

开源大数据框架在海洋信息处理中的应用

樊路遥¹, 张晶¹, 陈小龙², 刘驰¹

1. 北京理工大学软件学院, 北京 100081
2. 海军航空工程学院电子信息工程系, 烟台 264001

摘要 随着中国海洋观测技术和数据收集手段的发展, 海洋观测数据量日益庞大、数据复杂程度逐渐增多, 传统的数据存储方式和处理方法无法有效地对海洋数据进行管理和分析, 同时, 数据的区域化和分布化成为海洋数据统一管理的一个障碍, 对海洋数据挖掘造成一定困难。本文从大数据生命周期的维度, 介绍了主流开源大数据框架在海洋大数据获取、存储、处理、智能分析、安全保护和治理方面的技术原理和应用。

关键词 开源; 大数据框架; 海洋信息处理

中国拥有面积达 300 万 km² 的海洋国土面积, 其中丰富的海洋资源不仅关系到国计民生, 更具有重要的经济、政治和军事意义。但是, 海洋观测非常复杂, 其数据来源包括但不限于雷达、图像、声呐、卫星等设备, 覆盖了水文、气象、物理等多个学科^[1]。海洋数据在实际中具有巨大的应用价值和意义。通过对海洋数据的分析, 可以获得附近海洋的自然环境状况、海洋资源和船舶活动情况等, 这些信息对中国海洋管理具有重要意义。然而, 因为海洋数据的多样性、复杂性和多源性, 无法使用传统数据库工具对其进行管理和处理。

近年来, 随着大数据技术的不断发展, 对海洋数据进行分析处理有了很大进步。和传统型数据库工具相比, 大数据技术和框架在处理海洋数据方面有处理速度快、规模大的优势。在海洋数据处理中引入大数据的理念, 不仅能更好实现对海洋数据的组织管理, 其意义还在于能够通过大数据分析与挖掘技术, 发掘出海洋数据的潜在应用价值。

本文按照大数据生命周期顺序, 依次对开源大数据技术下海洋信息的获取、存储、处理、智能分析和安全保护与治理技术进行详尽介绍, 并描述大数据技术在不同海洋信息场景中的应用。

1 大数据下的海洋信息获取技术

海洋信息的来源多种多样, 如利用卫星遥感技术获取的

遥感图、通过雷达探测获得的雷达数据文件、船舶航行产生的航海日志、海上观测站收集到的全天候水文气象信息等^[2]。如此繁多的信息包含了图像、数字、自然文字等多种多样的结构化和非结构化数据格式, 这些数据的存储工具也千差万别。来源广、格式多等原因导致获取到的信息数据无法直接通过大数据技术进行处理与分析, 因此需要通过一定的大数据数据获取工具对已有数据进行二次获取, 转换成可用的数据存储格式^[3]。

针对不同的数据格式与存储方式, 大数据技术中有着多种数据获取技术框架, 以下介绍几种主流的开源数据获取技术框架及其适用的海洋信息获取场景。需要注意的是, 这里的大数据获取不是前端传感器的获取, 而是针对多源异构大数据的融合及与后端存储与处理系统的衔接。

1.1 Apache Kafka

Apache Kafka 是一个分布式的发布-订阅消息系统, 最初由 LinkedIn 公司研发, 之后成为 Apache 软件顶级开源项目。Kafka 在各个大数据平台系统上使用十分广泛, 这是由于 Kafka 的消息队列机制使其在高吞吐量的实时数据获取上具有得天独厚的优势。Kafka 总体架构如图 1 所示。

如图 1 所示, Kafka 包含以下重要组件: 消息生产者 (Producer), 能够将源端 (Source) 产生的信息进行收集并作为消息发送到特定的一个或多个话题 (Topic) 中; 话题 (Topic) 以及

收稿日期: 2017-09-25; 修回日期: 2017-10-10

基金项目: 国家自然科学基金项目 (61772072, 61501487, U1633122); 国防科技项目基金 (2102024); 中国科协“青年人才托举工程”专项

作者简介: 樊路遥, 硕士研究生, 研究方向为大数据系统, 电子邮箱: fanluyao@foxmail.com; 刘驰 (通信作者), 教授, 研究方向为物联网和大数据技术, 电子邮箱: chiliu@bit.edu.cn

引用格式: 樊路遥, 张晶, 陈小龙, 等. 开源大数据框架在海洋信息处理中的应用[J]. 科技导报, 2017, 35(20): 126-133; doi: 10.3981/j.issn.1000-7857.2017.20.014

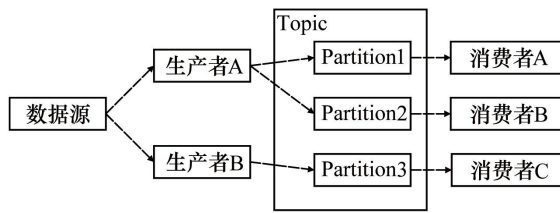


图1 Kafka架构示意

Fig. 1 Architectural diagram of Kafka

其中的区域(Partition)可以视为Kafka的分布式集群,Topic是作为区分不同种类消息的特定类型消息流,Partition则用来缓存传入其中的消息数据;消费者(Producer),可以从指定的一个或多个Topic中读取消息以便进行后续的处理,被消费的数据会被Kafka删除以便为新消息留出空间。消息队列机制本来是使用在系统不同模块间的通信方法。模块间在传递消息时并不需要建立一个专用的连接,而是通过写入或检索针对该模块的消息队列来进行通信,队列的使用避免了程序间对彼此的直接调用,除去了接收和发送应用程序同时执行的要求,实现了复杂系统不同模块间的有效隔离,防止了高并发环境下的堵塞问题。

