第三部分：排序

1、多目标排序模型

2、Multi-gate Mixture-of-Experts（MMoE）

3、预估分数的融合

4、视频播放建模

5、排序模型的特征

6、粗排

1、多目标排序模型

多目标排序模型，和下面的MMoE主要是用于精排

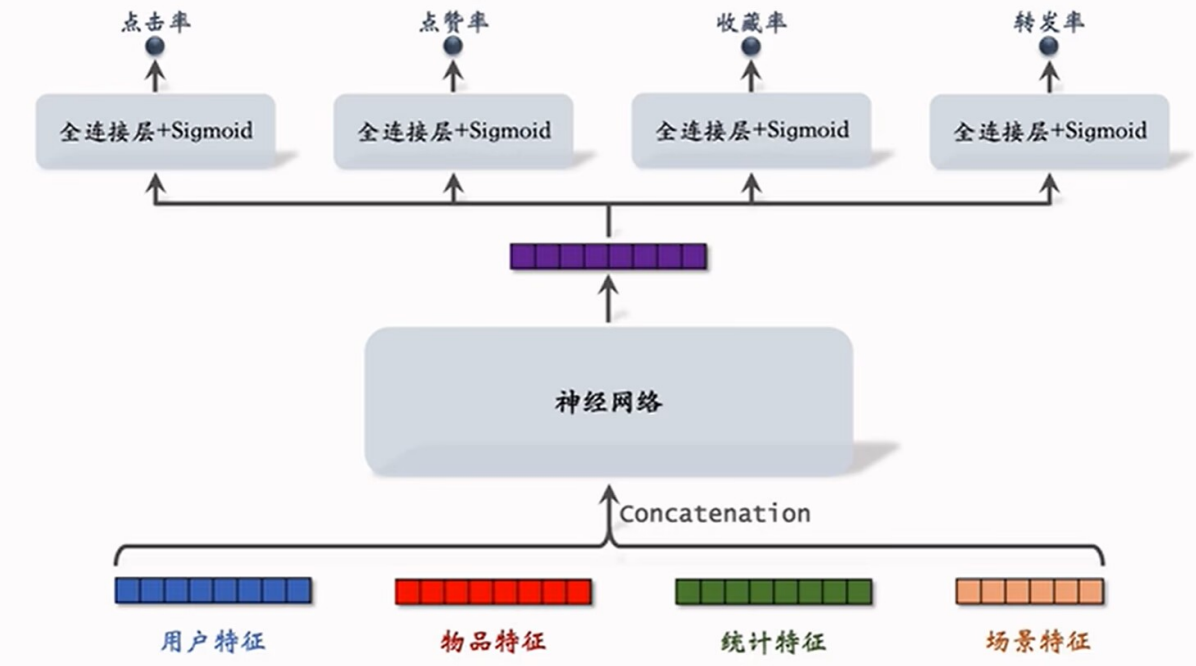
排序模型的依据

对于每篇笔记，系统记录以下用户-笔记的交互行为：

曝光次数、点击次数、点赞次数、收藏次数、转发次数

排序模型预估点击率、点赞率、收藏率、转发率等多种分数，利用加权和等方式融合这些分数，再根据融合的分数做排序

图例：



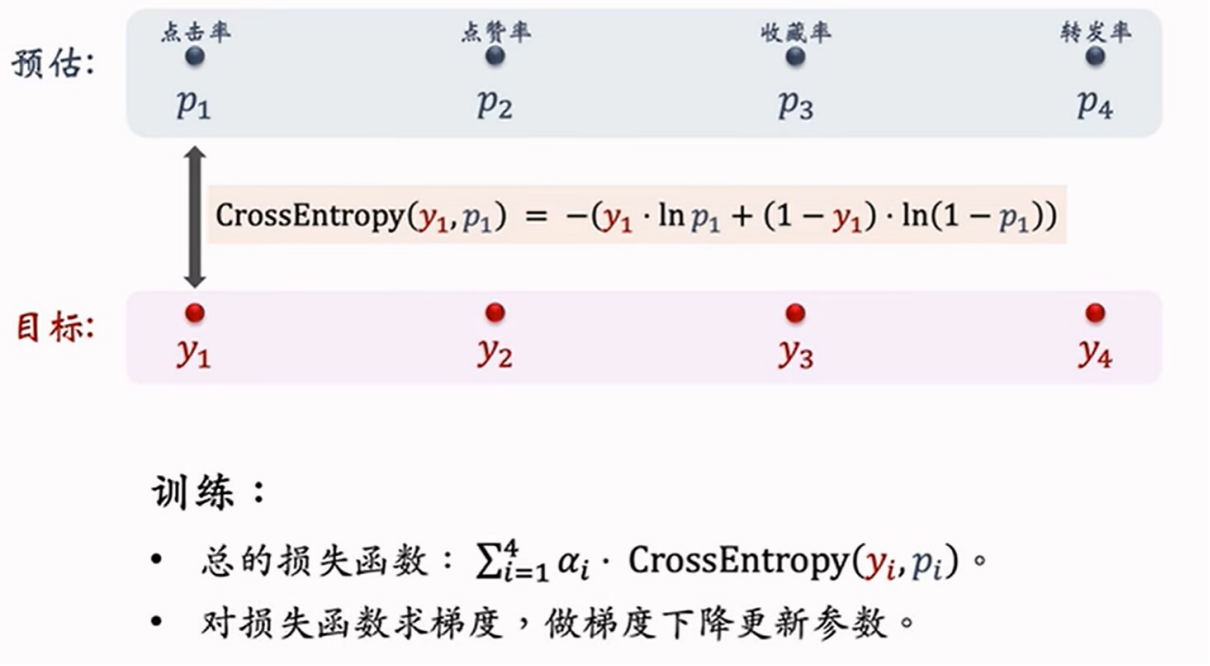
统计特征：包括用户的点击、点赞、收藏、转发等行为

场景特征：和用户行为关联的例如城市、日期等场景因素

依据得到的点击率、点赞率、收藏率、转发率进行排序

排序模型的训练

训练流程



本质是做二元分类任务，例如 y_1 为0则代表未点击，为1则代表点击

利用交叉熵损失函数进行优化，期望预估值和目标值接近

训练难点

类别不平衡问题：

100次曝光，约有10次点击，90次无点击

