21天搭建推荐系统

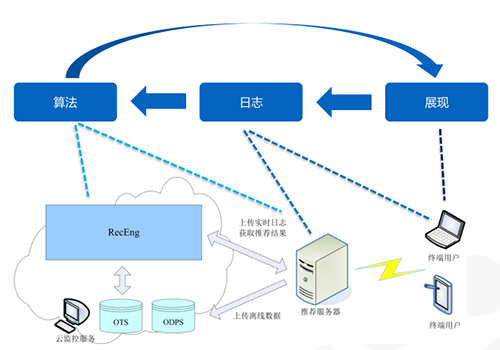
阿里云技术专家郑重

本文作者为阿里云技术专家郑重（卢梭），主要分享内容是如何在21天内快速搭建推荐系统。推荐系统的搭建是个复杂工程，涉及到实时计算、离线计算，以及各种数据采集、流转等，对自建推荐系统来说，1人年是跑不掉的。

　　本文介绍的内容还包括如何搭建一个个性化推荐系统所需的环境准备、基本配置和离线技术等基本功能的搭建，也有效果报表、算法优化和实时修正等高级功能的剖析。

　　大数据有三个非常经典的应用：计算广告、搜索、推荐。每一种应用最核心的地方都离不开三个字——个性化。广告不用说了，计算广告的基本要求就是要精准，为广告选择对其感兴趣的目标受众；搜索可以理解为对搜索关键词的个性化；而推荐，则需要在用户和物品之间建立兴趣关系。推荐的业态比较复杂，有类似淘宝天猫这样的真正意义上大数据场景，也有很多中小网站、应用，数据量其实并不是很大。[阿里云推荐引擎](https://data.aliyun.com/product/re)的初衷，是为了帮助阿里云的客户、创业者、中小网站，让他们能够更好的运营自己的产品或网站。

　　推荐系统一般包括展现子系统、日志子系统和算法子系统三个部分，三者互为一体。

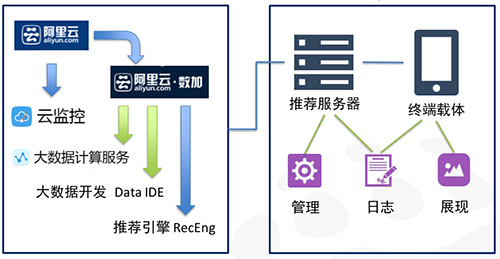


“展现”部分不仅要负担展现，还是数据采集的窗口，用户在展现系统的所有行为通过日志录入，采集到的数据经过算法子系统的计算，可以得到用户的偏好或者个性化兴趣，然后回过头来指导“展现”部分怎样做的更聚焦。

　　阿里云推荐引擎（RecEng）是推荐系统的一部分，主要实现的是算法子系统，需要和其他子系统配合工作。使用阿里云推荐引擎分为两大阶段

　　第一阶段：基本功能的搭建

　　Day1. 环境准备



　　环境准备分为两部分。图中左侧为云上资源的准备，我们需要拥有阿里公有云账号，然后开通云监控服务（可选）和阿里云数加服务（必选）；开通数加账号后，大数据计算服务（MaxCompute，原名ODPS）和大数据开发Data IDE就默认开通了（Data IDE相当于MaxCompute的可视化包装），最后开通推荐引擎。未来客户在推荐引擎中用到的数据，以及相关离线计算，都在客户自己的MaxCompute项目中完成。右侧为客户侧的准备，前端的展现，以及日志的采集和管理都需要客户自己完成，通过推荐引擎提供的API与推荐引擎进行交互。通常情况下，客户侧的后台相关功能会集中在推荐服务器中实现，这也是阿里云推荐引擎墙裂建议的方案。推荐服务器可以是客户自己的物理机，也可以是阿里云的虚拟机ECS，都是可以的。

　　Day2-3. 数据准备

　　DT时代的基本要求是数据要能够“存、通、用”。采集日志，并将其上传到公共云实现了数据“存”的过程；推荐引擎负责解决数据的“通”和“用”。“用”比较好理解，“通”则指的是所有进入推荐引擎的数据必须满足推荐引擎所定义的格式规范。推荐有三类数据：用户数据、物品数据和行为数据，我们定义了这三种表的格式规范，比较简单，具体细节可以[参考这里](https://help.aliyun.com/document_detail/shujia/RE/dataspec/datauploadspec.html)。

　　那么，如何把数据传到公共云上来呢？目前主要有两种方法，一是利用集成在MaxCompute console中的Tunnel命令，该命令的缺点只能上传文本格式数据；另一种方法是定制DataX上传，DataX作为连接各种数据库中间的节点，它除了可以作为文本上传，还可以把各种数据库打通。DataX的缺点是目前只能在Linux环境下运行。