Kafka的发布-订阅架构与消息队列机制使其可以很好地处理海量实时信息,既保证了生产-消费的有序性,同时也保证了集群的负载均衡性。在通信频繁与复杂的海上导航信息实时传输处理,以及持续收集海洋信息的海上观测站信息传输处理方面,Kafka能够发挥巨大作用。将海上导航产生的或者观测站收集到的各种信息,根据发布-订阅原则,按照各自的类型持续有序地存入指定的Topic,在处理端进行订阅接收后可对各种信息直接进行实时数据处理。Kafka目前已经在阿里巴巴集团的交易系统、搜狗公司商业平台等场景上得到充分应用,未来在海洋信息获取上的作用会愈发突出。

目前也有许多类似于Kafka这种基于消息队列机制、使用发布-订阅模式而被广泛使用的商用大数据获取工具。例如,MQTT(message queuing telemetry transport)是由国际商业机器公司(IBM)提出的一个基于消息队列机制的、使用发布-订阅模式的即时消息通信协议,如今已经成为结构化信息标准促进组织(OASIS)的规范。由于该协议的消息队列机制及简单的规范,因此在功耗较低、网络带宽有限的情况下十分适用,目前已经在遥感数据、智能家居、智慧城市、医疗医护等场景得到了广泛的应用。

MQTT工作流程主要分为客户端(Client)及服务端代理(Broker),两者通过网络进行连接。Client通过网络连接到Broker,可以发布产生的信息,也可以请求接受订阅的信息;Broker则负责接受、处理与转发Client端的信息。其中消息传递时的数据格式通过二进制位实现,并使用了UTF-8这一高效的Unicode字符编码格式。简单的总体结构、高效明确的数据格式,是MQTT在低功耗、有限网络带宽情况下能很好适用的巨大优势。考虑到远航船只传感器通信、无人深潜设

备控制等场景需满足低功耗、有限网络带宽等条件受限,可以将航船、深潜设备作为Client端,以MQTT协议作为消息规范,可以高效稳定地将产生的即时消息发送到Broker,Broker处理后再将消息返回。MQTT协议的使用会在这些情况下取得良好效果。

1.2 Apache Flume

Apache Flume最早是Cloudera公司提供的-一个分布式日志采集系统,现已成为Apache的顶级开源项目。它具有高可靠、高可用等特性,常用于大数据系统中的海量日志数据采集、聚合及传输任务,可将各个节点服务器中的数据收集起来,经简单处理后传输至指定地方,如送到Hadoop分布式文件系统(HDFS)或者HBase分布式数据库中存储起来。

Flume技术的核心就是1个Java进程——数据采集器(Agent)。Agent包含1个数据收集端(Source)、1个数据发送端(Sink)及1个中间的数据缓存端(Channel)。Source的作用是从指定位置读取持续增加的来源数据;Channel的作用是在Source把数据收集起来以后,将数据作为缓存存储在内存或中间文件中;Sink的作用是当Channel中缓存了一定量的数据后,一并发送到指定接收位置,发送成功后Channel才会将该批数据删除,若发送失败会进行重发尝试。在整个Flume的数据传输过程中,流动的其实是一个个带有头信息的字节数组(Event),数据从数据源进入Source被收集后,会被封装成一个Event,之后Sink端进行发送,最终在指定接收端将一系列Event转换为真实的数据形式。

由于Flume最初就是为收集持续产生的日志信息而设计的,其内部包含许多不同类型数据的库函数,拓展性很高,因此该技术框架完全适用于对远洋航行的船舶持续产生的航行信息或者航海日志收集任务等场景。可以通过在船舶系统或者海岸基站系统上启动一个专用的Flume进程,将持续产生的类日志信息收集并存储到指定的数据存储位置。同时最新的Flume-ng也在早先的Flume-og版本上进行了极大的优化,拓展性得到了提升。

1.3 Apache Sqoop

Apache Sqoop是一个大数据技术中常用的与传统数据库交互使用的数据库迁移工具,目前同样是Apache的顶级开源项目。Sqoop本身主要功能是将数据在大数据平台与传统数据库等其他存储平台间进行相互转移,并且其部署方便、使用简单、效率高、拓展性强,目前是所有大数据系统与传统系统数据库迁移工具的首选。

Sqoop的核心只包含一个实现不同平台数据转换的任务翻译器(Task Translator)。以数据在HDFS与传统关系型数据库Oracle为例,在Sqoop接收到数据迁移相关指令后,会与Hadoop中的MapReduce模块接口对接,转换成一个MapReduce任务,从而实现数据从Oracle到HDFS的导入或HDFS数据到Oracle的导出。对于目前已存储于传统数据库中的各类海洋信息数据,完全可以使用Sqoop这一迁移工具,轻松、高效地将历史数据分批转移到HDFS等大数据存储平台上,以

便后续对海洋信息数据进行处理与分析。

2 大数据下的海洋信息存储技术

从海洋中获得的数据,数量庞大而且非常复杂,因此普通的数据库无法进行有效存储和管理。下面介绍几种数据库,用于应对数据量大、数据源复杂的情况^[4]。

1) Apache HBase。Apache HBase 数据库是一个开源非关系式数据库。它是运行在 HDFS 文件系统之上、列式存储的分布式数据库,存储的数据类型是字节数组,可存储非结构化数据^[5]。HBase 的具体结构如图 2 所示。