100次点击，约有10次收藏，90次无收藏

如上数据，说明正负样本的数量极不平衡，解决方案——

负样本降采样：保留一小部分负样本，让正负样本数量平衡，节约计算

预估值校准

正样本，负样本数量分别为 n_+, n_-

对负样本做降采样，抛弃一部分负样本，即使用 $\alpha \cdot n_-$ 个负样本，α∈（0，1）是采样率

由于负样本变少，预估点击率大于真实点击率；α越小，预估和真实的偏差越大

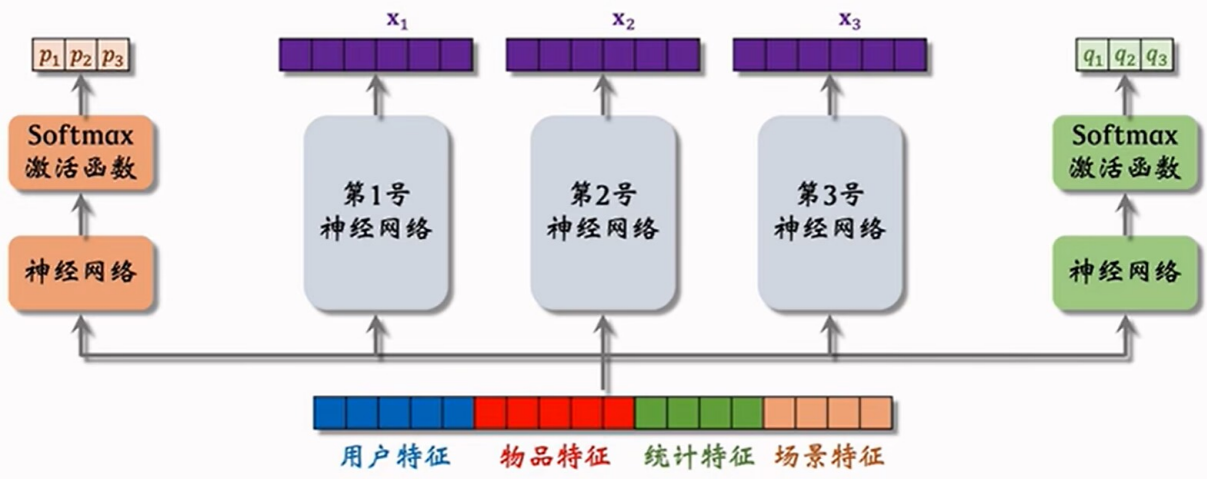
校准公式：

- 真实点击率：$p_{\text{true}} = \frac{n_+}{n_+ + n_-}$（期望）。
- 预估点击率：$p_{\text{pred}} = \frac{n_+}{n_+ + \alpha \cdot n_-}$（期望）。
- 由上面两个等式可得校准公式[1]：

$$p_{\text{true}} = \frac{\alpha \cdot p_{\text{pred}}}{(1 - p_{\text{pred}}) + \alpha \cdot p_{\text{pred}}}.$$

