文章编号:2096-1472(2023)12-0006-06
DOI:10.19644/j.cnki.issn2096-1472.2023.012.002

数据湖研究综述

郭利荣，童坤坤

(中数通信息有限公司大数据工作室，广东 广州 510650)
✉ glr@cndatacom.com; tongkunkun@cndatacom.com

摘 要:数据湖作为一种新兴的数据处理和分析技术,在处理大规模数据集方面表现出了显著的性能优势。国内外相关文献对数据湖的架构、关键技术和应用进行了全面而深入的研究,为相关研究人员提供了有价值的参考。文章首先对数据湖与数据仓库的概念进行了辨析,明确了两者的区别;其次概述了当前流行的数据湖框架和架构,并详细阐述了数据湖的核心功能,包括多源数据的集成、高效的数据存储和计算能力,以及有效的数据治理等;最后探讨了数据湖研究未来的发展方向,如存算分离技术和云原生应用等。

关键词:数据湖;数据存储;数据仓库;数据分析
中图分类号:TP391 **文献标志码**:A

Overview of Data Lake Research

GUO Lirong, TONG Kunkun

(*Big Data Studio, China DataCom Corporation Limited, Guangzhou* 510650, *China*)
✉ glr@cndatacom.com; tongkunkun@cndatacom.com

Abstract: As an emerging data processing and analysis technology, data lakes have shown significant efficiency in processing large-scale datasets. In recent years, relevant literature at home and abroad has conducted comprehensive and in-depth research on the architecture, key technologies, and applications of data lakes, providing valuable references for relevant researchers. Firstly, the concepts of data lake and data warehouse are analyzed and the differences between the two are clarified in this paper. Secondly, framework and architecture of the current popular data lake are summarized, and the core functions of the data lake are elaborated, including the integration of multi-source data, efficient data storage and calculation, and effective data governance. Finally, the future development directions of data lake research are explored, such as storage and computing separation technology, and cloud native applications.

Key words: data lake; data storage; data warehouses; data analysis

0 引言(Introduction)

随着大数据、云计算等技术的不断发展,数据的体量快速增长,数据的内容也越来越复杂,给传统的数据管理和分析带来了巨大挑战。早期,数据管理主要依靠传统关系型数据库,然而这些数据库在面对海量数据时已显得力不从心。数据结构和模式的固定性使得关系型数据库无法适应多变的数据形态和复杂的数据查询需求[1]。同时,由于不同数据库之间缺乏数据共享集成机制,导致数据孤岛问题日益突出。

为了解决上述问题,数据仓库的概念应运而生,它能够集成各种独立数据库中的数据以实现数据共享和分析。然而,传统数据仓库模型已无法满足半结构化和非结构化数据的存储与分析需求。

数据湖作为一种新兴的数据架构和解决方案,能够满足日益增长的多样化数据需求,并且支持结构化、非结构化和半结构化数据的存储和分析等,因此受到广泛关注。常见的结构化数据有数据库表数据,非结构化数据有图像、视频等,半结构化数据有 JSON、XML 等。与数据仓库相比,数据湖更加灵活,能够适应数据的快速变化和多样化的查询需求,能为企业更好地挖掘数据潜在的价值[2]。

本文旨在综述数据湖的相关概念、与数据仓库的关系、流行的实现框架、典型的技术架构、核心功能等,还讨论了数据湖的具体应用场景,并对其未来的发展趋势进行展望。

收稿日期:2023-04-12

1　数据湖概述(Overview of data lake)

1.1　数据湖定义

数据湖的概念于2010年被首次提出,旨在解决传统数据仓库和数据集市面临的问题[3]。首先,数据湖通过统一的元数据存储解决了数据集市之间的数据孤岛问题,实现了数据的集中管理和协作共享。其次,数据湖存储的是原始数据而非经过裁剪后的数据,避免了数据原始信息的丢失,从而为数据分析和挖掘提供了更全面和准确的资源[4]。数据湖是一个集中式存储库,可以以任意规模存储所有结构化和非结构化数据;可以按原样存储数据,并运行不同类型的分析,从控制面板和可视化到大数据处理、实时分析和机器学习,以指导数据使用者做出更好的决策。

众所周知,在数据分析的过程中,数据存储至关重要,而随着数据的增长及其多样性的提升,数据存储模型也在不断地发生改变。在过去的数据存储模型中,数据仓库是一种非常流行的模型。但是,数据仓库在存储数据的时候要求数据必须是预定义的格式和结构,这可能会限制数据的存储和处理[5]。与传统的数据仓库不同,数据湖作为一种新兴的数据存储模型,采用原始格式进行存储。数据湖不需要进行预定义,也没有格式和结构的要求,可以存储各种类型的数据,包括结构化、半结构化和非结构化的数据[6]。与数据仓库相比,数据湖具有以下优势。