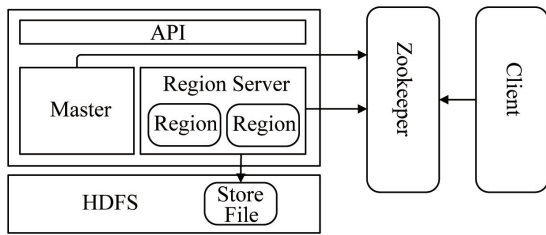


图 2 HBase 数据库结构
Fig. 2 HBase architecture

如图 2 所示, HBase 系统主要分为 3 个模块: Master、Region Server 和 Zookeeper。Master 用来协调多个 Region Server, 平衡 Region Server 之间的负载, 侦测它们之间的状态。在 Zookeeper 的辅助下, 允许多个 Master 节点共存, 但是只有 1 个 Master 提供服务, 其他节点处于待命状态。Region Server 的作用则是对表格的管理和实现读写的操作, Client 通过连接 Region Server 来获取到 HBase 中的数据。Zookeeper 是 HBase 中最关键的 1 个模块。它保证了至少有 1 个 Master 处于运行状态, 并负责 Region 和 Region Server 的注册^[6]。

当 1 个客户端访问 HBase 集群时, 先和 Zookeeper 通信, 然后找到 Region Server。每个 Region 都存储着数据, 多个 Region 同时存在提高了数据的并行性。当 Client 向 HBase 写入的数据超过 Region 存储能力的上限时, 会在底层文件系统生成 Store File, 并把数据转移到 Store File 中。此时, 数据就存储到了 HBase 的底层系统 HDFS 之中^[7]。

中国海洋面积广阔, 获得的数据量巨大, 至少达到 PB 级别。通过 HBase 可以很方便地对海洋中获得的非结构化数据进行存储, 例如图片、视频和音频信息等, 解决了传统型数据库存储能力小、无法存储非结构化数据和读写速度慢的问题。

2) MongoDB 文档存储数据库。MongoDB 是面向文档的数据库, 属于非关系型数据库。和普通关系型数据库相比, MongoDB 将以“row”存储的概念替换为“document”存储, 这样的存储方式只需要一条记录就可以记录非常复杂的数据层次关系。

MongoDB 面向 Collection, Collection 相当于关系型数据库的表。文档型数据具体存储的数据是“键值对”的集合。“键”是字符串, “值”可以为任意的数据类型。面向文档的数据模

式有很多优势, 例如, 在多台服务器之间可以自动分割数据, 可以平衡集群的数据和负载, 在处理数据量大的情况下, 如果插入新的数据, 只需要在集群中添加新的机器就可以继续处理接下来的数据。

在处理海洋大数据时, 由于数据的复杂性和数量庞大, 相比传统型数据库, MongoDB 在管理这类数据的存储时有很大优势。海洋的数据来源很多, 包括雷达、声呐、卫星影像等, 包含结构化、非结构化的数据, MongoDB 可以方便地存储。由于 MongoDB 的扩展性较强, 在海洋的数据量较大时, 通过扩展可以存储大量的数据。

3) Cassandra 数据库。Cassandra 是基于行存储的数据库, 它是由很多个节点构成的分布式数据库网络, 易于扩展, 存储的数据量大。相比传统型数据库, 它有突出的优势和特点: 模式灵活, 在系统运行的时候可以添加删除字段; 扩展性强, 易于拓展数据库的容量; 多数据备份, 如果一个数据节点丢失数据, 可以通过其他数据节点将记录恢复^[8]。

在 Cassandra 中, 各个节点相互交换信息。当客户端读取数据时, 与客户端连接的节点充当客户端和被请求数据节点的协调者, 假如请求的数据在 1 个节点中, 同时查询该节点及其节点副本, 根据时间戳判断哪个节点是最新的并返回, 最后输出到客户端。如果要写数据, 协调节点将写请求的节点发送到对应的节点。成功后, 每个节点的操作日志 commit log 捕获写操作来保持数据的持久性, 所有成功写入的内容将自动在集群中各个节点复制。

由于中国海洋幅员辽阔, 数据收集点较为分散, 当距离数据中心较远时, 大数据量的传输速度存在延迟, 并可能产生数据丢失。而 Cassandra 数据库则可以有效解决这个问题, 通过设置多数据中心, 即使某个节点数据丢失, 也可以恢复数据; 在获取数据时, 可以通过和自己距离最近的服务器节点来获得完整数据, 保证了传输的速度和质量。

4) Redis 基于内存的 Key-Value 数据库。Redis 是基于内存的高性能 Key-Value 存储的非结构化数据库, 支持多种数据类型存储, 包括 list、set、hash 等。为了保证效率, Redis 的数据都是缓存在内存中, 它可以周期性地把数据写入磁盘^[9]。具体结构如图 3 所示。

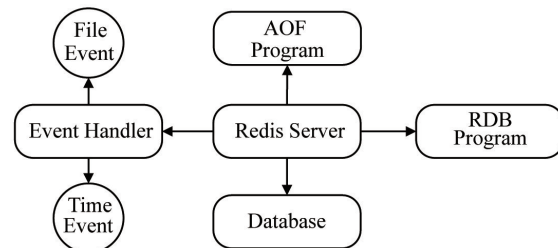


图 3 Redis 体系结构
Fig. 3 Redis architecture

从图 3 可以看出, FileEvent 处理和文件有关的事件, 接受客户端发来的命令, 并将命令的执行结果发送到客户端。Ti-

meEvent 处理和时间有关的事件,包括数据定期持久化、清理过期数据、附属节点同步等问题。AOF 模块负责命令日志的持久化,RDB 负责缓存数据的持久化。同样,Redis 也可通过部署集群来提高数据的读写速度和吞吐量,提高性能。

Redis 数据库的读写性能优异,海洋大数据获取可以实时地读取和写入。例如,实时定位监控船舶的位置,需要对卫星的数据进行实时读写,此时使用 Redis 系统就可大大提高系统的读写速度。如果处理的海洋数据量较大时,可以作为缓存,结合 Spark 等其他大数据处理工具结合使用以提高读写速度^[10]。

