推荐系统架构 – 如何设计一个推荐系统

在开始学习推荐系统的时候，相信大家都会很好奇它的架构是怎样的。作为一个较为复杂的软件系统，推荐系统通常会由诸多模块和子系统构成。其中的每一个模块都在整个推荐过程中起着不可或缺的作用，只有深入了解了其中的每一个模块，才能对推荐系统整体有一个较为完善的把控，从而学好推荐系统这门课程。

当然在实际工作中每个系统的组成都千差万别，本节课会把最经典、也是最核心的模块介绍给大家，之后在工作中希望同学们可以做到灵活变通

下面让我们从数据流的角度，学习一下推荐系统的各个经典模块：

一、数据采集

俗话说“巧妇难为无米之炊”，数据采集作为推荐系统的第一个模块，充当着整个系统信息输入源的作用。

对于数据采集模块，最重要的职责就是汇总各方数据源，对它们进行清洗、预处理，为后续的分析和推荐做准备。以电商推荐系统为例，通常会有2大类的数据源：业务数据和用户行为数据。

业务数据指的是和平台商业逻辑或者产品功能直接相关的数据，比如商品的基本信息，订单信息以及用户间的关注信息等等。业务数据一般而言都可以从产品中的业务数据库如MySQL、MongoDB中直接获取到，因此推荐系统想要使用这一部分数据，基本上也只需要搭建一个ETL流水线即可。

相对于业务数据，用户行为数据是另一部分十分重要的推荐数据源。通常而言行为数据是通过客户端如手机app或者网页h5埋点上报所获得的。这类数据由于和产品业务并无直接关系，因此并不会直接存入MySQL等业务数据库中。在业界比较常见的方法是通过Kafka等消息队列，异步地将客户端上报的数据存入离线存储数仓内（比如Cassandra或者HBase）。在需要使用的时候，通过Spark等大数据处理框架将数据读取出来然后进行计算。

二、数据存储

在采集到了所需的数据之后，下一步就是要把数据存储起来了。对于推荐系统而言，通常会使用到以下几类的数据存储工具：

1. 关系型SQL数据库
   SQL数据库由于其自身的特点，非常适合于存储结构化的数据。在推荐系统中，常用SQL数据库存储用户或物品的元信息。比如需要知道某个用户的年龄、地域，或者需要知道某个商品的价格区间，历史评价时，往往会从SQL数据库中读出相应数据。

2. 非关系型分布式存储
   现代的推荐系统是一门大数据科学，因此通常在背后都有着数以百万、千万的用户行为数据。这些数据往往并非结构化，而且对于他们的访问延迟和一致性要求较低。所以可以通过Cassandra或HBase等分布式工具来存储。

3. 存储中间件
   除了以上两种存储类型，在系统内往往还需要一些存储中间件。比如会使用Redis来存储系统的配置，或者最常用到的用户、物品的热信息。再比如还会用AWS S3，阿里云OSS等静态对象存储来保存训练好的模型。

三、算法召回

推荐系统本质上就是信息的筛选系统，通过帮用户快速的从数以百万计的物品中找出他最感兴趣几个从而完成推荐工作。而召回层就是这个大筛选器的第一道“筛子”。

召回的目的首先是完成一次初步过滤，把原本几百万的候选集根据用户喜好筛选到几百或者一千左右的规模。
而召回的另一个重要目的是在于处理冷启动。由于系统内总会有一些新用户或者新物品，我们对它知之甚少，无法很好地按照传统大数据的思路进行推荐。这时候召回就需要对它们做一次“服务降级”：舍弃高精度的算法，退而使用更保守、更基础的算法。先保证有物可推，再保证推荐的质量。

在具体算法上，协同过滤可谓是最经典的算法了。无论是基于User-Based的还是Item-Based的协同过滤都能很有效的达到候选集初筛的效果。

为了达到第二个目的，同时也为了保证召回结果的多样性。通常在这一层会使用“多路召回”，也就是同时采用多种独立的召回策略。比如热门召回，用户画像召回等策略就是工业级十分常用的多路召回策略之一。

四、结果排序

顾名思义，排序层就是在召回返回的较小规模的候选集上再进行一轮精确的排序。从而确定最终展现给用户的结果顺序是怎样的。

刚刚说了，召回层的目的是初筛，而且一定程度上希望保证结果的多样性。这就势必会召回一些用户并不真正喜欢的物品，排序层首先就是要去除掉这一部分结果。

其次，由于用户的兴趣时段非常有限，通常一次推荐的二三十个结果真正被用户看到的也就是前十个、甚至前五个结果，因此排序的更重要的作用就是将最有可能被用户喜欢的物品尽可能的排在前面。以电商为例：据我之前的工作经验，通常的手机app一屏大概可以放下5个商品，而用户一般只会往下划2-3页。因此排序结果的前10-15个物品是不是能有效捕获用户喜欢就显得至关重要。

由于其强大的拟合能力和灵活的结构，排序层模型近些年几乎被深度学习垄断。在传统MLP的基础上又发展出了如Wide&Deep, NeuralCF等各局特色的模型。

五、效果监控

以上几个模块皆是为了产出推荐结果，但是这个结果是不是真的被用户接受还需要实地验证才行。效果监控模块就是以线上真实用户和真实流量为依据，评判推荐系统的效果。

效果评估时一般会定义若干产品及商业指标作为监控对象（比如点击率，转化率，视频完播率等）。而为了科学有效的评估，AB测试也是必不可少的工具。

对于成熟的大公司通常会采用自研的效果评估平台，但是对于初创公司或者小团队而言，采用如Amplitude这种功能齐备、开箱即用的Saas产品也不失为一个很好的选择。

六、结果应用

在经历了召回排序之后，终于有了可以展示给用户的推荐结果。但是一个好的推荐系统显然并不仅仅在于“幕后”，“台前”的表现也很重要。这里需要结合产品场景，将推荐结果最有效的展示给用户，才能收获更好的效果。同学们要切记，一个产品的成功绝对不仅仅是一个子系统的功劳，要各个部门共同协助，才能真正的做好一个产品！