　　当然，未必每一个业务的数据都满足规范的要求，所以还需要做一些格式转换。Data IDE提供了比较友好的格式转换界面，还可以把配置好的任务设置为定时任务，每天定时调度；也可以在MaxCompute console下直接执行格式转换的SQL脚本，再利用系统的crontab命令实现定时任务。

　　Day4-5. 基本配置和离线计算

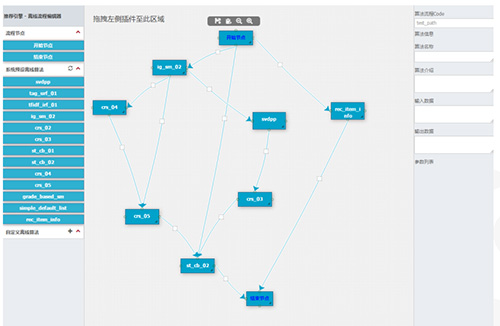


　　环境和数据都准备好了之后，接下来需要进入阿里云推荐引擎产品，真正开始使用推荐引擎了。不过在此之前，还需要对产品中的一些关键概念进行必要的说明。

　　第一个概念是业务。在阿里云推荐引擎中，业务指的是一组可被用来进行推荐算法计算的完备数据集，包括物品表、行为表、用户表这三张表。也可以简单的认为这三张表就构成了一个业务。

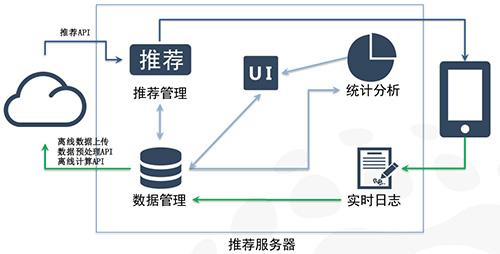
　　第二个概念是场景，所谓场景就是推荐的上下文。换句话说，就是在进行推荐时有哪些可用的参数。比如在进行首页推荐的时候，可用的参数只有用户的ID；在进行详情页推荐的时候，可用的参数除了用户ID，还可以由详情页上展示的物品ID，这样首页推荐和详情页推荐就是两个推荐的场景。一个业务可以包括多个场景。

　　第三个概念是算法流程，算法流程指的是数据端到端的处理流程，从客户的输入数据开始，到产出最终结果为止。推荐算法流程从属于场景，一个场景可以包含多个算法流程。每个推荐算法流程都包括两部分，离线计算流程和在线计算流程。离线计算流程负责从原始的业务数据（用户、物品、行为）开始，计算用户对物品的兴趣，输出本场景下用户可能会感兴趣的物品集合；在线计算流程实时接受推荐请求，从离线计算流程得到的物品集合中根据业务规则挑选出最合适的若干个物品返回给请求方。一个场景包含多个推荐算法流程这种设定使得我们在做效果对比变的比较容易，后面会介绍A/B Testing，在A/B Testing中，每个推荐算法流程都是一个可被效果指标度量的最小单元。在做完A/B Testing之后，通常只会在一个场景下保留一个效果最好的推荐算法流程。



　　产品里的配置都比较简单，配置业务基本信息、配置业务依赖的云资源、配置业务数据表，接着配置场景、配置API参数，最后配置算法流程，阿里云推荐引擎提供了两个默认的推荐算法流程模板，分别针对首页场景和详细页场景，图为首页场景的离线计算流程模板，图中每一个节点就是一个算法，最终产出离线计算结果。

　　Day6-8. 推荐API集成



　　到了这一步，云端推荐引擎里的推荐算法逻辑已经配置完成，剩下的事情就是把系统串起来，让推荐引擎和日志、展示两个子系统结合起来，成为推荐系统。阿里云推荐引擎提供了一组API，这里要做的就是把这些API集成到推荐服务器中。

　　首先需要把离线数据传上来，可以用前面提到的方法，Tunnel啊，DataX啊，都可以，但是一定要是定时任务，我们总不能每天都去手工执行数据上传。上传完成之后首先调用数据预处理API，对数据做一些预处理；然后调用离线计算API，启动离线计算。待离线计算完成后，通过推荐API就可以实时获取用户的推荐结果了。在离线计算的过程中，还可以通过查看计算任务状态API实时获取计算任务的状态，便于及时发现异常。

　　上图也展示了我们对推荐服务器的一些基本建议。诸如数据上传、启动离线计算这些功能建议由一个相对独立的数据管理组件来负责；而实时性要求比较高的推荐结果获取建议由专门的推荐管理组件来负责。推荐管理组件和数据管理组件为什么要有一个交互呢？这是因为从推荐引擎返回的结果中可能只包括了物品的ID，展示时不能只展示一个ID，还有很多材料，这些东西可以放在推荐服务器中，由数据管理模块负责管理。UI可以提供人工管理数据的界面，比如新录入了一个物品，或者某个物品卖完了要下线，需要做实时修正时就可以用到了。

　　这些工作都完成之后，一个具备最基本功能的推荐系统就可以运行起来了。