5) Neo4j 图形数据库。在 Neo4j 图形数据库中存储的是数据节点和数据连接节点之间的关系。相比传统型数据库的多表关联查询,Neo4j 通过简单查找就可以表示多个节点之间的关系^[11]。

中国海洋幅员辽阔,数据源节点多,使用关系型数据库无法将多个数据节点之间的关系特别是数据源之间的内生关联展示出来。在 Neo4j 中,通过设置每个数据源的元数据,并定义每两个数据节点之间的相似性关系,可以将多个数据节点联系起来。这种关系对很多海洋领域的应用具有重要意义,如海洋天气环境的观察和预测就需要利用多维异构传感器的信息。但是 Neo4j 只有在查询节点之间的关系时才会有比较好的性能和效果,在修改较多、查询较少、节点边数较多的情况下性能会下降。

3 大数据下的海洋信息处理技术

实现海洋信息在大数据技术下的存储后,相应的开源大数据处理技术可以对这些海量数据进行各种形式的处理,挖掘海洋信息所包含的巨大潜在价值,获得所需要的信息。针对不同来源、不同存储方式的数据,所使用的大数据相关技术也不尽相同,例如对于实时性要求不高的数据可以使用 Hadoop MapReduce 进行批处理,使用 Spark 进行基于内存计算的数据并行处理;而对于实时性要求高、持续不断增长的数据可以使用 Storm、Samza 等框架进行流数据处理。以下介绍 3 种主流处理方式的开源技术框架及其可适用的海洋信息处理场景。

3.1 Hadoop MapReduce

Hadoop 是 Apache 开发的一个分布式系统基础架构,也是最早被用来在分布式集群上对海量数据进行批处理的成熟架构。Hadoop 由诸多元素组成,其中包含 2 个核心部分: HDFS 分布式文件系统和 MapReduce 编程模型。HDFS 实现了海量数据的存储功能,其对于硬件要求较低,拓展性极高,可以通过部署在众多廉价机器上来搭建分布式集群,也可随时扩增集群规模;MapReduce 则在 HDFS 数据存储之上提供了一个基于大规模数据集的批处理编程模型,通过 Map(映射)与 Reduce(规约)的方式、分而治之的处理思想,实现对海量数据的批处理目的。MapReduce 的工作原理如图 4 所示。

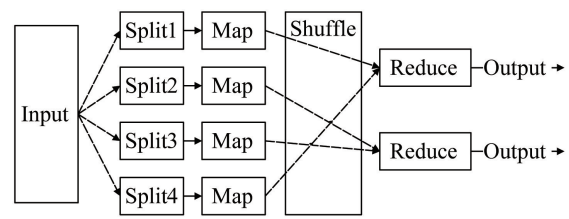


图 4 MapReduce 工作原理示意
Fig. 4 Working diagram of MapReduce

从图 4 可以看出,由于 Hadoop 本身是一个分布式集群,存在作为管理节点的 Master 节点及用于存储与计算的多个子节点(Slaves)。因此,其对于数据的存储与批处理技术不同于传统的数据处理方式。以计算海洋资源数量为例,Hadoop MapReduce 在处理该任务时,首先会对完整的海洋资源输入数据进行拆分,拆分成一个个小片(Split),并将每片分割成<key,value>对的形式,如:<0,一定量的海洋资源>。对拆分好的<key,value>使用 Map 方法处理。Map 方法处理完成后的输出也为<key,value>形式,区别在于此时输出为 Split 中各类海洋资源种类的数量,如:<1,“鲨鱼”>。Map 方法完成之后会按照 key 值进行结合(Combine)操作,将各 key 相同的 value 值累加,得到 Mapper 的最终输出结果。得到 Mapper 输出后,会进行一次 Shuffle 操作,将各 Mapper 输出分配到 Reducer,按照 Reduce 方法再对数据进行处理汇总,得到各类海洋资源的具体数量。

由于 Hadoop 是一个基础的开源大数据架构,几乎所有其他大数据技术皆搭建其上,因此在各系统中作为基础框架广泛使用。例如,美国第二大石油公司 Chevron 公司为便于其寻找油矿,使用 Hadoop 技术收集并处理了大量的海洋地震数据;创业公司 Skybox Imaging 也使用了 Hadoop 技术来存储处理海量的图片信息,从高清的卫星遥感图中探测地理的变化。可见,Hadoop 技术将越来越多地应用于海洋信息的分析之上。

3.2 Apache Spark

Apache Spark 是一个专为处理海量数据而设计的基于内存计算的并行处理引擎,诞生于美国加州大学伯克利分校 AMPLab,目前是 Apache 的顶级开源项目之一^[12]。Spark 与 Hadoop 的 MapReduce 引擎相比,最大优点是中间计算产生的输出保存在内存中,不用频繁对 HDFS 进行读写,节省了大量的 I/O 操作,更为高效。Spark 运行原理如图 5 所示。

如图 5 所示,Spark 核心设计是弹性分布式数据集(RDD)这一抽象数据结构,在 Spark 整个处理中都是通过对一系列 RDD 进行相关操作而实现的。RDD 是一个只读的记录分区集合,只能通过在存储器或其他 RDD 执行相关操作后被创建,并且只会存在于计算时占用的内存之中。Spark 开始计算时,会首先绘制出 1 个计算路径的有向无环图(DAG)。DAG 中每 1 步操作都会生成 1 个新的 RDD,RDD 之间会连成