(1)灵活性和可扩展性。数据湖可以存储各种类型的数据,并且支持异构数据的存储方式。

(2)不需要ETL(抽取转换加载)过程。数据湖可以直接进行数据分析和挖掘,而不需要进行ETL过程,灵活性更高。

(3)大数据的支持。数据湖能够处理大规模和多样化的数据,如海量的传感器数据、日志数据等。

1.2　数据湖和数据仓库的区别

上文介绍了数据湖与数据仓库之间的联系,而两者之间的详细区别如表1所示。

表1　数据湖与数据仓库的详细区别

Tab.1 Detailed differences between data lake and data warehouse

指标	数据湖	数据仓库
支持的数据类型	支持结构化、非结构化以及半结构化数据类型	主要支持结构化数据
数据存储	数据通常存储在Hadoop分布式文件系统、阿里云对象存储OSS等	数据往往会存储在关系数据库中
数据可靠性	数据质量一般,有可能出现数据沼泽	高质量、高可靠性
灵活性	不用提前建模,灵活度高	需要预先建模,灵活度低
数据访问方式	开放API(应用程序接口),直接读取SQL(数据库语言)	SQL为主,少量支持API
存储成本	相对较低	相对较高
数据处理方式	写入时不校验结构,分析时才定义Schema	写入数据时,需符合提前定义的Schema要求
使用场景	机器学习、数据科学等	BI(商业智能)、数据可视化等

2　常见数据湖框架(Common data lake framework)

随着技术的进步和需求的不断演变,数据湖的概念和实践也在不断发展,陆续出现了许多新的数据湖框架和工具,例如Apache Hudi、Apache Iceberg和Delta Lake等,它们提供了更多高级功能和增强的数据管理能力。这些框架不仅支持原始数据的存储,还提供了ACID(原子性、一致性、隔离性、持久性)的事务特性、元数据管理、数据分区和版本控制等功能,进一步增强了数据湖的一致性、可靠性和可管理性。目前,市面上流行的三大开源数据湖方案分别为Apache Iceberg、Apache Hudi和Delta Lake。

Apache Iceberg是一个由Netflix开发的开源数据湖表格格式,它提供了类似于传统SQL数据库中分区的功能,支持ACID事务和快照等特性。Apache Iceberg支持多种计算引擎(如Hive、Presto、Spark)和存储后端(如HDFS、S3),可以在不同的上层和下层系统中使用,从而实现数据的存储、查询和分析。核心抽象对接新的计算引擎的成本比较低,并且提供了先进的查询优化功能和完全的schema变更。

Apache Hudi是一个开源的数据湖流式处理框架,最初由Uber公司发起并捐献给Apache软件,其设计目标是支持大规模数据的增量计算和管理。Apache Hudi通过支持增量变更、脏数据删除和写入重试等操作,实现了快速、可靠、安全的大规模数据湖管理;它更注重于高效率的Upsert操作和近实时数据更新,提供了Merge On Read文件格式,以及便于搭建增量ETL管道的增量查询功能。Apache Hudi还提供了多种索引适配不同的场景,每种索引都有不同的优点和缺点,因此索引的选择需要根据具体的数据分布进行取舍,从而达到写入和查询的最优解。

Delta Lake是由Databricks推出的开源数据湖格式和处理引擎,它对Parquet数据格式进行了扩展,支持ACID事务、版本控制和流表查询等特性。Delta Lake可以与Spark、Presto等计算框架集成,适用于大规模数据处理和分析场景。此外,Delta Lake还能保证数据安全性和可靠性,能够满足企业级应用的需求。

3　数据湖架构(Data lake architecture)

数据湖可以认为是新一代的大数据基础设施,数据湖技术发展至今,其架构也经历了一些演进,正在逐步完善。早期数据湖架构采用的两层架构(图1)为临时数据区和原始数据区[7]。临时数据区可以作为临时存储区域,能够快速接收和存储各种类型的数据,而不用对其进行特殊的转换或规范化处理;原始数据区作为存储各类原始数据的持久化区域,数据在此处保留其原始状态。临时数据区注重灵活性和快速处理,适合实验和即时分析;而原始数据区注重数据保留和管理,适合数据溯源和长期分析需求。这种两层架构简单、直接,但在大规模数据处理和复杂分析场景下存在一些问题,比如处理速度慢、数据质量控制困难和出现数据不一致性。