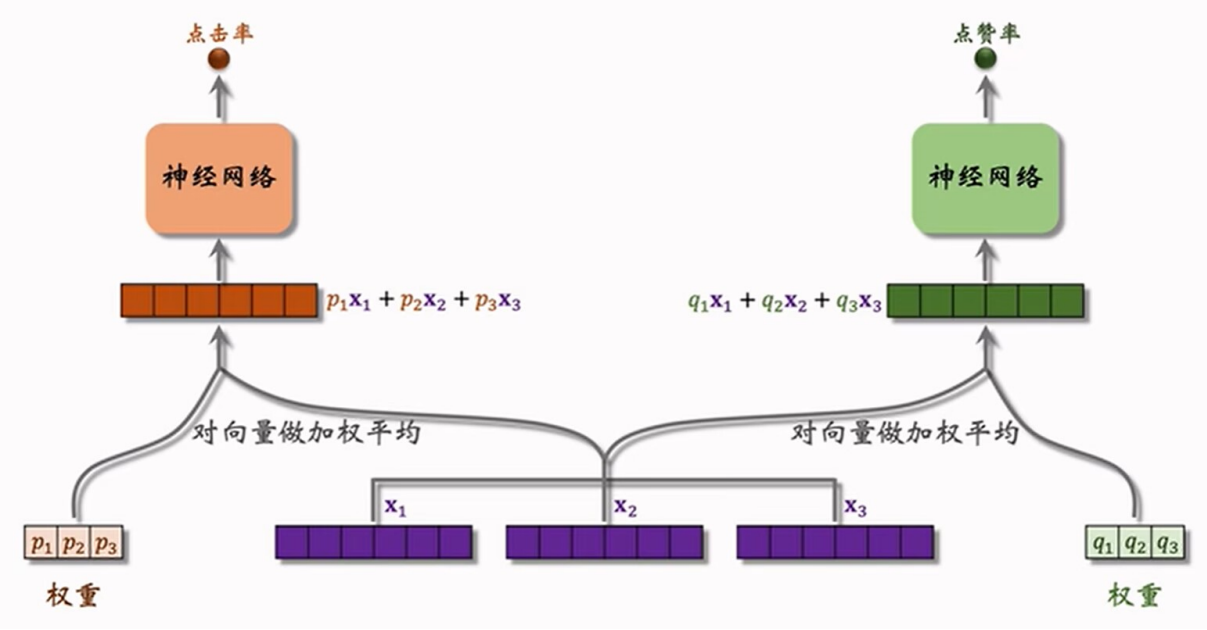
2、Multi-gate Mixture-of-Experts（MMoE）

模型流程



专家（Experts）：三个神经网络，专家数量是超参数需要调，一般是4个或8个

两旁神经网络输出的3个p（q）都大于0且相加为1，作为神经网络输出向量的权重，其中p和q的权重不同，进而输出不同指标的预估分数

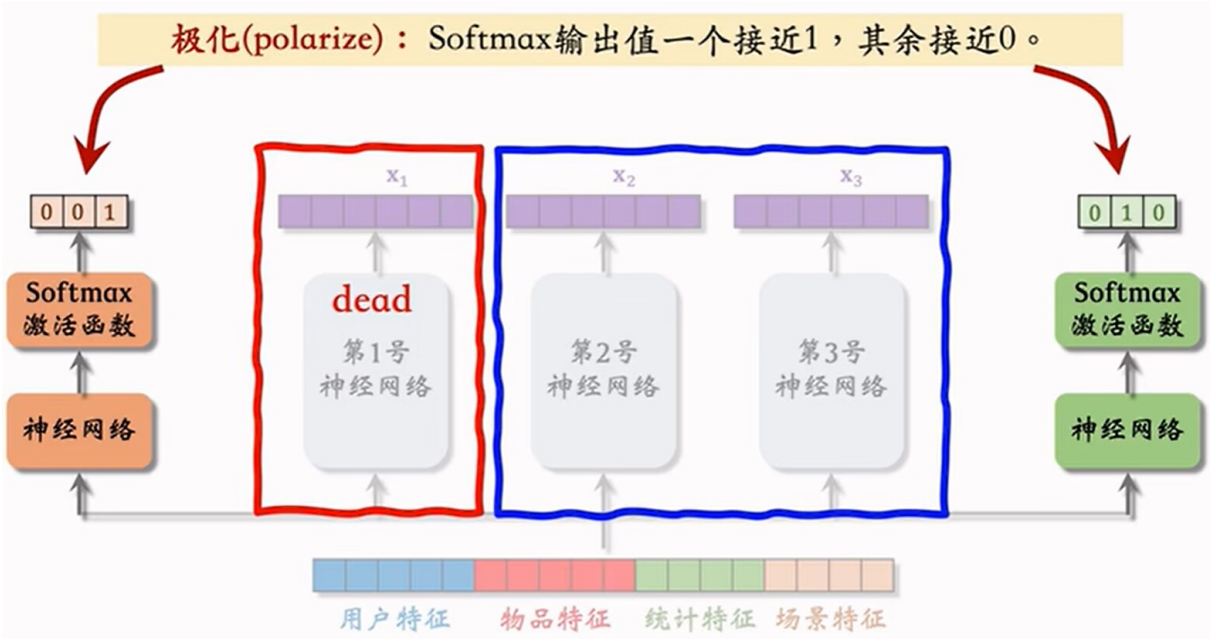


极化现象

内容

上面激活函数输出的向量3个值一个接近1，其余接近0

（相当于只用到了1个专家神经网络，没有实现多个专家神经网络的融合）



解决方法

训练时，对softmax的输出使用**dropout**

softmax输出的n个数值被mask的概率都是10%，即每个"专家"被丢弃的概率都是10%

*采用dropout后强迫每个任务根据部分专家做预测，能有效避免极化

（如果softmax输出为1的单元被mask，结果会错得很离谱）

3、预估分数的融合

普通方式（加权和）

简单的加权和

$$p_{\text{click}} + w_1 \cdot p_{\text{like}} + w_2 \cdot p_{\text{collect}} + \cdots$$

点击率乘以其他项的加权和

$$p_{\text{click}} \cdot (1 + w_1 \cdot p_{\text{like}} + w_2 \cdot p_{\text{collect}} + \cdots)$$