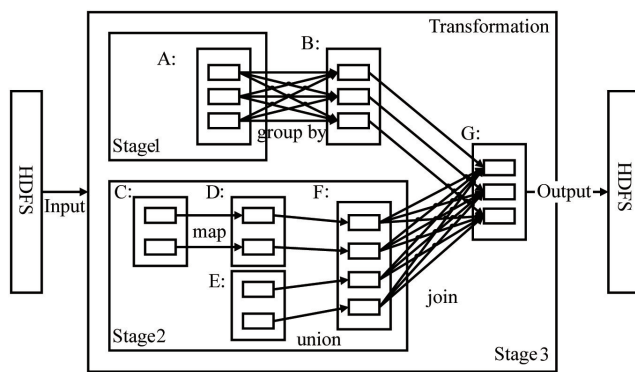


图5 Spark运行原理
Fig. 5 Spark operations

1条边,这些RDD与边便组成了该DAG图。形成了DAG图后,Spark内核接下来会根据DAG图将计算划分成一个个任务集,即Stage,这样便可以将任务提交到各分布式计算节点进行计算。划分Stage时会充分考虑在分布式计算中可流水线计算的部分来提高计算的效率,该过程中的主要根据是RDD的依赖类型。根据不同的转换操作,可以分为窄依赖和宽依赖两种类型。窄依赖指生成的RDD中每个Partition只依赖于父RDD固定的Partition;而宽依赖指生成的RDD中每个Partition都依赖于父RDD的所有Partition。宽依赖往往意味着分发操作,即Shuffle,这也是Spark划分Stage的主要边界。对于窄依赖,Spark会将其尽量划分在同一个Stage中,因为它们可以进行流水线计算。当DAG图被完整计算过后,便会产生最终的输出结果,一般会将该结果写回到HDFS等存储结构中。

Spark Streaming是基于Apache Spark核心的应用程序编程接口(API)构建的一套并发流处理库,它可以对实时数据进行流式处理。输入数据源既可以是Kafka、Flume等消息队列系统,也可以是传统的传输控制协议(TCP)套接字传输的原始数据。其工作机制为由Spark Streaming对实时的输入数据流进行接收,并将数据分割成批数据Batches供Spark核心系统处理,经Spark核心系统处理后,得到最终的流式批处理结果。Spark Streaming提供了一个名为离散流DStream的高级系统抽象,用于表示连续的数据流,在系统内部DStream同样由一系列RDD构成。

Spark目前已成为最流行的大数据并行计算处理框架。在所有需要大数据并行处理的海洋信息上,均可以根据处理的具体信息不同来设计不同的Spark程序,如对于存储在Hive等数据库上的海量信息,可以使用Spark SQL模块进行类数据库数据处理;对于存储于Neo4j图数据库上的海洋关系信息可以使用GraphX模块进行图数据处理;对于Kafka中的流数据信息,可以使用Spark Streaming模块进行处理。

3.3 Storm

Storm是一个分布式高容错的流数据处理平台。Storm与其他大数据技术不同之处在于,它不是像Hadoop一样针对固定数据进行批处理操作,而是专门针对持续流入的实时数据

进行实时流处理操作。该流式处理模式保证了每个任务只需在最初进行一次初始化操作,之后便会持续进行计算处理,每条流入数据都会被计算处理到,并输出需要的信息或写入到指定的存储器中。

Storm在正常运行时需要Zookeeper集群的支持,Zookeeper在这里可视为一个协调器,通过其独有的心跳机制与同步节点状态功能决定后面任务的分配。Storm中的Nimbus与Supervisor是服务端的守护进程,Nimbus负责整体的管理、协调并监控所有拓扑(Topology)的发布、任务指派及失败重分派等操作;Supervisor等待Nimbus分配任务后对其下属的Worker进行管理与故障监控;Worker是每个节点中一个或多个相互独立运行的Java虚拟机(JVM)进程;Executor是一个Worker中所运行的Java线程,一个Executor中可运行一个或多个指定的Task;Task包含Spout与Bolt,这两者组成了Topology,Tuple是Topology中处理的最小的消息单位。Spout组件可视为消息生产者、数据的输入源头,负责从异构的数据源读入数据并转换成消息流进行发送;Bolt组件是消息的消费者,接收从Spout发出的消息流,同时完成具体设定的处理逻辑,Bolt组件间可以进行串联,通过串联多个不同处理功能的Bolt实现整体复杂逻辑处理。

Storm目前已经被Twitter、Yahoo、阿里巴巴集团等互联网公司广泛使用。其强大的流数据处理功能与完善的系统架构,同样适用于对海洋观测站、气象遥感图等持续产生实时信息的场景。例如,Storm在接收到源源不断产生的实时数据后,对数据进行流处理操作,可以及时地将各类海洋信息转换成需要的信息,用作天气预报、水文观测、流量统计等用途。此外,还有其他一些类似于Storm的开源流处理框架,如Samza就是一个基于消息队列的分布式流数据处理框架,同样可以实现实时数据的流处理。

4 海洋智能信息分析与挖掘技术

随着人工智能技术的发展,越来越多的开源人工智能学习框架被提出和实现。通过人工智能学习算法,可以很好地对海洋大数据进行处理分析,挖掘数据的潜在价值^[13]。TensorFlow、Caffe2、CNTK和Paddle是当前较流行的机器学习框架。

4.1 4种常见开源机器学习框架

1) TensorFlow。TensorFlow是谷歌公司开源的机器学习框架,包含多个机器学习算法,在图像、音频、推荐系统场景下都有丰富的作用。TensorFlow提供GPU加速的功能,在处理数据量大时,可以辅助算法的执行进行加速^[14]。

使用TensorFlow框架,可以对海洋中获得的数据进行智能分析处理。例如使用预测算法对海洋中天气进行预测^[15],使用神经网络算法对卫星图像进行分析等。这些分析处理挖掘出数据的潜在意义,对海洋的管理有重要意义。

2) Caffe2。Caffe2是Facebook公司发布的开源深度学习框架,它是在Caffe框架基础上的重构和升级,集成了诸多深度学习算法,在保证运算性能和扩展性的基础上,加强了在

