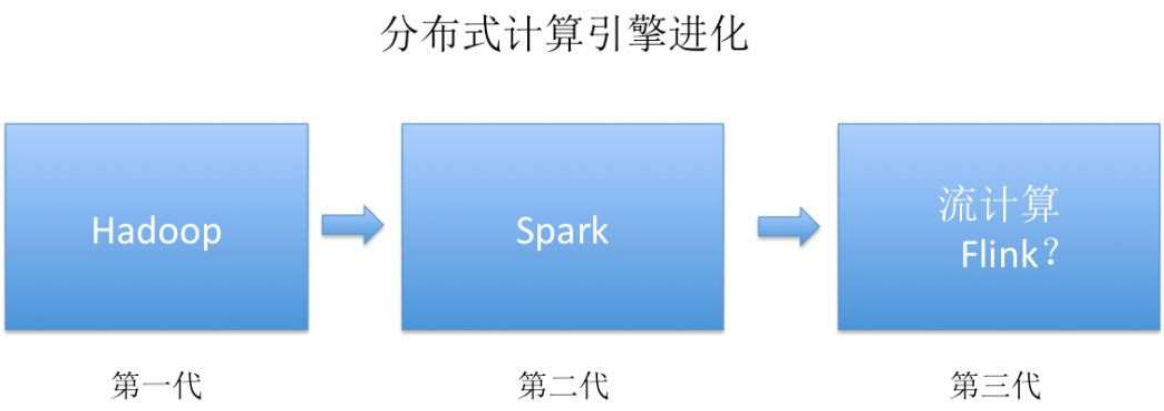
流式计算的认知

原创： 傲海 凡人机器学习

一

一些些背景

其实技术总在更新，做这个行业也是一直要走在学习并适应的路上，这也是人工智能领域最吸引我的地方，其实基础的理论是不变的，但是随着业务的发展，计算能力的发展，上层的实现总是在迭代，今天讲下我对于流计算的一些认知。



先聊下计算引擎的进化，随手画了上面的图。其实第一代分布式计算引擎是Hadoop，这是一个跨时代的创造，人们使用Hadoop的MapReduce框架实现了许多的算法，这些算法也发挥了很大的作用。Hadoop最大的特点是，数据计算依赖于硬盘存储，也就是说很多的计算过程中的结果都需要存在硬盘中，然后再从硬盘拉起，造成性能较低的问题。

Spark好在一点，把数据全部放到内存中进行计算，大大提高效率。但是无论是Spark或是Hadoop解决的都是批计算的问题，也称batch计算。离线计算需要把数据收集起来统一的去算，对于算法来讲，可能收敛会更快，因为参与计算的数据比较多。但是也有暴露一个问题，实时性很差。这个问题就引出了下一代计算引擎-流计算这样一个话题。

二

流计算

弄明白流计算，首先要搞清楚概念。先来看下流计算（stream compute）以及批计算（batch compute）的计算模型：

- 流计算：当一条数据被处理完成后，序列化到缓存中，然后立刻通过网络传输到下一个节点，由下一个节点继续处理。
- 批处理系统：当一条数据被处理完成后，序列化到缓存中，并不会立刻通过网络传输到下一个节点，当缓存写满，就持久化到本地硬盘上，当所有数据都被处理完成后，才开始将处理后的数据通过网络传输到下一个节点。

对于流计算，是不是有一点感觉了。相较于batch compute，stream compute对于业务上一定是更灵活，因为可以跟数据更实时性的相关联（数据的时间周期其实很重要，有机会我也会给大家分享我的看法）。

stream对于业务的优势我举一个例子，比如一个电商平台，有一个推荐系统，推荐模型都是每周根据离线数据做批训练生成的。但是突然有一天，这个电商搞了一个针对特殊人群的定向营销活动，有大量的特殊用户涌入，那针对这部分人群以前的老模型可能就不会起作用，这时候如果有一个实时训练模型的能力就会对这种场景有更快速地响应，这个就有是online learning的概念，那底层依赖的是流计算引擎。

三

真正的下一代流计算引擎

流计算引擎会是下一代的计算引擎，这里指的不是流计算替代批计算，而是下一代流计算引擎会兼容batch compute和stream compute，做到流批一体，Flink或许是一个答案。

当然流计算的挑战会比做批计算大很多，比如failover机制，批计算所有计算结果都是有存储的，可以回溯，流计算怎么解决宕机问题。比如exactly once机制，如何保证分布式流计算中的数据只被处理一次，而不是被多台机器多次处理。

不过还是详细，这些问题会被完美解决，未来的算法也一定是会向流式方向迁移。写了这么多作为开篇，接下来会分享一些流式算法的细节原理以及流式算法对于业务的影响，希望对大家有帮助。

写本系列文章的过程中，我自己也在不断学习和研究，也希望更多人可以留言讨论并输入更多地观点，欢迎分享转发，谢谢。