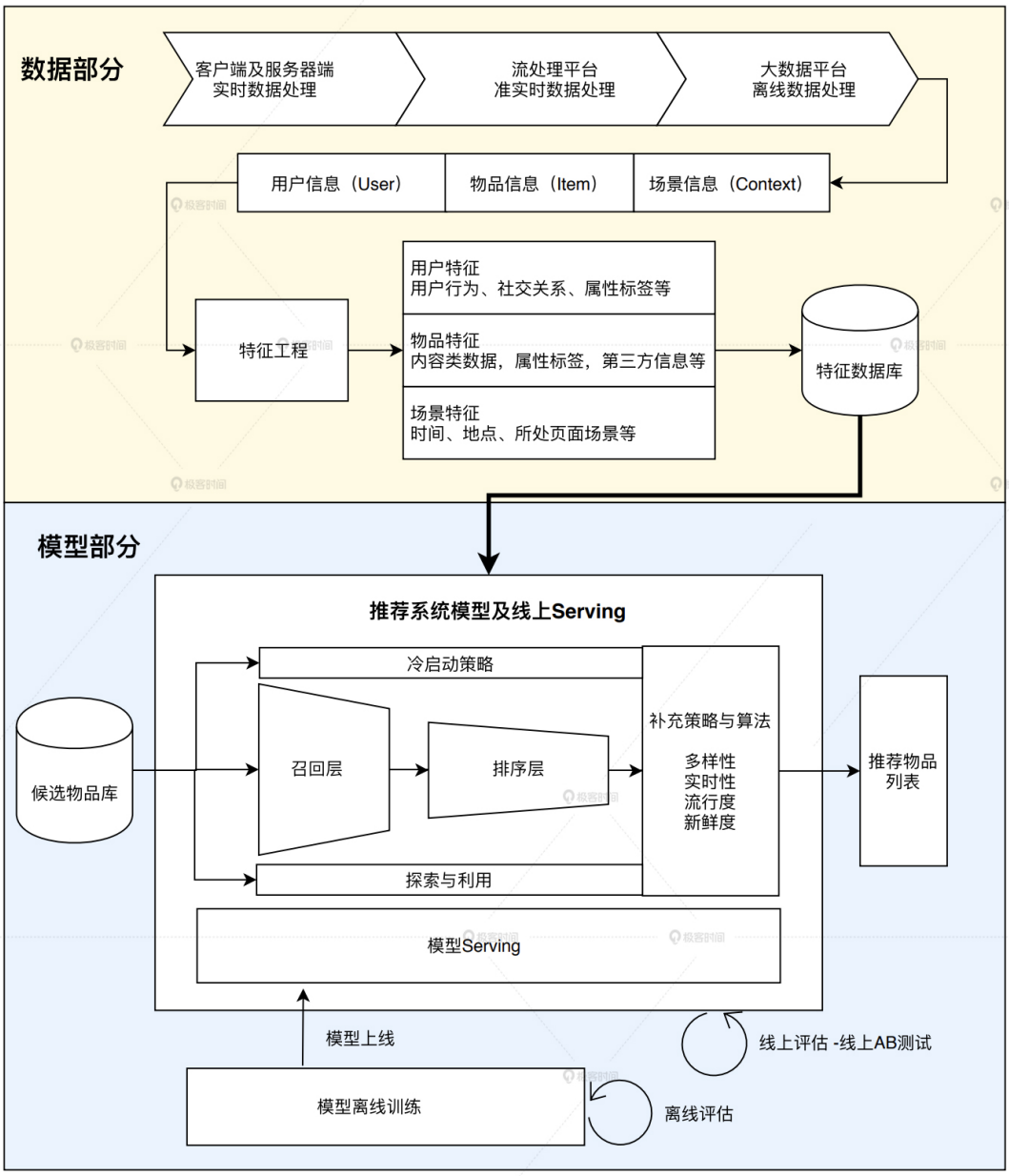
深度学习推荐系统的经典技术架构长啥样

推荐系统要解决的问题用一句话总结就是，在“信息过载”的情况下，用户如何高效获取感兴趣的信息。
从推荐系统的根本问题出发，我们可以清楚地知道，推荐系统要处理的其实是“人”和“信息”之间的关系问题。也就是基于“人”和“信息”，构建出一个找寻感兴趣信息的方法。