$p_{\text{click}} = \frac{\#\text{点击}}{\#\text{曝光}}$　　$p_{\text{like}} = \frac{\#\text{点赞}}{\#\text{点击}}$

海外短视频融分公式

海外某短视频APP的融分公式

$$(1 + w_1 \cdot p_{\text{time}})^{\alpha_1} \cdot (1 + w_2 \cdot p_{\text{like}})^{\alpha_2} \cdots$$

国内某短视频融分公式

国内某短视频APP的融分公式

- 根据预估时长 p_{time}，对 n 篇候选视频做排序。
- 如果某视频排名第 r_{time}，则它得分 $\frac{1}{r_{\text{time}}^{\alpha}+\beta}$。
- 对点击、点赞、转发、评论等预估分数做类似处理。
- 最终融合分数：

$$\frac{w_1}{r_{\text{time}}^{\alpha_1}+\beta_1}+\frac{w_2}{r_{\text{click}}^{\alpha_2}+\beta_2}+\frac{w_3}{r_{\text{like}}^{\alpha_3}+\beta_3}+\cdots$$

某电商融分公式

某电商的融分公式

- 电商的转化流程：

曝光 → 点击 → 加购物车 → 付款

- 模型预估：p_{click}、p_{cart}、p_{pay}。
- 最终融合分数：

$$p_{\text{click}}^{\alpha_1} \times p_{\text{cart}}^{\alpha_2} \times p_{\text{pay}}^{\alpha_3} \times \text{price}^{\alpha_4}$$

