Подпишитесь на DeepL Pro для редактирования данного документа.

Дополнительную информацию можно найти на странице www.DeepL.com/pro.

Технический университет Молдовы

Кафедра информатики и инженерных систем

БОЛЬШИЕ ДАННЫЕ

Краткий конспект лекций

Д-р Ион ГАНЕА

Кишинев, 2024

Общие сведения

Данные:

- *Представляют собой последовательности символов (слов) в некотором формальном или неформальном алфавите;*
- *Сами по себе данные не имеют никакого значения;*
- *Сырой материал, факты, символы, цифры, слова, картинки*
- *без самостоятельного значения, не интегрированные в контекст,*
- *без связи с другими данными или объектами.*

Информация:

- Данные, организованные и используемые в определенном контексте, становятся информацией, которая представляет собой:
- (a) Значение данных.
  - Данные, организованные и представленные систематическим образом, чтобы подчеркнуть смысл этих данных;

(b) + новые значения данных, которые могут быть получены с помощью знаний из известных, возможно полученных значений данных;

Информация (продолжение)

- (c) Каждый из новых элементов, по отношению к предварительным знаниям, содержащихся в значении символа или группы символов.
- Информация представляется в виде отчетов, статистических данных, диаграмм и т. д.

Знания

- *Сборники данных, информации, истин и принципов, накопленных с течением времени*
- *Информация по теме, запомненная и понятая, которая*
- *может быть использована при принятии решений,*
- *формирует суждения и мнения, становится знанием.*

Данные и знания

- *Имеют разные характеристики и функционируют по-разному.*
- *Появление новых фактов в базе знаний и/или установление новых связей между ними*
- *влекло за собой изменения в процессе принятия решений при решении проблем.*
- *Приводят к познанию.*

База знаний

- *(англ. Knowledge Base - KB)*
- *Означает технологию, используемую для хранения сложной информации, как структурированной, так и неструктурированной, используемой информационной системой.*
- *Термин первоначально использовался для экспертных систем, которые были первыми системами, основанными на знаниях.*

Знание

- *Совокупность данных, информации, концепций и опыта, накопленных человеком или сообществом, которые интегрированы в единую систему и используются для понимания и объяснения окружающего мира.*
- *Представляет собой как понимание фактов, явлений, отношений между ними, так и умение эффективно использовать их для решения проблем или принятия решений.*
- *Реальное положение вещей, основанное на фактах и рациональных аргументах, обоснованное логически или фактически и проверенное эмпирически или практически.*

Типы знаний

1. ***Фактическое*** *(декларативное): включает факты и конкретную информацию о событиях, объектах или концепциях.*
2. ***Процедурное****: относится к навыкам и методам, необходимым для выполнения деятельности или решения проблемы.*
3. ***Концептуальное****: предполагает понимание законов, принципов и отношений между различными сущностями или явлениями.*
4. ***Метакогнитивное****: это осознание и управление собственным процессом обучения и познания.*

Определение *BIG DATA*

- *Представляют собой чрезвычайно большие, сложные и разнообразные наборы данных, которые генерируются и собираются в ускоренном темпе из различных источников и постоянно расширяются:*
  устройства IoT, датчики, социальные сети, финансовые транзакции и многое другое.
- *Они контролируются компаниями, органами власти и другими крупными организациями.*
- *Подвергаются обширному анализу на основе использования алгоритмов машинного обучения.*
- *Из-за их экспоненциального роста и сложности их трудно обрабатывать и эффективно использовать с помощью*

Большие данные и сложность

- *Массовый сбор данных в исследуемой области может привести к проблемам, связанным с объемом, изменчивостью и сложностью данных.*
- *Анализ таких больших наборов данных требует передовых методов обработки, таких как машинное обучение и анализ данных.*
- *С постоянным сбором данных из различных экспериментов и организаций возникает проблема эффективного управления и анализа больших объемов данных.*
- *Технологии анализа массивов данных, такие как методы эффективного хранения, параллельной обработки и методы уменьшения размерности данных, становятся все более важными в этом контексте.*
- *Рост объема данных в определенной области привел к необходимости разработки централизованных баз данных и интегрированных платформ, которые облегчают хранение, управление и обмен данными между исследователями по всему миру.*
- *В этом смысле речь идет о хранилищах данных, векторных БД и озерах данных.*

