# 京东电商推荐系统实践

## 简介

协同过滤示意图

最经典的推荐系统就是协同过滤。协同过滤主要分为两种：user-based（基于用户的协同过滤）和item-based（基于商品的协同过滤）。

什么是协同过滤？

但是，现在绝大多数推荐系统都不会直接使用协同过滤来做推荐。目前主要用的learning to rank框架。

协同过滤直接用作推荐系统的弊端？

推荐系统框架示意图

整个推荐系统的功能可以分为两部分：在线部分和离线部分。

* 在线部分主要负责当用户访问时，如何把结果拼装好，然后返回给用户。主要模块有：“召回”、“排序”和“对结果的调整”。
* 离线部分主要是对用户日志的数据分析，应用于线上。

和新闻、视频这类的内容推荐相比，电商推荐系统又有一些特殊的地方，比如：

优化方向

电商推荐结果需要综合更多方面因素考虑，比如点击、销售额、停留时长、用户留存等指标。

推荐内容

电商推荐的内容会有很多种，尤其像是活动类（促销等）的内容，很多推荐结果都是算法和人工运营共同作用完成的。

电商推荐系统在线部分包括的主要模块：

召回

召回会有很多种方法，如协同过滤、热门商品、实时促销等和应用场景相关的召回，还有一些基于KNN的召回。

过滤

召回之后，会进行过滤，主要是和应用场景相关，如已购商品过滤掉、没有库存的过滤掉、或者敏感的商品过滤掉等等这些逻辑。

排序

排序目前主要用到的是DNN模型，某些流量比较小的地方会用到GBDT。

过滤

排序之后还会有些分页、同商品过滤等逻辑。

结果调整

最终调整过程中，主要有两部分逻辑，多样性和探索逻辑。

推荐系统线上部分所包含功能示意图

## 排序模块

### 模型结构

排序模型结构示意图

排序模型除了比较火热的深度学习ranking模型结构外，还有一种最经典的模型，其中用到了很多id的Embedding，然后这些Embedding规模都很大，这样训练和上线都比较耗时。针对于此，京东电商推荐系统中做了一些优化，比如做分布式的训练，并且会有一套Pipeline来完成模型的上线。另外，虽然模型很复杂，并且能带来很好的效果，但是特征工程还是必不可少的，很多指标的提升还是依赖于特征工程，当然也包括一些模型调整的工作。

### 实践

排序模型的上线过程：

首先最重要的部分是模型训练平台和排序服务，因为很多深度模型计算量要求很高，为了能达到比较快的效果，需要部署单独的排序服务。目前比较流行的是TensorFlow serving，可以很快速的来上线一个深度模型，并充分利用对分片、单机并行，达到很高的计算效率。

模型线上线下一致性问题对于模型效果非常重要，我们使用特征日志来实时记录特征，保证特征的一致性。这样离线处理的时候会把实时的用户反馈和特征日志做一个结合生成训练样本，然后更新到模型训练平台上，平台更新之后再推送到线上，这样整个排序形成了一个闭环。

排序模型生效流程示意图

## 实时更新

我们的特征和模型都需要做实时的更新。因为我们经常需要很快地catch一些实时的信号，比如需要实时的用户画像来抓住实时的用户兴趣的变化，还比如需要抓住实时的商品画像，因为经常会有一些活动或者爆品，我们需要快速地捕捉这些信号，并应用到推荐中。另外还有一些实时地召回和特征，比如一些交叉的特征、实时的点击率、实时订单等特征。

除了特征外，模型也需要实时更新，对于电商场景来说这是有一定困难的，因为订单是有延时的，延时可能是十几分钟到十几小时不等，这样实时的模型更新上就会采取一些保守的策略，比如用点击率对模型做些微调，然后订单数据再通过离线来获得，这属于业务场景的限制。

排序中存在的问题及解决方案

排序问题示意图

排序可以算是推荐系统中比较重要的一个环节，但是只有排序肯定是不够的，事实上，有一些问题是目前的排序框架无法解决的：

排序得到的结果非常相似，影响体验。

有多个优化目标，需要一个平衡（点击率、订单金额、用户交互时长等）。

计算能力有限，如果有无限的计算力，可以直接对全部候选集进行排序。

针对以上排序无法解决的问题，有如下解决方案：

多样性。

使用模型输出的结果一般都会非常相似，如果直接给用户看体验会很差，因此在模型之后我们需要加入多样性的逻辑。

比较通用的解决办法是多样性的ranking，这是一个贪心算法，从第一个商品开始选，当选第二个商品的时候，会重新计算下候选集中每个商品的score，然后选择一个score最高的。京东的方法是看novelty score候选商品的产品词分布和之前N个商品的产品词分布的KL距离。这样做的思路，就是选一个和已有商品最不像的商品，来更好地保证商品推荐结果的多样性。

由于纯基于算法的多样性可能会出现badcase，因此还需要一个规则来进行兜底，确保在极端情况下结果也能接受。

排序多样性问题解决示意图

多目标。

排序多目标问题优化示意图

我们的优化目标有很多，比如点击、转化、时长等，问题会变得比较复杂，单一的模型训练很难覆盖到所有指标。另外，经常我们需要在各个指标之间进行权衡，因此可调试性也非常重要。

一种很有用的方式是多模型ranking，然后用某种方式把所有模型的结果combine。

这也体现了一个思想，在算法的实际应用中，其实需要在算法先进性和系统可维护性、可调试性之间做一个平衡。往往paper里很有创意的算法落地的时候是有些困难的。

多轮排序。

## 召回和首轮排序

## 实验平台