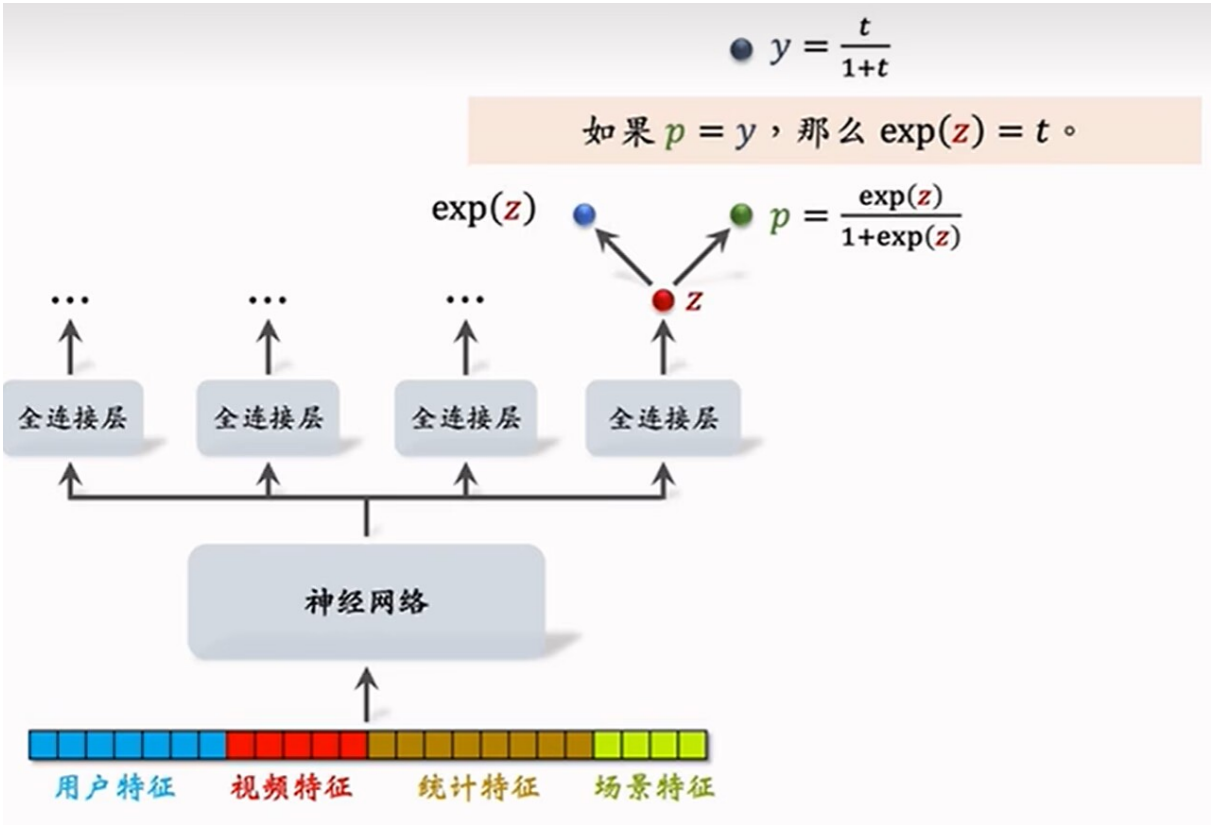
4、视频播放建模

图文笔记排序主要依据：点击、点赞、收藏、转发、评论.....