推荐系统要处理的问题就可以被形式化地定义为：对于某个用户U（User），在特定场景C（Context）下，针对海量的“物品”信息构建一个函数，预测用户对特定候选物品I（Item）的喜好程度，再根据喜好程度对所有候选物品进行排序，生成推荐列表的问题。

深度学习推荐系统的技术架构
一个工业级推荐系统的技术架构其实也是按照这两部分展开的，其中“数据和信息”部分逐渐发展为推荐系统中融合了数据离线批处理、实时流处理的数据流框架；“算法和模型”部分则进一步细化为推荐系统中，集训练（Training）、评估（Evaluation）、部署（Deployment）、线上推断（Online Inference）为一体的模型框架。推荐系统的技术架构图如下：



推荐系统的“数据部分”主要负责的是“用户”“物品”“场景”信息的收集与处理。按照实时性的强弱排序的话，它们依次是客户端与服务器端实时数据处理、流处理平台准实时数据处理、大数据平台离线数据处理。在实时性由强到弱递减的同时，三种平台的海量数据处理能力则由弱到强。比如使用 Spark 进行离线数据处理，使用 Flink 进行准实时数据处理等等。数据部分的主要作用有三个：

1，生成推荐系统模型所需的样本数据，用于算法模型的训练和评估。2，生成推荐系统模型服务（Model Serving）所需的“用户特征”，“物品特征”和一部分“场景特征”，用于推荐系统的线上推断。3，生成系统监控、商业智能（Business Intelligence，BI）系统所需的统计型数据。

推荐系统的模型部分

模型的结构一般由“召回层”、“排序层”以及“补充策略与算法层”组成。

1，“召回层”一般由高效的召回规则、算法或简单的模型组成，这让推荐系统能快速从海量的候选集中召回用户可能感兴趣的物品。“排序层”则是利用排序模型对初筛的候选集进行精排序。而“补充策略与算法层”，也被称为“再排序层”，是在返回给用户推荐列表之前，为兼顾结果的“多样性”“流行度”“新鲜度”等指标，结合一些补充的策略和算法对推荐列表进行一定的调整，最终形成用户可见的推荐列表。

2，从推荐系统模型接收到所有候选物品集，到最后产生推荐列表，这一过程一般叫做“模型服务过程”。为了生成模型服务过程所需的模型参数，我们需要通过模型训练（Model Training）确定模型结构、结构中不同参数权重的具体数值，以及模型相关算法和策略中的参数取值。

3，模型的训练方法根据环境的不同，可以分为“离线训练”和“在线更新”两部分。其中，离线训练的特点是可以利用全量样本和特征，使模型逼近全局最优点，而在线更新则可以准实时地“消化”新的数据样本，更快地反应新的数据变化趋势，满足模型实时性的需求。

形象点来说，你可以把这节课程的内容想象成是一颗知识树，它有根，有干、有枝、有叶，还有花。

其中，推荐系统的根就是推荐系统要解决的根本性问题：在“信息过载”情况下，用户怎么高效获取感兴趣的信息。而推荐系统的干就是推荐系统的逻辑架构：对于某个用户U（User），在特定场景C（Context）下，针对海量的“物品”构建一个函数，预测用户对特定候选物品I（Item）的喜好程度的过程。枝和叶就是推荐系统的各个技术模块，以及各模块的技术选型。技术模块撑起了推荐系统的技术架构，技术选型又让我们可以在技术架构上实现各种细节，开枝散叶。最后，深度学习在推荐系统的应用无疑是当前推荐系统技术架构上的明珠，它就像是这颗大树上开出的花，是最精彩的点睛之笔。