МИНИСТЕРСТВО ОБРАЗОВАНИЯ И НАУКИ РОССИЙСКОЙ ФЕДЕРАЦИИ

ФЕДЕРАЛЬНОЕ ГОСУДАРСТВЕННОЕ БЮДЖЕТНОЕ ОБРАЗОВАТЕЛЬНОЕ УЧРЕЖДЕНИЕ ВЫСШЕГО ПРОФЕССИОНАЛЬНОГО ОБРАЗОВАНИ «РОССИЙСКИЙ ГОСУДАРСТВЕННЫЙ ПЕДАГОГИЧЕСКИЙ УНИВЕРСИТЕТ им. А. И. ГЕРЦЕНА»

Институт компьютерных наук и технологического образования Кафедра компьютерных технологий и электронного обучения

КУРСОВАЯ РАБОТА

ТЕМА КУРСОВОЙ РАБОТЫ

Большие данные и инструменты для их обработки

Направление подготовки: «Технологии компьютерного моделирования»

Руководитель:

Доктор педагогических наук, профессор,

_____ Е.З. Власова

« ___» _____ 2025 г.

Автор работы студент

группы 1об_ИВТ-1

_____ А.С. Тихонов

« ___» _____ 2025 г.

Санкт-Петербург

2025

ВВЕДЕНИЕ

Большие данные - это термин, описывающий огромные объемы структурированных, полуструктурированных и неструктурированных данных, которые слишком сложны для обработки с использованием традиционных методов и инструментов анализа. Основные характеристики больших данных определяются с помощью модели **3V** (позднее расширенной до **5V**):

Volume (Объем) - огромное количество данных, измеряемое в терабайтах, петабайтах и более.

Velocity (Скорость) - высокая скорость генерации и обработки данных, часто в режиме реального времени.

Variety (Разнообразие) - разнородность данных, включающая тексты, изображения, видео, аудио, сенсорные данные и т.д.

Дополнительные характеристики:

Veracity (Достоверность) - качество и надежность данных.

Value (Ценность) - полезность данных для извлечения значимой информации.

Большие данные находят применение в различных сферах, таких как бизнес, наука, медицина, финансы и технологии, позволяя выявлять закономерности, прогнозировать тенденции и принимать обоснованные решения на основе анализа. Для работы с большими данными используются специализированные технологии.

Работа с большими данными связана с рядом сложностей и нюансов, которые обусловлены их объемом, разнообразием, скоростью генерации и другими характеристиками. Анализ больших данных требует использования сложных алгоритмов машинного обучения и статистических методов.

Методы обработки больших данных включают в себя широкий спектр технологий и подходов, начиная от распределенной обработки и хранения данных и заканчивая машинным обучением и визуализацией. Выбор конкретного метода зависит от задач, характеристик данных и доступных ресурсов. Современные инструменты и платформы, такие как Hadoop, Spark, Kafka и TensorFlow, делают обработку больших данных более доступной и эффективной, открывая новые возможности для бизнеса, науки и технологий.

Перечень технологий для работы с большими данными обширен: на рынке представлено множество коммерческих решений, которые позволяют организациям реализовывать широкий спектр аналитических задач - от генерации отчетов в реальном времени до внедрения приложений машинного обучения.

Помимо этого, доступно большое количество инструментов с открытым исходным кодом. Некоторые из них также предлагаются в виде коммерческих версий или входят в состав

платформ для работы с большими данными, а также предоставляются как часть управляемых сервисов. Далее я опишу все преимущества некоторых программ для обработки больших данных.

Программы для обработки больших данных.

KNIME

KNIME - это мощная платформа с открытым исходным кодом для анализа данных, интеграции и визуализации. Она широко используется в области Data Science, бизнес-аналитики и машинного обучения. KNIME предоставляет интуитивно понятный интерфейс, основанный на визуальном программировании, что делает его доступным как для технических специалистов, так и для пользователей без глубоких знаний в программировании.

Основные возможности KNIME:

- **Визуальное программирование:**

KNIME использует подход, основанный на создании рабочих процессов (workflows), где каждый шаг представлен в виде узла (node). Пользователи могут перетаскивать узлы и соединять их для создания сложных аналитических процессов.

Это позволяет визуализировать весь процесс анализа данных, от загрузки и предобработки до моделирования и визуализации.

- **Интеграция данных:**

KNIME поддерживает работу с различными источниками данных, включая базы данных (SQL, NoSQL), файлы (Excel, CSV, JSON), облачные хранилища и API.

Платформа позволяет объединять данные из разных источников, что делает её удобной для интеграции разнородной информации.