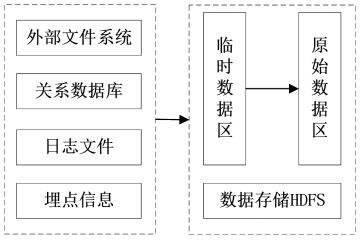

图1　两层数据湖架构

Fig. 1 Two-layered data lake architecture

为了克服两层架构的局限性，引入 Lambda 架构(图 2)[8]。Lambda 架构为三层结构，即批处理层、速度层和查询层。批处理层负责对数据进行批处理和离线处理。数据从各个来源发送到批处理层进行数据清洗、转换和存储。批处理层使用分布式存储系统(如 HDFS)存储原始数据和批处理结果，并结合大数据处理技术(如 MapReduce)进行数据分析和计算。速度层负责对数据进行实时处理和流式处理。数据从源头发送到实时处理层，经过即时处理和转换，产生实时结果和聚合。实时处理层使用流式处理引擎(如 Spark Streaming、Flink)处理连续流数据，并将结果存储在速度层数据库中。查询层会使用批处理层和速度层的结果提供实时查询和分析，以满足不同类型的查询需求。

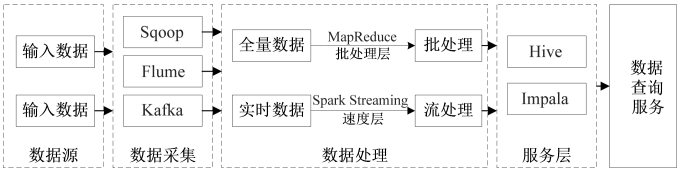

图 2 Lambda 架构

Fig. 2 Lambda architecture

总体来说，数据源被分为两个流，一个进入批处理层进行离线处理，另一个进入速度层进行实时处理。这样可以在保证实时性的同时，进行复杂分析和查询。然而，Lambda 架构需要维护两套数据处理流程和代码，并且存在数据一致性问题。

为了降低 Lambda 架构的复杂性，Kappa 架构被提出，如图 3 所示。Kappa 架构取消了批处理层，只使用速度层进行数据处理和存储。数据通过流处理方式进行实时处理，并将结果直接存储在数据湖中。Kappa 架构相较于 Lambda 架构，其简化了架构和技术栈，但无法应对需要大规模离线处理和计算的场景，并且可能难以实现和保证数据的一致性。

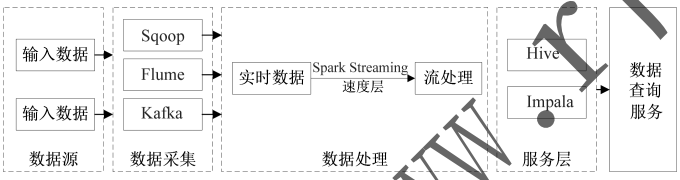

图 3 Kappa 架构

Fig. 3 Kappa architecture

上文介绍的架构都存在一个共同的特点，即它们都比较关注数据的存储和计算而忽略了对数据本身的管理。数据湖作为新型大数据基础设施，在继承大数据平台的存储计算能力的基础上，通过统一的数据接入、全面的元数据管理、精细化的数据治理等功能，实现对海量异构数据的深度管理与资产化利用，以便各类计算引擎能够深度融合，覆盖多种应用场景。基于上述思想，典型的数据湖架构如图 4 所示。

4 数据湖核心功能(Core functions of data lake)

数据湖强调对业务数据的保真存储，允许存储任意格式的数据，提供完善的数据管理能力，实现数据全生命周期管理。具体来看，数据湖至少包含原始数据和处理后的数据两类，通过统一的数据接入接口，进行数据源、连接、格式、模式等元数据管理，支持细粒度权限控制，追踪数据从接入、存储、处理到消费的全流程，重构数据血缘和流动过程，实现对海量异构数据的集中式、原始的存储与可控可治理的资产化利用。