视频排序的依据还有播放时长和完播（看完即使没点赞，也能说明感兴趣）

直接用回归拟合播放时长效果不好

视频播放时长



用z反映播放时长，p是z经过sigmoid函数得到的值，y中的t是用户的真实播放时间（用户没点击视频t=0），用来反映用户对视频的真实观看情况

p与y的交叉熵 $CE = y \cdot logp + (1-y) \cdot log(1-p)$，训练过程中要优化CE，使p接近于y

若p=y，则exp(z)就是播放时长，后续推理用exp(z)来预估时长t，把exp(z)作为融分公式中的一项

视频完播率

衡量完播的方法

1、回归方法

如：视频长度10分钟，实际播放4分钟，则实际播放率y=0.4

预估播放率p拟合y：

$$loss = y \cdot logp + (1-y) \cdot log(1-p)$$

线上预估完播率，模型输出p=0.73，意思就是预计播放视频的73%

2、二元分类方法

定义完播指标：例如完播80%

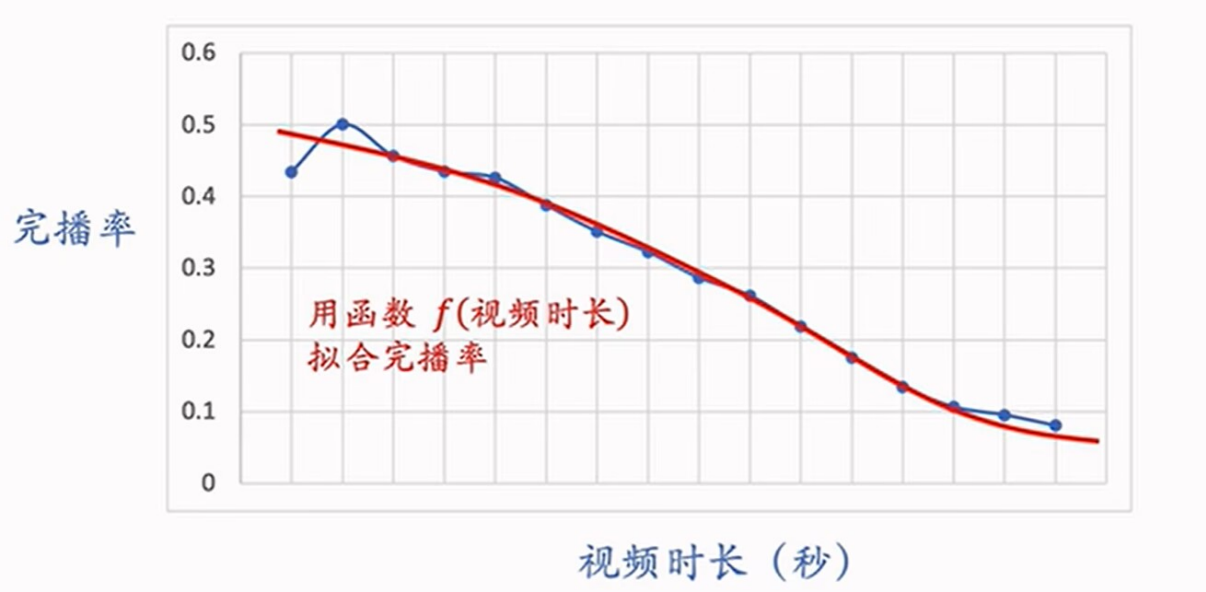
例如：视频长度10分钟，播放＞8分钟作为正样本，播放＜8分钟作为负样本

做二元分类训练模型：播放＞80%vs播放＜80%

线上预估完播率，模型输出p=0.73，代表P(播放＞80%)=0.73

完播率的调整

不能把预估的完播率用到融分公式——否则有利于短视频（播放率更容易高），对长视频不公平



调整步骤：

线上预估完播率，然后做调整：p_{finish} = 预估播放率/f(视频长度)，把 p_{finish} 作为融分公式中的一项参与排序

5、排序模型的特征

特征类别

前四个特征会存储到数据库中，线上服务时排序服务器会从数据库中取出数据，并处理数据作为特征给模型，模型就能预估出指标

用户画像（User Profile）

用户ID（在召回、排序中做embedding）

人口统计学属性：性别、年龄

账号信息：新老、活跃度

感兴趣的类目、关键词、品牌等

物品画像（Item Profile）

物品ID（在召回、排序中做embedding）

发布时间（或年龄）

GeoHash（经纬度编码）、所在城市

标题、类目、关键词、品牌.....

字数、图片数、视频清晰度、标签数.....

内容信息量、图片美学.....（算法打的分数，涉及CV、nlp对模型学习）

用户统计特征

用户最近30天（7天、1天、1小时）的曝光数、点击数、点赞数、收藏数.....

按照笔记图文/视频分桶（最近7天用户对图文笔记的点击率/对视频笔记的点击率）

按照笔记类目分桶（最近30天用户对美妆笔记的点击率/对美食笔记的点击率/对科技数码笔记的点击率.....）

笔记统计特征

笔记最近30天（7天、1天、1小时）的曝光数、点击数、点赞数、收藏数.....

按照用户性别分桶、按照用户年龄分桶.....