- **Анализ и машинное обучение:**

KNIME включает в себя широкий набор инструментов для статистического анализа, машинного обучения и искусственного интеллекта.

Поддерживаются популярные алгоритмы, такие как регрессия, классификация, кластеризация, а также интеграция с библиотеками машинного обучения, такими как TensorFlow, PyTorch и Scikit-learn.

- **Расширяемость:**

KNIME имеет модульную архитектуру, что позволяет расширять её функциональность с помощью дополнительных плагинов.

Пользователи могут писать собственные узлы на Java, Python, R или других языках

программирования.

- **Визуализация данных:**

Платформа предоставляет инструменты для создания интерактивных графиков, диаграмм и отчетов.

Результаты анализа можно экспортировать в различные форматы, такие как PDF, Excel или HTML.

- **Поддержка больших данных:**

KNIME интегрируется с технологиями для работы с большими данными, такими как Apache Spark и Hadoop, что позволяет обрабатывать крупные объемы информации.

- **Кроссплатформенность:**

KNIME доступна для Windows, macOS и Linux, что делает её универсальным инструментом для различных операционных систем.

Преимущества KNIME:

- **Простота использования:** Визуальный интерфейс делает KNIME доступной для пользователей с разным уровнем подготовки.
- **Гибкость:** Поддержка множества языков программирования и интеграция с различными технологиями.
- **Сообщество и поддержка:** Активное сообщество пользователей и разработчиков, а также наличие коммерческой версии с дополнительной поддержкой.
- **Бесплатная версия:** KNIME Analytics Platform доступна бесплатно.
-

Hadoop

Hadoop - распределенная среда для хранения данных и запуска приложений на кластерах стандартного аппаратного обеспечения. Hadoop была разработана как передовая технология работы с большими данными для обработки растущих объемов структурированной, неструктурированной и полу структурированной информации. Впервые выпущенная в 2006 году, она стала практически синонимом больших данных; с тех пор ее частично затмили другие технологии, но она по-прежнему широко используется.

Основные компоненты Hadoop:

HDFS (Hadoop Distributed File System):

Распределенная файловая система, предназначенная для хранения больших объемов

данных на множестве серверов.

Данные разбиваются на блоки (обычно 128 МБ или 256 МБ) и реплицируются на несколько узлов для обеспечения отказоустойчивости.

HDFS оптимизирована для работы с большими файлами и обеспечивает высокую пропускную способность.

MapReduce:

Программная модель для параллельной обработки данных на кластере.

Состоит из двух этапов:

Map: Разделение данных на части и их предварительная обработка.

Reduce: Агрегация результатов, полученных на этапе Map.

MapReduce позволяет эффективно обрабатывать большие объемы данных, распределяя задачи между узлами кластера.

YARN (Yet Another Resource Negotiator):

Система управления ресурсами и планирования задач в Hadoop.

Позволяет запускать различные приложения на одном кластере, включая MapReduce, Apache Spark и другие.

Hadoop Common:

Набор библиотек и утилит, которые поддерживают работу других компонентов Hadoop.

Преимущества Hadoop для обработки больших данных:

Масштабируемость:

Hadoop может масштабироваться от нескольких серверов до тысяч узлов, что позволяет обрабатывать эксабайты данных.

Отказоустойчивость:

Данные автоматически реплицируются на несколько узлов, что обеспечивает высокую надежность и устойчивость к сбоям.

Экономическая эффективность:

Hadoop работает на стандартном оборудовании, что снижает затраты на инфраструктуру по сравнению с традиционными системами хранения и обработки данных.

Гибкость:

Hadoop может работать с различными типами данных: структурированными, полуструктурированными и неструктурированными (тексты, изображения, логи и т.д.).

Высокая производительность:

Благодаря распределенной обработке данных, Hadoop обеспечивает высокую скорость

выполнения задач.

Apache Kylin

Apache Kylin - это распределенная аналитическая платформа с открытым исходным кодом, разработанная для ускорения обработки запросов к большим данным. Kylin специализируется на выполнении OLAP-запросов (Online Analytical Processing) в системах, где данные хранятся в Hadoop или других распределенных хранилищах. Он использует технологию **предварительного вычисления (pre-aggregation)**, чтобы значительно ускорить выполнение сложных аналитических запросов.

Основные особенности Apache Kylin:

Высокая производительность:

Kylin предварительно вычисляет агрегаты и сохраняет их в оптимизированном формате, что позволяет выполнять запросы за секунды, даже на огромных объемах данных.

Интеграция с Hadoop:

Kylin тесно интегрирован с экосистемой Hadoop, используя HDFS для хранения данных и Apache Hive для их подготовки.