Характеристики больших данных

- *В 2017 году наборы данных считались большими данными , если они соответствовали «17 V»*
- *Первые 5 «V» считаются фундаментальными:*

1) **VOLUME** (*ОБЪЕМ*) – *огромный объем генерируемых, собираемых и обрабатываемых данных;*
2) **СКОРОСТЬ** (*V*) – *быстрый темп, с которым они генерируются и обрабатываются;*
3) **VARIETY** (*РАЗНООБРАЗИЕ*) – *разнообразие типов данных: структурированные, неструктурированные, полуструктурированные;*
4) **VERACITY** (*ПРАВДИВОСТЬ / Достоверность*) – *степень качества, точности и надежности предоставляемых данных;*
5) **VALUE** (*ЦЕННОСТЬ*) – *Важность и значимость информации, извлеченной из данных путем их анализа и интерпретации.*

6) **VISUALISATION (ВИЗУАЛИЗАЦИЯ)**: *Способность эффективно представлять и интерпретировать данные визуально.*

7) **ВАЛИДНОСТЬ (VALIDITY)**: *Степень точности и соответствия данных реальности.*

8) **VOLATILITY (ВОЛАТИЛЬНОСТЬ)**: *Скорость, с которой данные изменяются или обновляются.*

9) **VISCOSITY (ВЯЗКОСТЬ)**: *Степень связности и совместимости между различными источниками данных.*

10) **ВИРАЛЬНОСТЬ**: *Способность данных быстро и широко распространяться.*

11) **VENUE (МЕСТО)**: *Важность и значимость географического местоположения, связанного с данными.*

12) VARIABILITY (ВАРИАБИЛЬНОСТЬ): Степень изменения и вариации в данных.

13) СЛОВАРНЫЙ ЗАПАС (VOCABULARY): Стандартизация и согласованность в терминологии и языке.

14) VAGUENESS (НЕДОСТАТОЧНОСТЬ): Степень неопределенности или недостаточной ясности данных.

15) VERBOSITY (МНОГОСЛОВНОСТЬ): Степень детализации и исчерпываемости информации.

16) VOLUNTARINESS (ДОБРОВОЛЬНОСТЬ): Добровольный характер предоставления данных пользователями.

17) VERSATILITY (УНИВЕРСАЛЬНОСТЬ): Способность использовать и применять данные в различных контекстах и для различных целей.

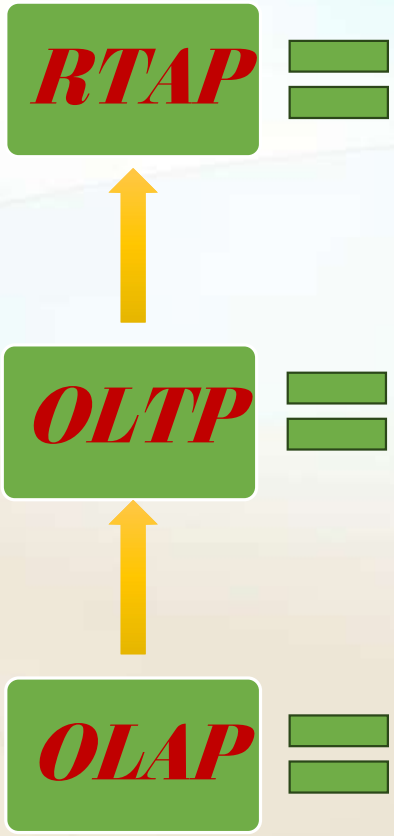
Основные компоненты и экосистема Big Data

- *Методы анализа данных*:
  - *Машинное обучение* (Machine Learning - ML) и *Глубокое обучение* (Deep Learning - DL);
  - *Обработка естественного языка* (Natural Language Processing - NLP);
  - *Технологии обработки больших объемов данных*, такие как бизнес-аналитика, облачные вычисления, базы данных, хранилища данных (Data Warehouse – DW), озера данных (Data Lake);
- *Визуализация*, такая как диаграммы, графики и другие формы отображения данных.