轻量级平台上部署的能力,支持多种移动平台。

相比其他机器学习框架,Caffe2可以在移动设备上多机部署。在海洋大数据处理方面,其数据来源较为分散、地域分布较广,因而部署大型计算设备对数据进行分析是不太现实的。移动端部署则为这种情况的解决提供了可能性,降低了硬件成本。在数据源周围的移动端部署Caffe2处理平台,可以实时获取数据并及时进行分析和挖掘,例如在海洋灾害预测等对数据实时性要求很高的应用场景^[15]。

3) CNTK。CNTK是微软公司开源的深度学习框架,具有较好的交互能力。其支持多种神经网络模型,将深层神经网络的学习解释为一系列的有向图计算。有向图的边表示矩阵运算,节点表示输入的值或者参数。通过拓展节点,用户可以轻松构建多种神经网络进行相关研究^[16]。

相比其他开源机器学习框架,CNTK支持在Windows系统下运行。同时,其性能在处理图像、语音识别方面较其他框架稍好。其次,使用也较为简单,只需要通过配置文件作为参数即可构建神经网络模型。但由于CNTK框架相对还没有成熟,相关文档还比较缺乏,代码也还在完善之中,在学术

界和工业界广泛应用尚需要一定时间。

4) Paddle。Paddle是百度公司研发的深度学习平台,同时支持海量数据训练、多机并行处理,并提供了图像分类、推荐、语义分析、情感分析等算法的接口,可以较为简单地调用。相比其他机器学习网络框架,Paddle采用基于操作(OP)和基于层(Layer)混合的系统,兼顾了灵活性和封装性,在性能上有较好的表现。

在海洋智能信息分析与挖掘方面,Paddle可以发挥其作用。例如,对海量卫星图像的识别、海洋灾害的预测、海洋资源的探测等^[17]。

4.2 4种常见开源机器学习框架比较

上述4种机器学习框架都包含了深度学习的神经网络算法,包括卷积神经网络、循环神经网络、前馈神经网络和反馈神经网络等。

Tensorflow相比其他深度学习框架,不仅实现了深度学习的神经网络模型,而且实现了其他机器学习算法,例如分类算法、聚类算法、逻辑回归等^[18]。表1比较了4种框架支持的算法模型。

表1 4种常见的开源机器学习框架对比

Table 1 Comparison of four open source machine learning platforms

框架\算法模型	循环神经网络 (RNN/LSTM)	卷积神经网络 (CNN)	前馈、反馈神经网络	分类算法	回归分析	聚类算法	贝叶斯网络
TensorFlow	√	√	√	√	√	√	√
Caffe2	√	√	√	×	√	×	×
CNTK	√	√	√	×	×	×	×
Paddle	√	√	√	×	√	×	×

注:“√”表示框架支持该算法模型,“×”表示框架不支持该算法模型。

5 大数据下的海洋信息治理技术

基于大数据生命周期的数据获取、存储、处理与分析技术虽然有很多应用前景,但由于海洋信息纷繁复杂,数据量庞大,数据格式的异构,数据来源多样,大数据治理问题同样重要^[19]。此外,中国海洋信息关系到国家安全,若信息遭到窃取,会造成难以估量的损失,大数据安全技术不容忽视。

5.1 大数据全生命周期管理

1) Apache Falcon。Apache Falcon是一个为Hadoop集群数据管理和数据生命周期管理提供服务的系统框架。它通过声明数据管理及处理方案,解决对Hadoop数据复制、业务连续及血统追踪等难题。Falcon主要体现在对数据生命周期的集中管理,促进了数据快速复制来实现业务连续性和灾难恢复,并通过实体沿袭追踪和审计日志收集为审计和合规性提供基础,方便用户设定数据管理及处理方案,并将其提交到Hadoop集群调度执行。

Falcon本质上是通过标准工作流引擎将用户的数据集及其流程配置转换成一系列重复的活动,本身不会做繁琐的工作。所有功能及工作流状态管理需求都是由内部配置好的

工作流调度器进行调度。由于其本身并没有对工作流做额外的工作,使得Falcon可以专注于保持数据流程实体之间的依赖和联系。使用Falcon建立工作流时完全不会感觉到Oozie调度器及其他基础组件存在,该设计理念使重心可以完全放在数据及处理本身,而不需要进行任何多余操作。

Falcon通过实体沿袭追踪和审计日志收集为审计和合规性提供基础,可以方便地设定数据管理以及处理方案,并将其提交到Hadoop集群调度执行。在海洋大数据处理方面,Falcon的使用可以完整地追踪。某一批数据最初是由某船只、观测站或者雷达产生的,产生后最初的存储位置及接下来在大数据处理技术中的流程流向,都可以清晰方便地做到追踪与管理。

2) Apache Atlas。Apache Atlas是一个可伸缩、可扩展的元数据管理与大数据治理服务。元数据是描述数据的数据,是指从信息资源中抽取出来用于描述其特征与内容的数据。Atlas设计的目的是为了与其他大数据组件交换元数据,改变以往标准各异的元数据管理方式,构建统一的元数据库与元数据定义标准,并可与大数据技术中其他各类组件

相集成,建立统一、高效、且可扩展的元数据管理平台。

在Atlas的架构中,其核心组件包括:类型系统(type system)、导入/导出、图引擎3部分。其中,类型系统是指Atlas中可根据自身需求来对元数据对象进行建模,该模型由被称为“类型”的概念组成,类型的实例被称为“实体”,实体能够呈现出元数据管理系统中实际元数据对象的具体内容。导入/导出:Atlas中的导入模块允许将元数据添加到Atlas中,导出模块将元数据的状态暴露出来,当状态发生改变时会生成相应可被下游消费者组件获取并消费的事件,从而实时地对元数据的改变作出响应。图引擎:Atlas内部使用图模型数据结构来表示元数据对象,其优势在于可以获得更好的灵活性,同时有利于在不同元数据对象之间建立丰富的关系。图引擎负责对类型系统中的类型和实体进行转换,并与底层图模型进行交互。除了管理图对象,图引擎也负责对元数据对象创建合适的索引,使元数据搜索变得更为高效。目前,Atlas使用Titan图数据库存储元数据对象。