接下来，本文从数据获取、数据存储、数据计算、数据治理

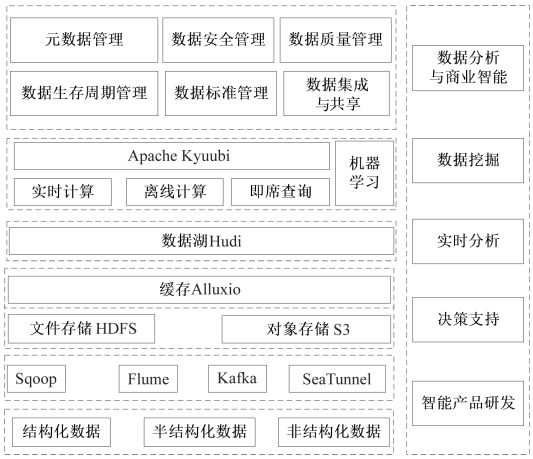

图 4 典型的数据湖架构

Fig. 4 Typical data lake architecture

等方面详细介绍数据湖技术。

4.1 数据获取

数据湖作为一种集中存储和管理企业各种类型和格式的原始数据的架构，其数据输入与获取技术对于数据湖的建设和应用至关重要。Sqoop 用于将关系型数据库中的数据导入数据湖中，支持各种常见的关系型数据库，如 MySQL、Oracle 等；Flume 是一个广泛使用的分布式数据采集工具，适用于从多个数据源(如日志、消息队列)采集、聚合和移动数据到数据湖中；相比 Sqoop 的批量传输，Kafka 实现了从源端不间断地获取数据，使数据湖可以直接对接实时数据，支持实时分析应用[9-10]。

SeaTunnel 是一种用于数据集成和数据同步的解决方案，它提供了多种功能，使用户能够轻松地在不同的数据源之间进行数据传输和同步。SeaTunnel 支持各种类型的数据源，包括关系型数据库、大数据存储、文件系统等。无论数据源是什么类型，SeaTunnel 都能够直接连接并获取数据。这使得用户可以将数据源中的数据集成到一个统一的平台上，方便进行数据分析和处理。

SeaTunnel 还提供了强大的数据转换功能。用户可以通过使用内置的转换规则或自定义的转换脚本转换数据格式和结构。例如，用户可以将一个数据集中的列重新排列、过滤掉特定的行或进行数据分组和聚合等操作。这些功能使得用户能够根据自己的需求对数据进行灵活的处理，从而更好地满足分析和业务的需求。

SeaTunnel 具有高效的数据传输和同步功能，它使用了高性能的数据传输协议和压缩算法，确保数据在传输过程中的安全性和高效率。同时，SeaTunnel 支持增量同步，即只传输发生变化的数据，大大减少了数据传输的时间和带宽的消耗。这使得用户可以实时地将数据从一个数据源同步到另一个数据源，保持数据的一致性和及时性。

SeaTunnel 提供了丰富的监控和管理功能。用户可以监控数据传输和同步的进程和状态，及时发现和解决问题。SeaTunnel 还支持任务调度和自动化，用户可以预先设置数据传输和同步的时间和频率，减少手动操作的工作量。此外，SeaTunnel 提供了数据一致性校验和错误处理等功能，确保数据传输和同步的可靠性和准确性。

4.2 数据存储

数据湖作为企业中全量数据的单一存储，可以集成和存储来自不同数据源的数据，包括关系型数据库、日志文件、传感器

数据等。这种存储方式可以使不同部门和用户在同一个存储中查找和访问数据,促进数据共享和协作。数据湖的数据存储技术作为一种集成多种存储方式和支持多种数据格式的解决方案,旨在满足企业对于海量数据的集中存储和管理需求。为了具备性价比,数据湖常选用相对便宜的存储引擎,对应的存储技术主要包括关系数据库存储、HDFS 存储和对象存储等方式[11]。关系数据库存储主要适用于结构化数据的存储,可以提供高效的数据查询和处理功能。HDFS 存储是分布式文件系统的一种,可以支持大规模数据的存储和处理,适用于结构化和非结构化数据的存储。对象存储可选择云存储,如 S3、OSS 和 OBS,具备弹性和按需扩容的特性[12]。对象存储非常适用于大量非结构化数据的存储,例如图片、视频、日志等。

4.3 数据计算

数据湖作为一个综合性的数据管理平台,其中一项关键功能就是数据计算。数据计算在数据湖中起着至关重要的作用,它能够处理和分析各种类型的数据,以支持各种业务需求。为了让数据湖支持多源异构数据的联合分析,计算框架的选择尤为重要。现有的 Spark、Flink 等计算框架可以用于流批一体的数据处理,但在支持复杂 SQL 解析和优化方面还不够完善。此外,不同的计算框架缺乏统一的接口标准,给多引擎集成带来困难。Apache Kyuubi 是一个开源的分布式 SQL 引擎,它可以优雅地解决数据湖计算的难题,提供了基于 Thrift 的 JDBC/ODBC 和 REST 两种标准服务接口。Apache Kyuubi 可以对接 Spark、Flink、Hive 等主流的分布式计算框架,以及 Doris、Trino 等新型的分析查询引擎,还支持任何遵循 JDBC 标准的数据库。