Аспекты курса «Большие данные»

- *Этот курс посвящен техническим и практическим аспектам управления и анализа больших наборов данных,*
  - *в том числе на технологиях и методах, которые являются ключевыми в области Big Data:*
- ***Искусственный интеллект**: применение ИИ для больших данных.*
  - *Техники ИИ для анализа и интерпретации больших данных. Использование нейронных сетей и глубокого обучения в Big Data.*
- ***Data Mining**: методы и алгоритмы извлечения знаний из больших данных.*
  - *Внедрение и оптимизация техник Data Mining на массивах данных.*
  - *Платформы и инструменты для Data Mining в Big Data (например, Hadoop, Spark).*
- ***Машинное обучение**: масштабируемые алгоритмы машинного обучения для больших данных.*
  - *Техники обучения и оценки моделей на больших наборах данных.*
  - *Использование машинного обучения в больших данных для прогнозирования и предсказания.*
- ***Text Mining**: обработка и анализ больших объемов текстовых данных.*
  - *Масштабируемые алгоритмы текстового анализа. Применение текстового анализа в*

Увеличение объема данных



Расширенная обработка в реальном времени
(Архитектура и технология Big Data)
Обработка аналитики в реальном времени
(Архитектура и технологии Big Data)

Онлайн-обработка транзакций (SGBD)
Онлайн-обработка транзакций (DBMS)

Аналитическая обработка онлайн (Хранение данных)
Онлайн-аналитическая обработка (хранение данных)

RTAP vs OLTP vs OLAP

- RTAP – технология, позволяющая анализировать данные в режиме реального времени, то есть по мере их генерации или получения. Используется в ситуациях, когда быстрые ответы имеют критическое значение, например при мониторинге систем, обнаружении мошенничества или анализе поведения пользователей в режиме реального времени.
- OLTP – ориентирована на управление операционными транзакциями в режиме реального времени, например обновление базы данных после продажи или регистрации.
- OLAP – сосредоточен на анализе исторических данных, позволяя пользователям выполнять сложные запросы по большим наборам данных, но не обязательно в режиме реального времени.

Компоненты Data Science

- ***Data Science*** *(DS) – это обширная область, которая включает в себя анализ данных, сочетающий математику, статистику, информатику и машинное обучение для получения ценной информации из данных, прогнозирования и автоматизации процессов.*

- ***Анализ данных*** *– исследование, интерпретация и визуализация данных с целью извлечения полезной информации и поддержки принятия бизнес-решений.*

Сосредоточен на обнаружении закономерностей и взаимосвязей в данных с помощью статистических методов и других техник интерпретации.

- ***Машинное*** *обучение (ML) – относится к подразделу науки о данных, который включает в себя разработку алгоритмов и моделей, способных учиться на данных и автоматически делать прогнозы или принимать решения, без явного программирования для каждого конкретного случая.*

- ***Инженерия данных (DE)*** *- создание, управление и оптимизация инфраструктур данных, которые позволяют эффективно собирать, хранить и обрабатывать большие объемы данных.*

Цель состоит в том, чтобы обеспечить доступность, чистоту и удобство использования данных для аналитиков и исследователей в области науки о данных.

Возможности *Big Data*

- *Организации могут преобразовывать данные в ценную информацию, которая позволяет им принимать правильные решения и получать конкурентное преимущество.*
- *На основе анализа собранных данных обычно делаются прогнозы по организациям, исходя из их поведения в различных ситуациях и с использованием передовых аналитических методов.*
- *Можно выявить тенденции, потребности и поведенческие изменения этих организаций.*

Различия и взаимосвязи

- *Анализ данных сосредоточен на исследовании и визуализации данных с целью извлечения полезной информации.*
- ***DS*** *– более широкая область, включающая анализ данных, но также интегрирующая машинное обучение и инженерию данных для создания прогнозных моделей и автоматических решений.*
- ***ML*** *– подмножество науки о данных, сосредоточенное на алгоритмах, которые обучаются на данных и делают автоматические прогнозы.*
- ***DE*** *– отвечает за инфраструктуру, необходимую для эффективного сбора, хранения и обработки данных, которые используют аналитики и специалисты по данным.*

Явления
и
моделирование

Понимание явлений и моделирование

- *Понимание сложных явлений:*
  - *Применяя интеллектуальный анализ данных, исследователи могут выявлять модели и корреляции, которые могут быть труднозаметны при использовании традиционных методов, что помогает глубже понять исследуемую область и решить актуальные проблемы.*
- *Моделирование, симуляции и прогнозирование поведения объектов:*
  - *Разработка сложных математических моделей и алгоритмов машинного обучения для моделирования процессов может помочь предсказать поведение этих объектов в зависимости от различных факторов, таких как условия окружающей среды, доступные ресурсы и входные параметры.*
- *Эти факторы способствуют разработке оптимальных стратегий и оценке воздействия изменений на системы.*

