **Diverse** - Don't show the same recommendation to a user many times in a short period. Mixture of various content type, genre, or network / brand

3.1 多样性

- 对用户：
 - 同质化的内容可能会让用户看腻了，多样性可以降低用户疲劳
 - 多样性也可以更全面的发掘用户的兴趣，提高用户粘性
- 对产品：
 - 多样性有利于长尾的视频和作者

3.2 多样性的定义

- 用户视角
 - 推荐的物品差异性大
 - 覆盖用户的不同兴趣
- 平台视角
 - 能够推荐到长尾的物品，以及发掘出新的作者

3.3 如何做多样化

- 用户侧
 - 减少密集推荐同质化内容
 - 多探索用户的不同兴趣
- 平台
 - 促进平台内容的多样化

3.4 用指标衡量多样性

- ILAD指标
 - 先对每个用户的每次推荐的物品的相似度求平均
 - 再对所有用户的相似度求平均
 - 1-
- ILMD指标
 - 求出推荐给用户的物品的最小相似度
 - 再对所有用户的相似度求平均
 - 1-

统计类指标：

- 覆盖率、渗透率
 - 人均看过的tag数目
- 推荐密集的比例
 - 来自相同作者等的比例 >0.3
- 系统指标
 - 不同召回的来源占比
 - 不同粉丝、视频长度等占比

3.5 提升用户推荐多样性的手段

减少密集推荐

减少密集

多样化召回

- 规则召回：多路召回（关注的人、经常看的人、收藏过的作者、查过的词条等等）
- Embedding-based召回：加入噪声，建模target，建模方式（graph emb）
- 通用技巧：相似query替换、强制探索、随机流量

基于规则打散

- 频控：一天最多n个
- 队列打散
- 滑动窗口打散：相同品类m出n（tag、author、music）
- 强插：第n位插入



基于数值打散

- 非互动：MMR, DPP
- 互动：UCB, Reinforcement Learning, DQN

• MMR

- 用超参调节 物品与用户的相关度 以及 物品之间的相似度

减少密集

1. MMR (Maximal Marginal Relevance)

Algorithm 1 MMR: Maximal Marginal Relevance

Input: Ground set of items E

$S \leftarrow (), L = |E|$

while $|S| < L$ **do**

- $e^* \leftarrow \arg \max_{e \in E-S} (1 - \lambda)w(e) + \lambda(f(S + e) - f(S))$

Append item e^* to list S

Output: List of recommended items S

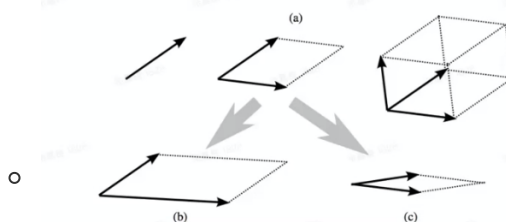
$$f(S + e) - f(S) = -\max_{e' \in S} \text{sim}(e, e')$$

$$MMR \stackrel{\text{def}}{=} \text{Arg} \max_{D_i \in R \setminus S} \left[\lambda(\text{Sim}_1(D_i, Q)) - (1 - \lambda) \max_{D_j \in S} \text{Sim}_2(D_i, D_j) \right]$$

• DPP

- 最大化行列式

2. DPP (Determinantal Point Process)



• For DPP, there exists a PSD $M \times M$ matrix L such that $\mathcal{P}(Y) \propto \det(L_Y)$

• Example: $M = 2, Z = \{1, 2\}, L = \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix}$

$$\begin{array}{ll} \mathcal{P}(\emptyset) & \propto 1 \\ \mathcal{P}(\{1\}) & \propto 2 \\ \mathcal{P}(\{2\}) & \propto 1 \\ \mathcal{P}(\{1, 2\}) & \propto 1 \end{array} \quad \Rightarrow \quad \begin{array}{ll} \mathcal{P}(\emptyset) & = 0.2 \\ \mathcal{P}(\{1\}) & = 0.4 \\ \mathcal{P}(\{2\}) & = 0.2 \\ \mathcal{P}(\{1, 2\}) & = 0.2 \end{array}$$