作者特征：发布笔记数、粉丝数、消费指标（曝光数、点击数、点赞数、收藏数）

场景特征（context）

用户定位GeoHash（经纬度编码）、城市

当前时刻（分段，做embedding）

是否是周末，是否是节假日

手机品牌、手机型号、操作系统（安卓和苹果用户的点击率、点赞率等指标差异非常显著）

特征处理

离散特征

方式：做embedding

用户ID、笔记ID、作者ID（容量巨大，消耗巨大）

类目、关键词、城市、手机品牌（容量相对较小，消耗较少）

连续特征

1、做分桶，变成离散特征

年龄、笔记字数、视频长度（把它们变成年龄段、字数范围段、视频时长范围段）

2、其他变换

针对曝光数、点击数、点赞数等数值：

（1）做log(1+x)，否则数量大得离谱，在训练和推理时会出现异常

（2）转化为点击率、点赞率等值，并作平滑（去掉偶然性的波动）

特征覆盖率

很多特征无法覆盖100%样本

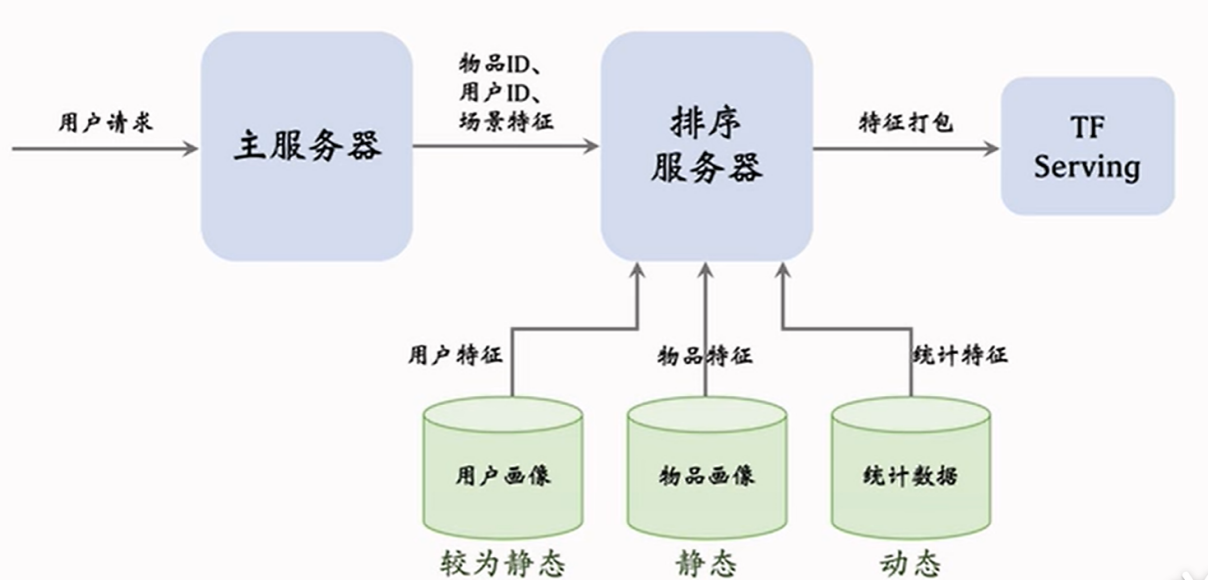
例如：

（1）很多用户不填年龄，因此用户年龄特征覆盖率远小于100%

（2）很多用户设置隐私权限，APP不能获得用户地理定位，因此场景特征有缺失

提高特征覆盖率，可以让精排模型更准（需要考虑特征缺失时把什么作为特征的默认值）

数据服务



TF Serving对排序服务器打包的物品进行打分，然后返回给排序服务器，最后通过筛选又返回给主服务器

6、粗排

粗排和精排的区别

粗排	精排
• 给几千篇笔记打分。	• 给几百篇笔记打分。
• 单次推理代价必须小。	• 单次推理代价很大。
• 预估的准确性不高。	• 预估的准确性更高。

本章之前用到的用户特征、物品特征、统计特征、场景特征串接输入神经网络，再分别进入全连接层输出点击率、点赞率、收藏率、转发率的模型是针对精排

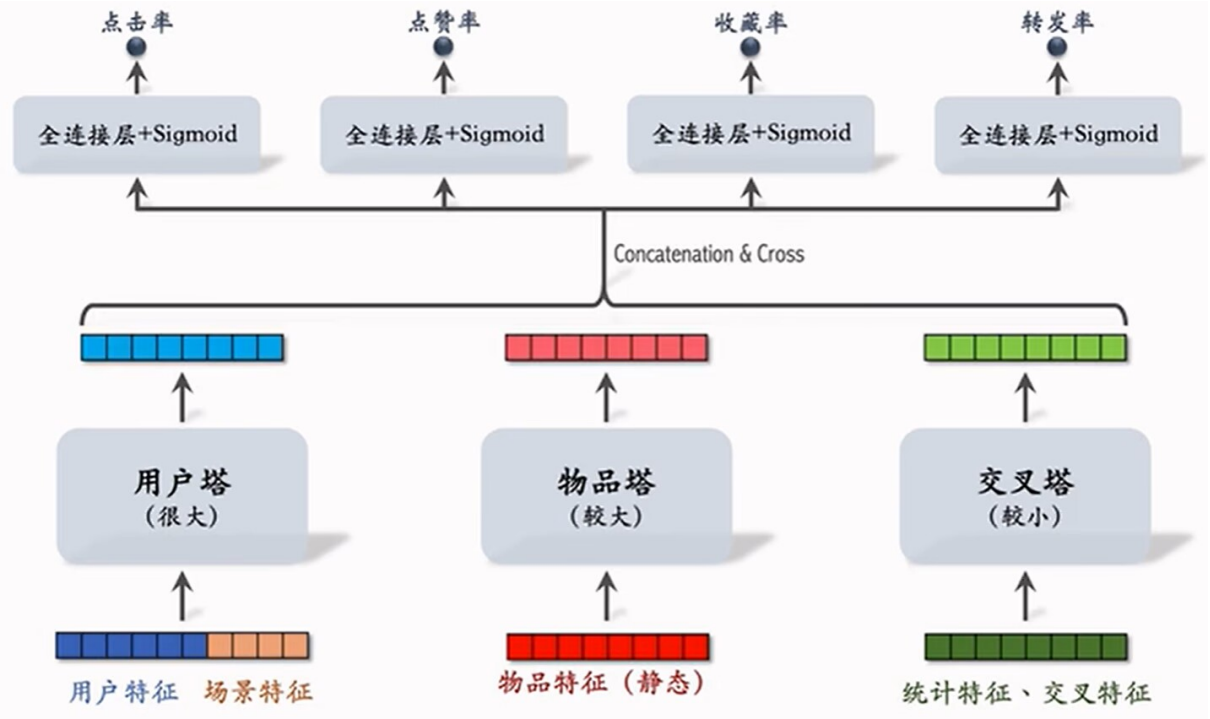
这属于**前期融合**：先对所有特征进行concatenation，再输入神经网络，这样线上推理的代价很大：如果有n篇候选笔记，整个大模型要做n次推理