Модель

- *Модель – это упрощенное или абстрактное представление реальной системы, процесса, концепции или явления.*
- *Модели используются для понимания, анализа, объяснения или предсказания поведения системы.*
- *Типы моделей:*
  - *Математические (уравнения, формулы);*
  - *Графические (диаграммы, схемы, графики);*
  - *Физические (машины или прототипы);*
  - *Символьные (коды, программы);*
  - *Вербальные (концептуальные описания).*

Моделирование

- *Процесс построения, анализа и использования модели для решения проблем, получения информации или принятия решений.*
- *Это методология, с помощью которой сложная система преобразуется в форму, которую легче управлять, изучать или оптимизировать.*
  - *Модель vs Моделирование:*
- *Модель – это инструмент,*
- *Моделирование – это процесс, с помощью которого этот инструмент создается и используется.*

Модели

- *Разработанные модели могут способствовать принятию обоснованных решений в области управления ресурсами и операциями.*
- *Машинное обучение и искусственный интеллект: интеллектуальные технологии могут использоваться для анализа больших наборов данных, генерируемых различными процессами, с целью выявления скрытых моделей, корреляций и тенденций.*
- *Они также могут способствовать разработке алгоритмов ранней диагностики проблем или других критических аспектов, связанных с оптимальной работой систем.*
- *Междисциплинарное сотрудничество: Эта тема предполагает тесное сотрудничество между информатиками, экспертами в области искусственного интеллекта и специалистами из различных областей.*

Применение моделей и моделирования

1. **Точные науки** (например: модели прогнозирования климата, физика, биология);
2. **Технологии** (например, проектирование машин, 3D-симуляции);
3. **Экономика и бизнес** (например, финансовые модели, прогнозы продаж);
4. **Образование и исследования** (например, модели обучения, анализ данных);
5. **Управление бизнесом и принятие решений** (например, модели SSD, бизнес-модели).

Этапы процесса моделирования

1. ***Определение цели***: *установление цели, для которой создается модель.*
2. ***Сбор данных***: *определение соответствующих параметров и переменных.*
3. ***Разработка модели***: *выбор типа модели и ее построение.*
4. ***Валидация модели***: *сравнение полученных результатов с реальными для проверки точности.*
5. ***Использование моделирования***: *применение моделирования для анализа, моделирования или прогнозирования.*

Междисциплинарное общение

- *Трансдисциплинарная коммуникация и обмен знаниями имеют решающее значение для прогресса исследований в этой области.*
- *Интеграция информатики и интеллектуальных технологий оказывает значительное влияние на устойчивость и эффективность в различных секторах.*
- *С помощью соответствующих инструментов можно получить более глубокое понимание процессов и разработать более эффективные стратегии для устойчивого управления ресурсами.*
- *Оптимизация ресурсов: использование технологий анализа данных для оптимизации использования ресурсов, таких как вода, энергия и материалы.*
- *С помощью этих методов исследователи и лица, принимающие решения, могут способствовать устойчивому развитию, адаптируясь к изменяющимся условиям и снижая воздействие на окружающую среду и ресурсы.*

Цели *Big Data*

- ***Бизнес-возможности***: *Анализ данных может привести к обнаружению новых тенденций, улучшению обслуживания клиентов, оптимизации процессов и созданию новых продуктов.*
- ***Более эффективные решения***: *благодаря анализу данных компании могут принимать более обоснованные и быстрые решения.*
- ***Инновации***: *Big Data можно использовать для разработки новых продуктов и услуг, а также для улучшения существующих.*