$$L_Y = \begin{bmatrix} L_{11} & L_{12} \\ L_{21} & L_{22} \end{bmatrix}$$

$$\det(L_Y) = L_{11}L_{22} - L_{12}L_{21}$$

$$L_{ij} = q_i * q_j * S_{ij}$$

$$S_{ij} = \frac{1 + \langle f_i, f_j \rangle}{2} = \left\langle \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ f_i \end{bmatrix}, \frac{1}{\sqrt{2}} \begin{bmatrix} 1 \\ f_j \end{bmatrix} \right\rangle \in [0, 1].$$

$$R_u = \arg \max_{Y \subseteq Z \setminus P_u} \mathcal{P}(Y) \propto \det(L_Y) \quad \text{s.t.} \quad \#Y = N.$$

兴趣探索和学习

样本工程

- 主动探索，无偏流量
- 引入其他场景的消费样本

特征工程

- 主动探索，无偏流量
- 引入其他场景的消费样本

训练范式和模型结构

- Point-wise ranking -> List-wise ranking
- 多目标建模
- 兴趣建模

兴趣探索和学习

- 兴趣建模：从历史行为挖掘兴趣
 - DIN, DIEN, DISN, MIND
- 长短兴趣结合：记住长期兴趣
 - MIMN, HPMN
- 因果建模：发现真的兴趣，而不是人云亦云
 - DICE

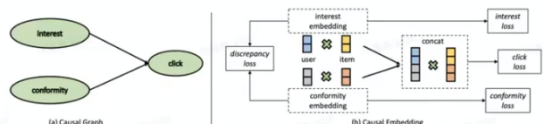
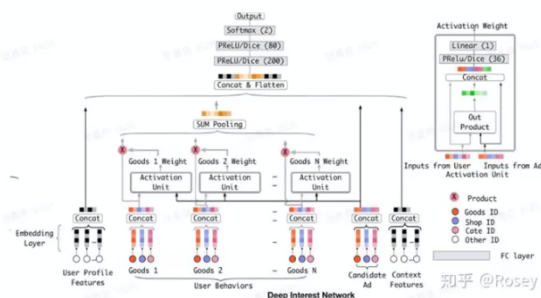
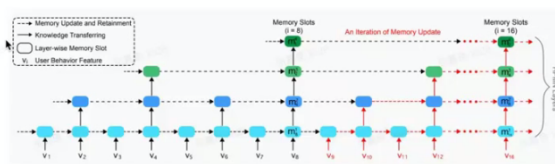


Figure 3: Causal graph and causal embeddings. (a) We make concise causal modeling on each click that it results from two independent causes, interest and conformity. (b) We adopt separate embeddings for interest and conformity, thus each user or item has two embeddings. We force each embedding to capture only one cause by training different embeddings with cause-specific data and adding direct disentanglement supervision, under the framework of multi-task curriculum learning.



4. 召回部分

Deep Retrieval: 大规模召回场景下的端到端模型

背景

召回模型现状

- 1. 两段式模式
 - a. 训练一个点乘模型
 - b. 利用FBT或者HNSW (或者其他ANN), 来设计高效的近似近邻搜索算法。这些算法侧重于快速和有效的召回最近邻目标
- 2. 主要缺点
 - a. 模型结构局限于双塔结构, 无法进行复杂模型调研

Deep Retrieval简介

1. Deep Retrieval (简称DR) 是字节跳动Data-AML原创的端到端召回算法系统。DR同时学习模型参数 (NN, embedding等) 和一个可以快速搜索的 K^D 离散结构, 从而能把学习的目标和召回索引更加直接和紧密的联系起来。从这个角度来讲, DR和近似近邻搜索算法在本质上有较大的不同。

Deep Retrieval主要贡献

1. 端到端的学习, 打破小纬度Vector点乘和距离搜索限制, 索引可以和任意深度复杂结构结合
2. Graph编码, 单item(group, ad)可以赋予多个索引, 表征形式强于传统树形单索引结构
3. 算法系统为产品而设计, 采用online EM算法支持大规模分布式在线学习
4. K^D 编码常数复杂度不受候选集影响, Beam Search高效召回serving
5. 证明了端到端学习在大规模召回场景的理论和实际上的可行性, 给了我们未来更大的想象空间

