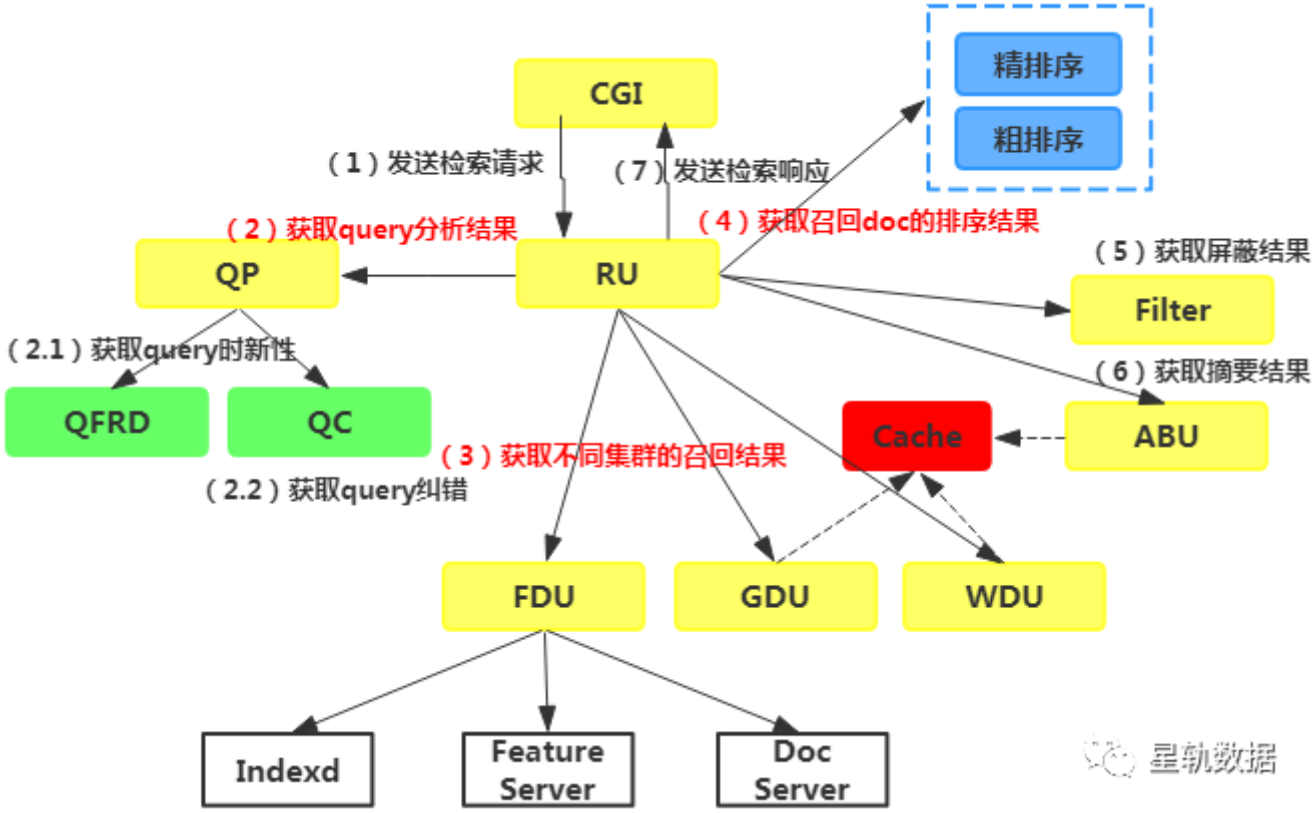
搜索系统的架构设计

原创 XG数据 WePlayData 2019-10-30

搜索系统由多个不同作用的模块组成，多个模块相互配合和兜底返回用户看到的结果。如何设计搜索系统的架构是个比较关键的问题。本文介绍一种工业界常见的搜索架构设计，其思路同样可用于推荐系统、计算广告中等任务中。因为这三类任务本质上都可以分为**召回+排序**两个过程。下图给出了在一次完整的搜索流程中，各个子模块是怎样相互调用和作用的：



首先是前端CGI负责接收用户的检索请求，发送至RU模块（ROOT UNIT）。RU是一个总控模块，负责和其它模块的调用交互。接着RU会请求QP（QUERY PROCESSER）模块获取query的分析结果，生成检索下发项（不止一个），也可以看成是确定检索策略的过程。在QP模块中又会分别去调用QC和QFRD模块去获取query的时新性和纠错结果。然后RU利用QP生成的检索策略去索引集群中求交召回，这里索引集群分为三种类型，分别是FDU（Fresh Doc Unit）存储一些比较新的文章，GDU存储一些优质的文章，WDU存储全量的文章。这样设计的原因是当全量文章比较多时会对检索造成比较大的压力，实际使用中优先召回GDU中的doc。每个索引集群下面又连着feature server（获取离线计算的doc特征，用于排序计算）和doc server（获取doc的正文）。RU获取召回结果后，会将召回doc送入到排序模块依次做粗排序（query-doc的单点排序）和精排序（query-doc list的整体排序）。后者相比于前者减少了候选doc的数量，可采用更为负责的特征。排序完成后，RU会请求Filter模块过滤一些非法信息。最后通过ABU获取摘要和飘红结果用于搜索结果的展示。其中cache模块的加入可以缩短命中cache的query检索耗时。