Примеры использования *Big Data*

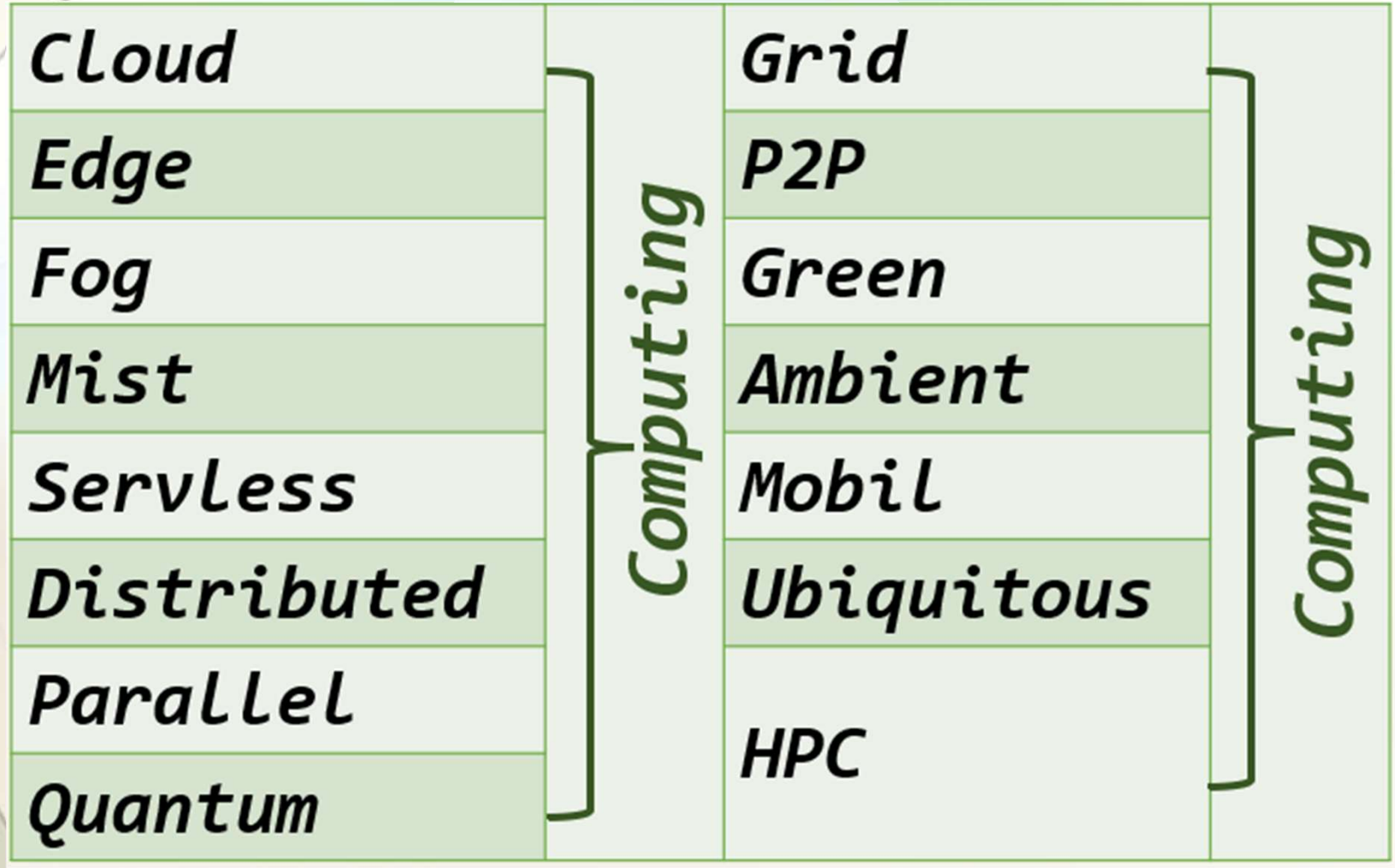
- ***Электронная коммерция**: персонализированные рекомендации по продуктам, выявление мошенничества, оптимизация цен.*
- ***Здравоохранение**: анализ медицинских данных для открытия новых методов лечения, мониторинг состояния здоровья пациентов.*
- ***Финансы**: выявление отмывания денег, оценка кредитного риска, оптимизация инвестиционного портфеля.*
- ***Маркетинг**: сегментация клиентов, персонализированные маркетинговые кампании, измерение эффективности кампаний.*

*ПАРАДИГМЫ ВЫЧИСЛЕНИЙ **И** АРХИТЕКТУРЫ*

Парадигмы

- Вычислительные парадигмы и связанные с ними архитектуры предназначены для эффективного управления разнообразными требованиями современных приложений, такими как обработка данных, сокращение задержек, масштабируемость и безопасность.
- Выбор подходящей парадигмы зависит от требований приложения, таких как задержка, масштабируемость, потребление ресурсов, затраты и местоположение данных.
- Каждая парадигма имеет свою специфическую роль и может использоваться отдельно или в сочетании с другими для получения оптимальных решений.