Поддержка SQL:

Kylin поддерживает стандартный SQL-интерфейс, что делает его удобным для аналитиков и разработчиков, уже знакомых с SQL.

Масштабируемость:

Kylin может обрабатывать петабайты данных, распределяя вычисления на кластере Hadoop.

Поддержка многомерного анализа:

Kylin позволяет создавать OLAP-кубы (кубы данных), которые обеспечивают многомерный анализ данных.

Интеграция с BI-инструментами:

Kylin поддерживает интеграцию с популярными BI-инструментами, такими как Tableau, Power BI и Apache Superset.

Преимущества Apache Kylin:

Скорость выполнения запросов:

Благодаря предварительному вычислению агрегатов, Kylin обеспечивает высокую скорость выполнения аналитических запросов.

Экономия ресурсов:

Kylin уменьшает нагрузку на кластер Hadoop, так как большая часть вычислений выполняется заранее.

Простота использования:

Поддержка SQL и интеграция с BI-инструментами делают Kylin доступным для широкого круга пользователей.

Масштабируемость:

Kylin может обрабатывать огромные объемы данных, что делает его подходящим для крупных предприятий.

Spark

Spark - это механизм обработки и анализа данных в памяти, который может работать на кластерах, управляемых Hadoop YARN, Mesos и Kubernetes, или в автономном режиме. Он позволяет выполнять масштабные преобразования и анализ данных; может использоваться как для пакетных, так и потоковых приложений

Основные особенности Apache Spark:

Высокая производительность:

Spark использует оперативную память для хранения промежуточных данных, что делает его значительно быстрее, чем Hadoop MapReduce, который работает с диском.

Поддержка различных типов обработки:

Пакетная обработка: Анализ больших объемов данных, хранящихся в распределенных системах.

Потоковая обработка: Обработка данных в реальном времени (Spark Streaming).

Машинное обучение: Встроенная библиотека MLlib для построения моделей машинного обучения.

Графовая аналитика: Библиотека GraphX для работы с графами.

Простота использования:

Spark предоставляет API на Java, Scala, Python и R, что делает его доступным для широкого круга разработчиков.

Поддержка SQL-запросов через Spark SQL.

Интеграция с экосистемой больших данных:

Spark может работать с данными, хранящимися в HDFS, Apache Cassandra, Amazon S3 и других системах.

Поддерживает интеграцию с Hadoop YARN, Apache Mesos и Kubernetes.

Масштабируемость:

Spark может работать на кластерах из тысяч узлов, что позволяет обрабатывать эксабайты данных.

Архитектура Apache Spark:

Driver Program:

Главная программа, которая координирует выполнение задач на кластере.

Cluster Manager:

Управляет ресурсами кластера. Spark поддерживает несколько менеджеров: Standalone, Apache Mesos, Hadoop YARN и Kubernetes.

Executor:

Процессы, запущенные на узлах кластера, которые выполняют задачи и хранят данные в памяти.

RDD (Resilient Distributed Dataset):

Основная структура данных в Spark, представляющая собой распределенную коллекцию объектов. RDD устойчивы к сбоям благодаря механизму восстановления данных.

DataFrame и Dataset:

Более высокоуровневые API для работы со структурированными данными. DataFrame представляет собой таблицу с именованными столбцами, а Dataset — типизированную версию DataFrame.

Основные компоненты Spark:

Spark Core:

Основной модуль, который обеспечивает распределенную обработку данных и управление задачами.

Spark SQL:

Модуль для работы со структурированными данными и выполнения SQL-запросов.

Spark Streaming:

Модуль для обработки потоковых данных в реальном времени.

MLlib:

Библиотека машинного обучения, которая включает алгоритмы классификации, регрессии, кластеризации и другие.

GraphX:

Библиотека для графовой аналитики, которая позволяет работать с графами и выполнять алгоритмы, такие как PageRank.

Преимущества Apache Spark:

Скорость:

Благодаря использованию оперативной памяти, Spark работает в 10–100 раз быстрее, чем Hadoop MapReduce.

Универсальность:

Поддержка различных типов обработки данных (пакетная, потоковая, машинное обучение, графовая аналитика).

Простота интеграции:

Spark легко интегрируется с существующими системами, такими как Hadoop, Kafka и облачными платформами.

Активное сообщество:

Spark имеет большое сообщество разработчиков и пользователей, что обеспечивает постоянное развитие и поддержку.

Примеры использования Apache Spark:

Анализ логов:

Обработка и анализ больших объемов логов веб-серверов, приложений и сетевых устройств.