海洋信息本身的复杂多样性导致不同种类的信息都会有各自的元数据标准,甚至不同的机构对同样的信息也会有不同的元数据标准,这会极大阻碍海洋信息的迁移使用,增加大数据技术在海洋领域新的尝试的工作量,浪费人力物力。而Atlas的提出与使用,通过定义统一的元数据标准,建立高效的元数据交换体系,提供友好的业务定义接口,可视化的血统查询显示与数据审计功能,在海洋信息大数据治理的实际应用中提供了有力的支持。

5.2 大数据安全

现有的大数据技术虽然本身也有一定安全保护机制,但多属于粗粒度控制,很容易出现数据泄露情况,因此需要其他安全管理组件完善^[20]。Apache Ranger是一个集中式的安全管理框架,提供集中管理的安全策略和监控用户访问,而且它可以对其他大数据技术组件进行细粒度的数据访问控制,并解决授权和审计。管理员可以轻松地通过配置策略来控制用户访问权限,同时Ranger还提供全面的日志审计,能够清晰看到请求信息和数据的权限状态。Ranger通过LDAP/AD(Lightweight Directory Access Protocol,轻量目录访问协议)支持同步用户和组的信息。图6展示了Ranger体系结构。

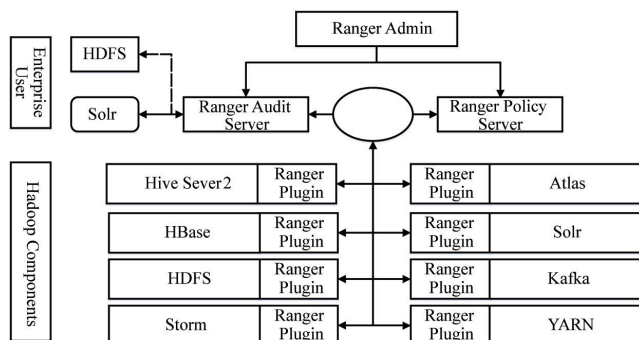


图6 Ranger体系结构

Fig. 6 Ranger architecture

从图6可以看出,Ranger安全认证机制整体框架主要包括Ranger Admin服务、Ranger Plugins和Ranger Usersync服务3部分。Ranger Admin服务是安全管理的核心接口,也是Ranger框架的管理中心,在服务提供的Web UI上可直接管理系统用户权限,创建和更新权限认证策略,然后将这些策略存储在配置的数据库中;Ranger Plugins组件是权限安全管理的核心,它是一个轻量级的可以嵌入在各个集群组件中的JAVA程序,实现了Ranger对各组件的直接权限管理控制;Ranger Usersync服务是一个非常重要的工具,它用于从Unix系统或LDAP中读取用户/用户组信息并同步到Ranger Admin中,起到了用户权限初始化与权限同步的作用^[20]。

当实现海洋大数据存储后,可通过Ranger对各类数据做到精细化管理。例如,首先在Ranger中对各海洋专家分配身份认证密钥,雷达类数据只能由某专家组进行访问,其他人无访问权限,并且专家组中每人可访问的具体雷达数据也不相同。通过身份密钥认证与身份权限机制,对海洋信息做到了精细化管理,明确每个人的访问权限,当发生数据泄密时,可快速追查到责任人,提高了海洋信息的安全性。

6 结论

从大数据生命周期的角度,阐述了不同的大数据技术框架如何组成一套完整的大数据获取、存储、处理、智能分析、安全保护和治理系统。将上述系统应用在海洋信息处理上,以海上观测站为例,可以通过Kafka作为获取框架来获取源源不断产生的观测信息,HBase作为存储框架来存储Kafka获取到的观测信息,同时Storm、Spark Streaming等同步对Kafka中的数据进行处理得出实时结果,TensorFlow等机器学习框架可以作为补充再对数据进行二次分析,挖掘出更深的价值。

大数据技术在当今各领域的应用越来越广泛,也在许多其他领域发挥了重要的作用,一些在海洋信息处理领域的尝试,如浙江省舟山市的海洋大数据中心建立,机器学习对雷达目标的检测等也卓有成效,因此应该越来越多地尝试在海洋信息处理方面使用大数据技术,发现、挖掘出海洋信息中蕴含的巨大潜在价值。

参考文献(References)

- [1] 宋坤. 大数据理念在海洋环境观测数据共享中的应用研究[J]. 海洋开发管理, 2015(6): 43-45.
- [2] Song Kun. Application of the concept of big data in the marine environment observation in data sharing[J]. Marine Development and Management, 2015(6): 43-45.
- [3] 徐文, 鄢社锋, 季飞, 等. 海洋信息获取、传输、处理及融合前沿研究评述[J]. 中国科学(信息科学), 2016, 46(8): 1053.
- [4] Xu Wen, Yan Shefeng, Ji Fei, et al. Review of frontier research on marine information acquisition, transmission, processing and fusion[J]. Scientia Sinica Informations, 2016, 46(8): 1053.
- [5] 解鹏飞, 刘玉安, 赵辉, 等. 基于大数据的海洋环境监测数据集成与应用[J]. 海洋技术学报, 2016, 35(1): 93-101.