Парадигмы


Cloud
Edge
Fog
Mist
Servless
Distributed
Parallel
Quantum
Computing
Grid
P2P
Green
Ambient
Mobil
Ubiquitous
HPC
Computing


ИНФРАСТРУКТУРА
И АРХИТЕКТУРА
БОЛЬШИЕ ДАННЫЕ

Инфраструктура Big Data

- ***Распределенные системы хранения**: эти системы могут хранить огромные объемы данных на нескольких серверах, обеспечивая масштабируемость и избыточность (Hadoop Distributed File System (HDFS), Apache Cassandra).*
- ***Процессоры массивных данных**: эти процессоры специализируются на параллельном анализе больших объемов данных (Apache Spark, Apache Flink).*
- ***Базы данных NoSQL**: эти базы данных более гибкие, чем традиционные реляционные базы данных, и могут хранить структурированные, полуструктурированные и неструктурированные данные. (MongoDB, Apache Hbase).*
- ***Инструменты анализа**: позволяют визуализировать и анализировать данные, а также создавать прогнозные модели. (Tableau, Power BI, Python с библиотеками Pandas, NumPy, Scikit-learn).*
- ***Облачные платформы**: облачные платформы предоставляют гибкую и масштабируемую инфраструктуру для управления большими данными. Примеры: Amazon Web Services (AWS), Microsoft Azure, Google Cloud Platform (GCP).*

Архитектура *Big Data*

- *Концептуальная модель, которая облегчает обсуждение требований безопасности в Big Data и вводит терминологию, используемую в этой области.*

Общие компоненты:

- ***Генерация данных**: источники данных могут быть различными, от датчиков и устройств IoT до веб-приложений и социальных сетей.*
- ***Сбор данных**: данные собираются и хранятся в хранилище данных.*
- ***Обработка данных**: данные очищаются, преобразуются и интегрируются для анализа (процессы ETL).*
- ***Анализ данных**: Данные анализируются с целью выявления закономерностей и тенденций.*
- ***Визуализация данных**: Результаты анализа представляются в понятном формате, обычно в виде графиков и диаграмм.*

Жизненный цикл анализа данных

- ***Сбор данных*** *– процесс поиска данных и определения их достаточности для проведения анализа.*
- ***Подготовка данных*** *– этот этап может включать в себя множество задач по преобразованию данных в формат, подходящий для инструмента, который будет использоваться.*
- ***Выбор модели*** *– этот этап включает выбор метода анализа, который наилучшим образом отвечает на вопрос с помощью имеющихся данных.*
- ***Анализ данных*** *– процесс тестирования модели по отношению к данным и определения, являются ли модель и проанализированные данные надежными. Смогли ли вы ответить на вопрос с помощью выбранного инструмента?*
- ***Презентация результатов*** *– процесс доведения результатов до сведения лиц, принимающих решения.*
- ***Принятие решений*** *– Руководители организации включают новые знания в общую стратегию. Процесс начинается заново со сбора данных.*

Вызовы и перспективы

- ***Безопасность данных**: защита персональных данных и конфиденциальность имеют первостепенное значение.*
- ***Этика данных**: ответственное использование данных для предотвращения дискриминации и других негативных последствий.*
- ***Качество данных**: данные должны быть чистыми и точными, чтобы получить надежные результаты.*
- ***Таланты**: для управления и анализа больших объемов информации необходимы специалисты в области данных.*

Основные типы анализа больших данных

- *Существует четыре основных типа:*

  1) *Дескриптивный анализ;*

  2) *Диагностический анализ;*

  3) *Прогностический анализ;*

  4) *Прескриптивный анализ.*

Дескриптивный анализ

- *Эти инструменты дают фундаментальное понимание эффективности маркетинга путем обобщения предыдущих данных.*
- *Популярные варианты включают Apache Spark и Google BigQuery.*
- *Они отлично справляются с ответом на вопросы «что произошло».*
- *Например, они могут показать, какие маркетинговые каналы привлекли больше всего трафика на сайт, или определить пиковые периоды приобретения клиентов.*
- *Эти фундаментальные знания открывают путь для дальнейшего анализа.*

Диагностический анализ

- *Опираясь на описательные перспективы, диагностические инструменты углубляются, чтобы выявить «почему» за маркетинговыми тенденциями.*
- *Представьте себе резкое снижение активности в социальных сетях.*
- *Диагностические инструменты могут проанализировать настроения клиентов, выявить проблемный контент или определить технические проблемы, влияющие на охват кампании.*
- *Выделяя основные причины, маркетологи могут устранить проблемы с производительностью.*