双塔模型属于**后期融合**：它在线上计算时只对用户塔进行推理，物品塔中的向量存在数据库中（线上不做推理），代价很小；双塔模型把用户、物品特征先分别输入不同的神经网络，不对用户、物品特征做融合，在神经网络输出后再融合

*后期融合的准确率不如前期融合，因此前期融合用于精排，后期融合用于召回

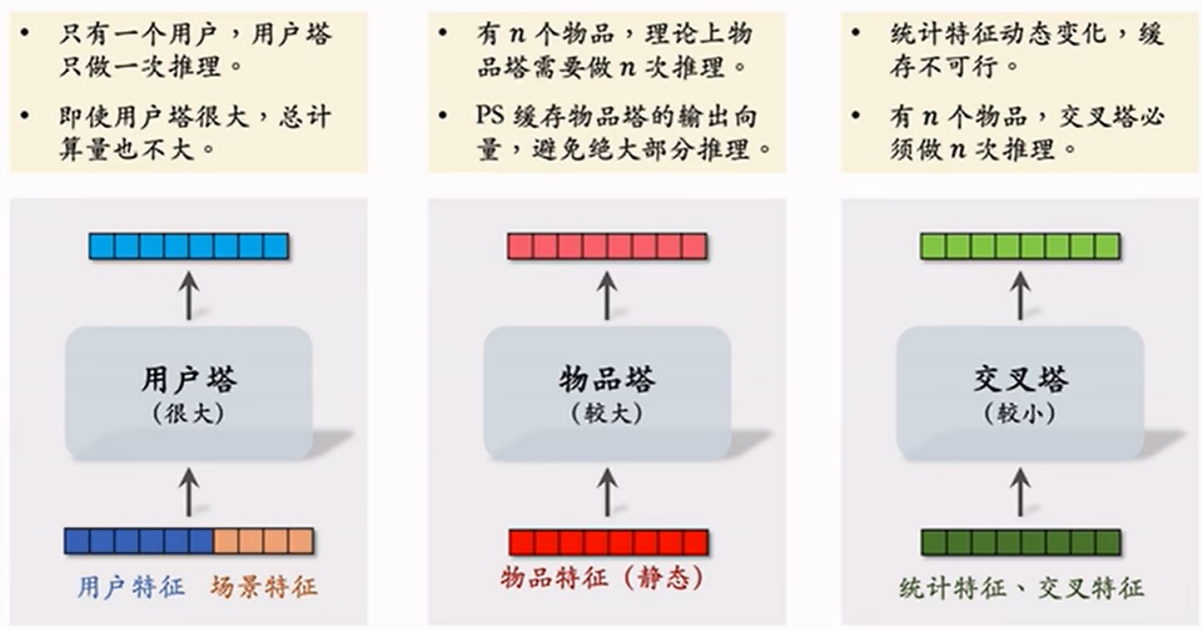
粗排的三塔模型

整体结构

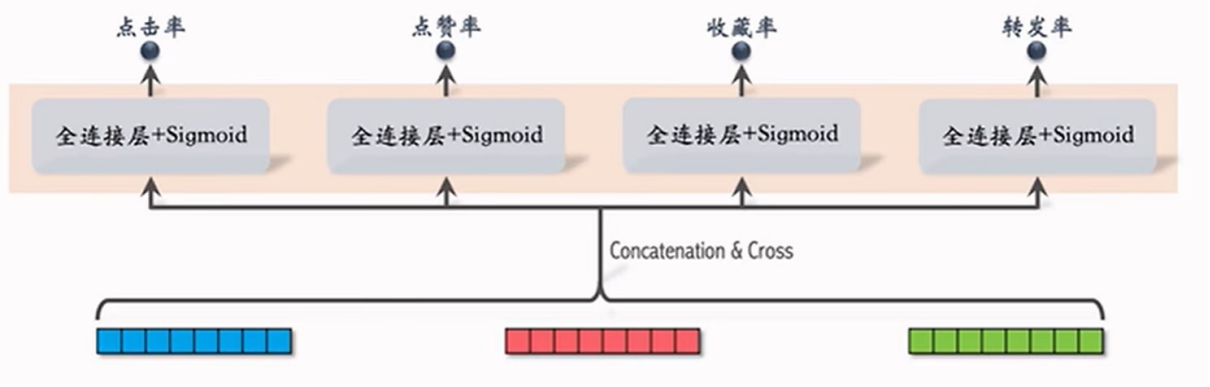


*三塔模型介于前期融合和后期融合之间，目的是减少计算量，使模型给几千篇笔记打分

下层



上层



有n个物品，模型上层就需要做n次推理，粗排推理的大部分计算量在模型上层（比交叉塔n次推理代价大）