Apache Kyuubi 具有多租户隔离、查询负载均衡等分布式数据库的特性,使其能够应对企业中的多种大数据处理场景,如数据提取转换加载、业务智能报表等需求。Apache Kyuubi 的目标是利用其框架优势,为构建企业数据湖提供标准化和统一的 SQL 访问接口;它允许用户用常规的 SQL 查询方式处理存储在数据湖中的结构化、半结构化及非结构化数据。同时,它正在朝着一个面向无服务器化 SQL 分析的 Lakehouse 服务方向演进,可以通过对各种计算框架的标准化对接,构建一个池化、弹性的分布式 SQL 计算平台,为企业级数据湖的 Serverless 化提供支撑。

在数据湖中,数据计算的功能可以支持离线计算、实时计算、即席查询和机器学习等多种计算模式。离线计算是基本的计算模式,它主要是基于批量数据处理的思想对大量数据进行处理和分析。离线计算通常以天或者周为单位进行,处理的数据量比较大且计算过程可能需要耗费较长时间。在数据湖中,离线计算通常使用分布式计算框架如 Hadoop 和 Spark 等实现。

此外,还有一种重要的计算模式是实时计算,它主要是对实时流入的数据进行实时处理和分析。实时计算对于要求数据处理速度和实时性的场景非常有用,例如在线游戏、实时监控等。在数据湖中,实时计算通常使用流处理框架如 Apache Kafka 和 Apache Flink 等实现。

即席查询是一种灵活的数据查询方式,它可以根据不同的查询需求进行即时的数据处理和分析。即席查询通常用于探索性分析和业务人员的自助分析场景中,可以根据分析人员的需要灵活地选择不同的数据集、指标和可视化方式。在数据湖中,交互式即席查询可以通过可视化工具或 Trino 途径实现。

对一个成熟的数据湖平台来说,其计算引擎模块应具备可扩展性与可插拔性,能够平滑地兼容不同的机器学习框架与算法。目前,主流的 TensorFlow 和 PyTorch 深度学习框架已经原生支持直接从分布式文件系统和对象存储中读取数据进行模型训练。这种灵活性使得数据湖能够更好地支持各种机器学习任务,为数据驱动的决策提供更强大的能力。

总的来说,数据湖的数据计算功能能够高效、灵活和可扩展地处理和分析各种类型的数据,支持各种业务需求。无论是离线计算、实时计算、即席查询还是机器学习,数据湖都可以提供强大的数据处理和分析能力,帮助企业更好地发掘数据的价值,推动业务的创新和发展。

4.4 数据治理

数据治理是数据湖的重要功能,目的是保证数据湖中的数据质量,让数据为企业创造更高价值[13]。数据治理涵盖了数据湖的整个生命周期,包括制定数据采集策略,从各种源系统中抽取数据,对数据进行转换整理,将不同格式的数据集成到数据湖中。此外,数据治理会建立完备的数据目录,记录每一个数据集的关键元数据信息,方便数据的发现和利用。在数据湖运行过程中,需要持续监控数据流和数据变更,及时发现问题并做出优化。同时,通过访问控制、加密等手段保证数据安全,避免非授权访问以及数据泄露。数据治理还会对数据湖中的数据集进行质量检查,识别重复、错误和无效数据,并进行修正和过滤。所有这些治理措施的目的都是让数据湖成为高质量、高可靠性的基础数据平台,为企业的决策分析和业务创新提供可信可用的数据支持[14-16]。

4.4.1 元数据管理

元数据是描述数据的数据,主要是描述数据属性的信息。在数据湖中,元数据管理是数据治理的重要组成部分[17-18]。通过良好的元数据管理,可以对数据湖中的数据进行有效的分类、标记和描述,使用户能够更快速地找到所需的数据集,并理解数据的结构、含义和质量标准[19]。元数据管理能够提供数据湖中数据的整体视图,帮助用户更好地理解和利用数据。元数据管理模块还会持续跟踪元数据的变更,提供版本控制、血缘追踪等功能,为企业构建知识图谱奠定基础,让数据资产可追溯、可解释。目前,实现智能化和自动化的元数据管理是数据湖建设的重要方向,可以运用机器学习、自然语言处理等技术提取数据特征和数据之间的关系,减轻手工录入工作量,使元数据更完整、可靠。