Прогнозная аналитика

- *Используя силу исторических данных и статистических моделей,*
- *инструменты прогнозирования предсказывают будущее поведение клиентов и тенденции рынка.*
- *Ruler Analytics является ведущим игроком в этой области.*
- *Специалисты по маркетингу могут использовать эти инструменты для персонализации клиентского опыта,*
- *для оптимизации таргетинга кампании и прогнозирования будущего спроса на продукты и услуги.*

Прескриптивный анализ

- *используя информацию из предиктивного анализа,*
- *прескриптивные инструменты рекомендуют конкретные действия для оптимизации маркетинговых усилий.*
- *Специалисты по маркетингу могут использовать эти инструменты*
- *для определения наиболее эффективных маркетинговых каналов для охвата целевой аудитории*
- *предложить оптимальные ценовые стратегии и*
- *и персонализировать рекомендации по продуктам для отдельных клиентов.*

Специфические технологии **BIG DATA**

Ключевые технологии

- *Hadoop, Spark, TensorFlow, облачные вычисления, NoSQL*.
- *Распределенное программирование с MapReduce и Hadoop*
- *Расширенная обработка с Apache Spark*.
- *RDD – Resilient Distributed Datasets.*

Специфические технологии, используемые в анализе больших данных

- ***Hadoop**: открытая платформа, позволяющая хранить и обрабатывать данные на распределенных кластерах компьютеров.*
- ***Spark**: фреймворк для обработки данных в памяти, гораздо более быстрый, чем Hadoop, идеально подходящий для приложений реального времени.*
- ***NoSQL**: базы данных, которые могут хранить неструктурированные или полуструктурированные данные, такие как тексты, изображения и графические данные. Примеры: Neo4j, MongoDB, Cassandra.*
- ***Облачные вычисления**: такие платформы, как AWS, Azure и GCP, предоставляют инфраструктуру, необходимую для хранения и обработки данных в больших объемах.*
- ***Машинное обучение** и **глубокое обучение**: передовые алгоритмы, которые позволяют системам учиться на данных и делать прогнозы.*
- ***Хранилище данных**: хранилища данных, которые хранят огромные объемы интегрированной информации из различных источников, облегчая анализ.*
- ***BI**: инструменты, позволяющие визуализировать и анализировать данные для*

Характеристика	База данных	Хранилище данных	Озеро данных	Море данных
Цель	Хранение и управление структурированными данными для операционных приложений.	Расширенный анализ, отчетность, бизнес-аналитика.	Хранение необработанных, структурированных и неструктурированных данных для последующего анализа.	Неструктурированное хранение и массовый сбор данных из различных источников.
Тип данных	Структурированные (например, таблицы SQL).	Структурированные и полуструктурированные (оптимизированные для быстрых запросов).	Структурированные, полуструктурированные и неструктурированные.	Преимущественно неструктурированные, без четкой организации.
Структурирование	Организованная (четко определенные схемы).	Организованная (основанная на концептуальных и реляционных моделях).	Минимально организованная, метаданные — опциональные.	Отсутствие четкой структуры или организации.
Доступность	Быстрая для транзакционных операций.	Оптимизирована для сложных аналитических запросов.	Требует дополнительной обработки для анализа.	Труднодоступна или трудноанализируема без предварительной обработки.
Архитектура	Основана на SQL, транзакционная (OLTP).	Основана на OLAP.	Распределенная, масштабируемая архитектура (Hadoop, Spark).	Хаотичная, без определенной архитектурной модели.
Хранение	Ограниченная емкость, оптимизированная для транзакций.	Большая емкость, оптимизированная для анализа.	Очень большая емкость, горизонтально масштабируемая.	Чрезвычайно большая емкость, но без оптимизации.
Связь между данными	Реляционная, четко определенная.	Реляционная, с денормализованными схемами.	Отсутствие явных отношений, но они могут быть определены позже.	Данные не связаны и не организованы.
Предварительная	Требует проверки и очистки при хранении.	Данные очищаются и преобразуются перед	Необработанные данные, преобразование	

Выбор подходящей технологии

зависит от:

- *Объема и типа данных: структурированные, неструктурированные, полуструктурированные данные.*
- *Скорость обработки: необходимость получения результатов в режиме реального времени или в автономном режиме.*
- *Сложность анализа: от простых описательных статистических данных до сложных моделей машинного обучения.*
- *Бюджет и ресурсы: Стоимость приобретения и обслуживания технологии, а также наличие квалифицированного персонала.*