Рекомендательные системы:

Анализ поведения пользователей для построения персонализированных рекомендаций (например, в интернет-магазинах или стриминговых платформах).

Обработка данных IoT:

Анализ данных с датчиков и устройств Интернета вещей (IoT) в реальном времени.

Финансовый анализ:

Обнаружение мошенничества, оценка рисков и анализ транзакций.

Машинное обучение:

Построение моделей для прогнозирования, классификации и кластеризации.

Delta lake

Delta Lake — это открытый проект, разработанный компанией Databricks, который добавляет возможности управления данными и транзакционную поддержку к существующим хранилищам данных, таким как Apache Spark. Delta Lake решает многие проблемы, связанные с обработкой больших данных, такие как обеспечение согласованности данных, управление версиями и поддержка ACID-транзакций. Это делает его мощным инструментом для построения надежных и масштабируемых data pipelines.

Основные особенности Delta Lake:

ACID-транзакции:

Delta Lake обеспечивает атомарность, согласованность, изолированность и долговечность (ACID) для операций с данными, что позволяет избежать проблем с частично завершенными или некорректными записями.

Управление версиями (Time Travel):

Delta Lake сохраняет историю изменений данных, что позволяет "путешествовать во времени" и восстанавливать предыдущие версии данных.

Масштабируемость:

Delta Lake может обрабатывать петабайты данных, используя распределенные вычисления Apache Spark.

Поддержка структурированных и полуструктурированных данных:

Delta Lake работает с табличными данными, поддерживая форматы Parquet и JSON.

Оптимизация производительности:

Delta Lake использует индексацию, кэширование и оптимизацию запросов для ускорения обработки данных.

Интеграция с экосистемой Spark:

Delta Lake полностью интегрирован с Apache Spark, что позволяет использовать его в существующих Spark-приложениях.

Преимущества Delta Lake:

Надежность данных:

Поддержка ACID-транзакций предотвращает потерю данных и обеспечивает их согласованность.

Управление версиями:

Возможность восстановления предыдущих версий данных упрощает отладку и аудит.

Производительность:

Оптимизация запросов и использование индексов ускоряют обработку данных.

Гибкость:

Delta Lake поддерживает как пакетную, так и потоковую обработку данных.

Интеграция с Spark:

Delta Lake легко интегрируется с существующими Spark-приложениями, что упрощает переход на эту технологию.

Заключение

Большие данные стали неотъемлемой частью современного мира, оказывая значительное влияние на бизнес, науку, технологии и общество. Их объем, разнообразие и скорость генерации требуют применения специализированных методов и инструментов для эффективной обработки, анализа и хранения.

Современные инструменты, такие как Apache Hadoop, Apache Spark, Delta Lake и KNIME, предоставляют мощные возможности для работы с огромными объемами информации. Hadoop обеспечивает распределенное хранение и обработку данных, Spark ускоряет выполнение задач за счет использования оперативной памяти, Delta Lake добавляет надежность и управляемость к data lakes, а KNIME упрощает анализ данных благодаря визуальному интерфейсу. Каждый из этих инструментов имеет свои преимущества и области применения, что позволяет выбирать наиболее подходящее решение в зависимости от задач и характеристик данных.

Однако работа с большими данными сопряжена с рядом сложностей, таких как обеспечение качества данных, масштабируемость, высокая стоимость инфраструктуры и необходимость соблюдения юридических и этических норм. Для успешной реализации проектов в области больших данных требуется не только современная технологическая база, но и квалифицированные специалисты, способные эффективно использовать эти инструменты.

В заключение можно отметить, что большие данные открывают огромные возможности для инноваций и принятия обоснованных решений. Их правильная обработка и анализ позволяют выявлять скрытые закономерности, прогнозировать тенденции и оптимизировать бизнес-процессы. В будущем развитие технологий больших данных будет продолжаться, что приведет к появлению новых инструментов и методов, способных еще больше упростить и ускорить работу с информацией.

Литература

1. Мамедли Р. Э., Казиахмедов Т. Б. - Большие данные и NoSQL базы данных – 2024
2. Макшанов А. В., Журавлев А. Е., Тындыкарь Л. Н. - Большие данные. Big Data - 2024
3. Журавлев А. Е., Макшанов А. В., Тындыкарь Л. Н. - Компьютерный анализ. Практикум в среде Microsoft Excel - 2023
4. Макшанов А. В., Журавлев А. Е., Тындыкарь Л. Н. - Современные технологии интеллектуального анализа данных - 2020
5. Виктор Майер-Шенбергер - Большие данные - 2021