4.4.2 数据安全管理

数据湖中的数据通常包含机密、敏感或受限制的信息。数据湖必须具备强大的数据安全管理功能,以保护数据的机密性、完整性和可用性[11,20-22]。数据治理能够为数据湖建立访问权限控制、数据加密和身份验证等安全策略,确保数据在存储、传输和使用过程中得到有效的保护,并遵守相关的数据保护法规和合规要求。

4.4.3 数据质量管理

数据质量是数据湖中的重要考量因素。数据湖中的数据来自不同的源头,可能存在重复、冗余、不一致等问题。数据治理通过建立数据质量管理框架和规范,使数据湖中的数据经过验证、清洗和标准化,确保数据的准确性、一致性和完整性。数据质量管理还可以通过监控和度量数据质量指标实时监控数据湖中数据的质量,并快速响应和修复潜在的数据质量问题。

4.4.4 数据生存周期管理

数据湖中的数据具有不同的生存周期,包括数据的创建、

更新、使用、存储和删除等阶段。数据治理可以提供数据生存周期管理策略和流程，确保在数据湖中的数据按照规定的生存周期管理方法进行管理和操作。数据生存周期管理可以帮助审计数据使用情况、规划数据存储需求、控制数据增长和存储成本，并且保证数据的合规性。

4.4.5 数据标准管理

数据治理通过建立数据标准化的方法和过程，确保数据湖中的数据按照一致的标准进行管理和使用。数据湖中的数据来源广泛且多样，可能包含不同格式、结构和质量的数据。通过数据治理，可以建立数据规范和数据词典，定义和标准化数据的命名约定、数据结构和数据元素等。数据标准提供了一致的数据语义和结构，使不同用户在数据湖中能够理解和使用数据，从而提高数据集成和数据共享的效率。

4.4.6 数据集成与共享

数据湖作为一个集成多源数据的架构，促进了数据的集成和共享。数据治理在数据湖中的数据集成和共享方面起到重要作用。数据湖中的数据来源可能包括内部和外部的多个数据源，并且以不同的格式和结构存在。通过数据治理，可以建立数据集成策略和流程，将不同来源的数据集成到数据湖中，并确保数据的一致性和可靠性。数据湖作为一个统一访问和查询的数据存储，使得用户可以共享数据，进行跨部门和跨应用的数据分析和应用开发。

5 数据湖应用场景(Data lake application scenarios)

在当今企业信息化建设中，高效管理应用海量、复杂数据是一项关键任务。只有充分利用数据资产，企业才能更好地挖掘数据的价值，提高业务运营效率，优化决策过程，从而在激烈的市场竞争中获得优势。数据湖的出现为企业提供了一种更好的数据管理和分析工具，使企业能够快速、高效地管理、使用和分析数据，可以在多个领域帮助企业解决实际问题。

5.1 金融领域

交易分析：金融机构可以将所有交易数据集中存储在数据湖中，利用数据湖分析市场趋势、分析客户的行为模式以及帮助金融机构进行风险和欺诈检测[23]。

客户行为分析：通过整合不同的数据源，如交易历史记录、客户反馈、社交媒体数据等，数据湖可以帮助金融机构理解客户的行为模式，并提供个性化的产品和服务。

5.2 医疗领域

疾病诊断与预测：数据湖可以集中存储患者的临床数据、基因组数据、医疗图像和传感器数据等信息，通过分析这些数据，医疗机构可以提供更精确的疾病诊断、预测和保健建议[24]。

医疗研究：数据湖可以帮助医疗研究人员整合和分析大量的医疗数据，加速新药研发和更好地开展医学研究和临床试验。

5.3 零售领域

消费者行为分析：通过整合顾客的交易记录、网站浏览数据、社交媒体数据等，数据湖可以帮助零售商了解消费者的购买行为和偏好，进而提供个性化的产品推荐和营销策略。

库存管理：通过与供应链数据和销售数据的整合，数据湖可以帮助零售商准确预测需求、优化库存管理，并提高供应链的效率。

5.4 能源领域

智能电网管理：数据湖可以集中存储来自智能电表、传感器和设备的大量数据，通过对数据的分析，能够实现对能源消耗的监测、实时故障检测和优化能源分配[25]。

风能和太阳能预测：数据湖可以整合气象数据、能源生产数据和能源消耗数据等，通过分析这些数据，预测风能和太阳能的产生情况，帮助能源公司做出更准确的能源规划和决策。

5.5 烟草领域

经营分析：整合现有数据源情况，完成营销、物流、专卖、财务四大数据源的数据入湖，实现数据集成、数据处理、数据服务，可视化呈现 BI、报表、经营分析等，帮助烟草公司全面开展数据资产化运营工作。