- Xie Pengfei, Liu Yu'an, Zhao Hui, et al. Integration and application of marine environmental monitoring data based on large data[J]. Ocean Technology, 2016, 35(1): 93-101.
- [4] 黄冬梅, 邹国良. 海洋大数据[M]. 上海: 上海科学技术出版社, 2016. Huang Dongmei, Zou Guoliang. Ocean big data[M]. Shanghai: Shanghai Scientific & Technical Publishers, 2016.
- [5] 崔松雪, 刘艳艳, 陈戈. 数据仓库技术在海洋大气地理信息系统平台中的应用[J]. 中国海洋大学学报(自然科学版), 2009(增刊1): 455-458. Cui Songxue, Liu Yanyan, Chen Ge. Application of data warehouse technology in marine atmospheric geographic information system platform [J]. Periodical of Ocean University of China, 2009(Suppl 1): 455-458.
- [6] 王立峰. HBase 数据库中大对象存储方案的研究[J]. 电脑知识与技术, 2014(23): 5401-5402. Wang Lifeng. Research on storage scheme of large object in HBase database[J]. Computer Knowledge and Technology, 2014(23): 5401-5402.
- [7] Dimiduk N, Khurana A. HBase in action[M]. Grand Forks, ND: Manning Publications, 2012.
- [8] 王博千, 于齐, 刘辛, 等. 面向 Cassandra 数据库的高效动态数据管理机制[J]. 计算机科学, 2016, 43(7): 197-202. Wang Boqian, Yuqi, Liuxin, et al. Efficient dynamic data management mechanism for Cassandra database[J]. Computer Science, 2016, 43(7): 197-202.
- [9] 马豫星. Redis 数据库特性分析[J]. 物联网技术, 2015(3): 105-106. Ma Yuxing. Analysis of Redis database characteristics[J]. Internet of Things Technologies, 2015(3): 105-106.
- [10] Carlson J. Redis in action[M]. Grand Forks, ND: Manning Publications, 2013.
- [11] 王余蓝. 图形数据库 NEO4J 与关系数据库的比较研究[J]. 现代电子技术, 2012, 35(20): 77-79. Wang Yulan. A comparative study of graphic database NEO4J and relational database[J]. Modern Electronics Technique, 2012, 35(20): 77-79.
- [12] 刘驰, 符积高, 徐闻春. Spark 原理、机制及应用[M]. 北京: 机械工业出版社, 2016. Liu Chi, Fu Jigao, Xu Wenchun. Principle, mechanism and application of Spark[M]. Beijing: China Machine Press, 2016.
- [13] 陈小龙, 关键, 何友. 微多普勒理论在海面目标检测中的应用及展望[J]. 雷达学报, 2013, 2(1): 123-134. Chen Xiaolong, Guan Jian, He You. Application and prospect of micro Doppler theory in sea surface target detection[J]. Journal of Radars, 2013, 2(1): 123-134.
- [14] 黄文坚, 唐源. TensorFlow 实战[M]. 北京: 电子工业出版社, 2017. Huang Wenjian, Tang Yuan. TensorFlow practice[M]. Beijing: Publishing House of Electronics Industry, 2017.
- [15] 王冬海, 卢峰, 方晓蓉, 等. 海洋大数据关键技术及在灾害天气下船舶行为预测上的应用[J]. 大数据, 2017, 3(4): 81-90. Wang Dongmei, Lu Feng, Fang Xiaorong, et al. Key technology of ocean big data and its application in prediction of ship behavior under severe weather[J]. Big Data Research, 2017, 3(4): 81-90.
- [16] Chen X L, Guan J, Bao Z H, et al. Detection and extraction of target with micro-motion in spiky sea clutter via short-time fractional Fourier transform[J]. IEEE Transactions on Geoscience and Remote Sensing, 2014, 52(2): 1002-1018.
- [17] 陈小龙, 关键, 黄勇, 等. 雷达低可观测目标探测技术[J]. 科技导报, 2017, 35(11): 30-38. Chen Xiaolong, Guan Jian, Huang Yong, et al. Radar low observable target detection technique[J]. Science & Technology Review, 2017, 35(11): 30-38.
- [18] 周志华, 王珏. 机器学习及其应用[M]. 北京: 清华大学出版社, 2007. Zhou Zhihua, Wang Yu. Machine learning and its applications[M]. Beijing: Tsinghua University Press, 2007.
- [19] 黄冬梅, 赵丹枫, 魏立斐, 等. 大数据背景下海洋数据管理的挑战与对策[J]. 计算机科学, 2016, 43(6): 17-23. Huang Dongmei, Zhao Danfeng, Wei Lifei, et al. Challenge and countermeasure of ocean data management in big data background[J]. Computer Science, 2016, 43(6): 17-23.
- [20] 刘驰, 胡柏青, 谢一, 等. 大数据治理与安全: 从理论到开源实践[M]. 北京: 机械工业出版社, 2017. Liu Chi, Hu Baiqing, Xie Yi, et al. Big data governance and security: From theory to open source practice[M]. Beijing: China Machine Press, 2017.

Open source big data framework in marine information processing

FAN Luyao¹, ZHANG Jing¹, CHEN Xiaolong², LIU Chi¹

1. School of Software, Beijing Institute of Technology, Beijing 100081, China

2. Department of Electronic and Information Engineering, Naval Aeronautical and Astronautical University, Yantai 264001, China

Abstract With the rapid development of China's sea observation technologies and data collection methods, we have an ever expanding amount of observed data with more and more complexity. Therefore, the traditional data storage and processing techniques cannot effectively manage and analyze the big ocean data. Meanwhile, the regionalization and the distribution of these data become an challenge for a unified data management, during the data mining process. From the perspective of the big data life cycle, this paper reviews the technical principles of the existing open source big data framework in areas of data acquisition, storage, processing, intelligent analysis, security protection and data governance, focusing on their applications for the big ocean data.

Keywords open source; big data framework; marine information processing

(编辑 王志敏)