Paper by AML: [Deep Retrieval: An End-to-End Learnable Structure Model for Large-Scale Recommendations](#)

5. YouTube推荐

5.1 背景

Background:

Youtube视频推荐碰到的挑战:

1. 海量数据, 涉及到海量的用户和视频, 需要高效的分布式学习算法和线上服务系统;
2. 新鲜度
 - o YouTube视频更更新频率很高, 每秒有小时级别的视频上传, 需要在新发布视频和已有存量量视频间进行balance
 - o 用户实时行为切换很快, 模型需要很好的追踪用用户的实时行为
3. 数据稀疏性造成噪声:
 - a. 用户的行为记录很少, 并且受到外部影响, 不易预测。
 - b. Filter bubble (回声壁效应)
4. 用户满意度不好定义

5.2 系统架构

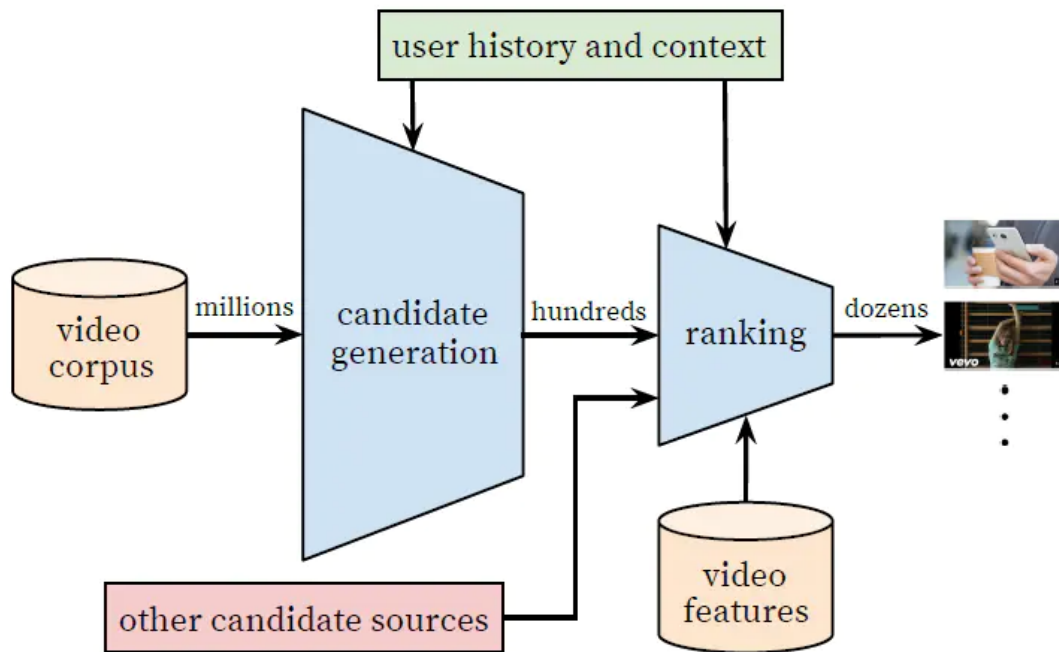


Figure 2: Recommendation system architecture demonstrating the “funnel” where candidate videos are retrieved and ranked before presenting only a few to the user.

- 候选集生成(召回)：从百万级的视频中快速精确的找到数百的视频
 - 指标：采用recall等，因为AUC是衡量排序的
- 粗排 (optional): 轻巧的精排，改精排做预删除，减少精排的压力。
 - 指标：AUC、跟精排的一致性
- 排序：对这数百个视频进行精排序，最后将排名靠前的呈现为用户。
- 问题：各自的主要模型指标是什么？ AUC？

DPP 论文: <https://proceedings.neurips.cc/paper/2018/file/dbbf603ff0e99629dda5d75b6f75f966-Paper.pdf>

Deepretival: <https://arxiv.org/pdf/2007.07203.pdf>