6 进一步研究方向(Further research directions)

针对当前数据湖技术的研究进展，本文给出未来数据湖技术比较有价值的研究方向。

6.1 存算分离

存算一体的数据湖架构，在资源扩展时，需要同时升级存储和计算节点，无法对指定资源进行独立扩展，而存算分离作为一种新兴的数据处理模式，将数据的存储和计算分离开来，使得计算能力可以弹性扩展，并可以与多个计算引擎集成。数据湖技术可以借鉴存算分离的思想，进一步发展多计算引擎集成的能力，实现高效的弹性伸缩和资源利用，降低运维成本，优化存储和计算的协同工作，提高数据湖的处理效率和性能。

6.2 云原生技术

传统数据湖通常需要大量的硬件资源，包括服务器、存储设备、网络设备等，成本较高，并且需要投入更多的人力和资源进行系统的配置、监控、维护和升级，增加了管理的复杂度和成本；而云原生技术是构建和部署在云平台的应用程序的一种方法，它强调容器化、自动化。数据湖技术可以应用云原生技术，如容器化和微服务架构，实现更灵活、可扩展和高可用的数据湖部署和管理。云原生技术的发展可以使数据湖更好地适应云平台的特性和要求。

6.3 湖仓一体化

数据湖技术具有开放性和灵活性的特点，容易将原始的、未经过验证和清洗的数据直接存储到数据湖中，这可能导致数据质量问题，如数据重复、缺失、不一致等。相比之下，数据仓库更注重数据质量管理，通过清洗、转换和整合等步骤确保数据的准确性和一致性[26]。湖仓一体化将数据湖和数据仓库进行整合，构建统一的数据管理平台，能够提供统一的数据视图。下一步的发展方向包括构建数据湖和数据仓库的联邦查询和数据融合机制，实现跨数据湖和数据仓库的数据查询和分析。湖仓一体化还可以进一步优化数据存储和数据处理的接口，提高数据集成和传输效率。

6.4 数据安全和隐私保护

随着数据价值的不断提升，数据安全和隐私保护成为研究热点，因此需要进一步研究和开发数据加密、访问控制、数据脱敏等安全技术和措施，不断提高保护数据安全和隐私的能力。

6.5 智能元数据管理

随着数据湖中存储的数据量不断增长，元数据管理变得至关重要，元数据管理可以帮助组织对数据进行治理，包括数据的标准化、规范化、权限管理等，同时可以帮助用户了解数据的使用情况，并实现数据的共享和重用。下一步的发展方向包括利用存储层的元数据和计算引擎的元数据进行数据质量评估和监控，实现自动化的数据清洗和验证，并借助相关技术实现元数据的快速采集、维护和查询；使用机器学习、知识图谱等技术让元

数据管理更自动化和智能化。可以根据数据内容和关系自动生成元数据,并进行持续的学习优化。

7 结论(Conclusion)

本文从多个方面介绍了数据湖技术,包括技术诞生背景、与传统数据仓库的区别及其核心功能未来的发展方向等。数据湖技术具有强大的数据处理和分析能力,它已经成为数字化时代企业和组织的重要工具。随着数据湖技术的进一步研究和应用,可以期待它发挥更加重要的作用,为各个领域的发展和决策提供更加准确和实时的数据支持。

参考文献(References)

[1] MILOSLAVSKAYA N,TOLSTOY A. Application of big data,fast data,and data lake concepts to information security issues[C]//IEEE. Proceedings of the 2016 IEEE 4th International Conference on Future Internet of Things and Cloud Workshops. Piscataway, Piscataway: IEEE, 2016: 148-153.

[2] HLUPIĆ T,OREŠČANIN D,RUŽAK D,et al. An overview of current data lake architecture models[C]//IEEE. Proceedings of the 2022 45th Jubilee International Convention on Information,Communication and Electronic Technology. Piscataway:IEEE,2022:1082-1087.

[3] DIXON J. Pentaho, hadoop, and data lakes [EB/OL]. (2010-10-14) [2020-12-10]. https://jamesdixon.wordpress.com/2010/10/14/pentaho-hadoop-and-data-lakes.

[4] FANG H. Managing data lakes in big data era:what's a data lake and why has it became popular in data management ecosystem[C]//IEEE. Proceedings of the 2015 IEEE International Conference on Cyber Technology in Automation,Control,and Intelligent Systems. Piscataway: IEEE, 2015:820-824.

[5] 陈氢,张治. 融合多源异构数据治理的数据湖架构研究[J]. 情报杂志,2022,41(5):139-145.

[6] KHINE P P,WANG Z S. Data lake:a new ideology in big data era[C]//Edition Diffusion Presse Sciences. Proceedings of the 4th Annual International Conference on Wireless Communication and Sensor Network. Paris:EDP Sciences,2018:1-11.

[7] GIEBLER C,GRÖGER C,HOOS E,et al. Leveraging the data lake:current state and challenges[C]//Ordonez C, Song I-Y,Anderst-Kotsis G,et al. Big Data Analytics and Knowledge Discovery. Cham:Springer,2019:179-188.

[8] MUNSHI A A,MOHAMED Y A-R I. Data lake lambda architecture for smart grids big data analytics[J]. IEEE Access,2018,6:40463-40471.

[9] SURIARACHCHI I,PLALE B. Crossing analytics systems: a case for integrated provenance in data lakes[C]//IEEE. Proceedings of the 2016 IEEE 12th International Conference on e-Science. Piscataway:IEEE,2016:349-354.

[10] JOHN T,MISRA P. Data Lake for Enterprises[M]. Birmingham:Packt Publishing,2017:596-596.

[11] BEHESHTI A,BENATALLAH B,NOURI R,et al. CoreDB: a data lake service[C]//ACM. Proceedings of the 2017 ACM on Conference on Information and Knowledge Management. New York:ACM,2017:2451-2454.

[12] ZAGAN E,DANUBIANU M. Cloud DATA LAKE:the new trend of data storage[C]//IEEE. Proceedings of the 2021 3rd International Congress on Human-Computer Interaction,Optimization and Robotic Applications. Piscataway:IEEE,2021:1-4.

[13] 张宁,袁勤俭. 数据治理研究述评[J]. 情报杂志,2017,36(5):129-134,163.

[14] DERAKHSHANNIA M,GERVET C,HAJJ-HASSAN H,et al. Data lake governance:towards a systemic and natural ecosystem analogy[J]. Future Internet,2020,12(8):126.

[15] BROUS P,JANSSEN M,KRANS R. Data Governance as Success Factor for Data Science[M]. Switzerland:Springer,2020:431-442.

[16] RIFAIE M,ALHAJJ R,RIDLEY M. Data governance strategy:a key issue in building Enterprise Data Warehouse [C]//IIWAS. Proceedings of the 11th International Conference on Information Integration and Web-based Applications & Services. New York: ACM,2009:587-591.

[17] SAWADOGO P N,SCHOELLY É,FAVRE C,et al. Metadata systems for data lakes: models and features [C]// Springer Science. New Trends in Databases and Information Systems. Berlin:Springer,2019:440-451.

[18] MACCIONI A,TORLONE R. KAYAK:A framework for just-in-time data preparation in a data lake[M]// KROGSTIE J, REIJERS H A. Advanced Information Systems Engineering. Cham:Springer,2018:474-489.

[19] SAWADOGO P,DARMONT J. On data lake architectures and metadata management[J]. Journal of Intelligent Information Systems,2021,56(1):97-120.

[20] MARTY R. The security data lake[M]. Sebastopol:O'Reilly Media,2015:1-2.

[21] RAVAT F,ZHAO Y. Data lakes:trends and perspectives [C]//Springer Science. Database and Expert Systems Applications. Cham:Springer,2019:304-313.

[22] BERTINO E,FERRARI E. Big data security and privacy [M]//Springer. Studies in Big Data. Cham: Springer, 2017:425-439.

[23] 王富彬. 基于数据湖的银行 OLAP 系统研究与实现[D]. 上海:华东师范大学,2022.

[24] RANGARAJAN S,LIU H,WANG H,et al. Scalable architecture for personalized healthcare service recommendation using big data lake[C]//BEHESHTI A,HASHMI M,DONG H, et al. Service Research and Innovation. Cham:Springer,2018:65-79.

[25] 曾飞,杨雄,苏伟,等. 基于区块链与数据湖的电力数据存储与共享方法[J]. 电力工程技术,2022,41(3):48-54.

[26] 陈氢,宋仕伟. 数据治理视角下的湖仓一体架构研究[J]. 数字图书馆论坛,2023,19(4):19-28.

作者简介:

郭利荣(1977-),男,硕士,工程师。研究领域:数据治理,人工智能。

童坤坤(1992-),男,硕士,工程师。研究领域:数据仓库,数据湖。