Примеры успешных компаний

- ***Netflix**: использует большие данные для персонализации рекомендаций по фильмам и сериалам для каждого пользователя, тем самым повышая лояльность клиентов.*
- ***Amazon**: анализирует данные клиентов, чтобы предсказывать спрос, оптимизировать цены и персонализировать опыт покупки.*
- ***Uber**: использует большие данные для оптимизации маршрутов, оценки времени ожидания и установления динамических цен.*
- ***Google**: использует Big Data для улучшения результатов поиска, разработки новых продуктов (таких как Google Assistant) и персонализации рекламы.*
- ***В медицинской отрасли**: Big Data используется для открытия новых методов лечения, персонализации лечения для каждого пациента и повышения операционной эффективности больниц.*

Влияние больших данных на общество

- *Улучшение государственных услуг: оптимизация дорожного движения, управление отходами, предотвращение преступности.*
- *Разработка новых продуктов и услуг: повышение инновационности и конкурентоспособности.*
- *Персонализация клиентского опыта: повышение удовлетворенности клиентов и лояльности к брендам.*
- *Улучшение процессов принятия решений: принятие более обоснованных и быстрых решений.*
- *Создание новых бизнес-возможностей: Появление новых бизнес-моделей, основанных на данных.*

Тенденции в области больших данных

- *IoT и IIoT: генерация еще большего количества данных от подключенных устройств.*
- *ИИ и МО: Разработка более сложных алгоритмов и их применение в все более разнообразных областях.*
- *Edge Computing: обработка данных у источника, сокращение задержек и затрат.*
- *Квантовые вычисления: потенциал для решения сложных задач, превышающих возможности классических компьютеров.*
- *Этика данных: Разработка этических рамок для использования данных и защиты частной жизни.*

Распределенное программирование

с

HADOOP и MAP REDUCE

Распределенное программирование

- *Модель программирования, которая распределяет вычислительные задачи между несколькими рабочими станциями (узлами) в сети.*
- *Это позволяет чрезвычайно быстро обрабатывать огромные объемы данных, что невозможно на одном компьютере.*
- *MapReduce и Hadoop – две ключевые технологии, которые революционизировали подход к распределенному*

Модель программирования *MapReduce*

- ***MapReduce*** *– это* ***парадигма (модель) программирования,*** *используемая для параллельной и распределенной обработки больших объемов данных в кластере Hadoop.*
- *Эта модель позволяет распределять вычислительные процессы на тысячи машин, гарантируя масштабируемость и эффективность.*
- *Идеально подходит для массивных неструктурированных наборов данных, которые необходимо обрабатывать распределенным образом.*
- *Система MapReduce* ***управляет распределенными серверами,*** *обеспечивает параллельное выполнение различных задач,*

- ***Управляет коммуникацией и передачей данных между различными компонентами системы и обеспечивает поддержку в случае возникновения ошибок в системе.***
- ***MapReduce можно реализовать на нескольких языках – Python, Java, C++, Scala и т. д.***
  - ***Функциональность:***
- ***Hadoop MapReduce – реализован на Java (классический фреймворк).***
- ***Streaming API – позволяет писать код Map и Reduce на Python, Perl, Ruby и т. д., а Hadoop занимается распределением задания.***
- ***Библиотеки в Python – mrjob, PySpark ...***

Мар (Сопоставление)

- *Этап сопоставления состоит в обработке входных данных, которые разделяются на небольшие сегменты и обрабатываются параллельно.*
- *Каждый сегмент анализируется функцией «Мар», которая присваивает значение каждому ключу. Ключ извлекается или генерируется на основе каждого элемента данных, а значение присваивается в зависимости от того, что именно обрабатывается.*
- *Таким образом, создается пара «ключ-значение».*
- *Этот процесс можно рассматривать как своего рода классификацию или группировку данных на основе ключа.*

Компоненты *MapReduce*

- ***MapReduce*** *– простой, но мощный модель программирования, которая разделяет сложную задачу на* ***две*** *более простые* ***подзадачи****:*
- ***Map****: каждый элемент из исходного набора данных обрабатывается индивидуально, в результате чего получается список пар* ***«ключ-значение»****.*
- ***Reduce****: Пары ключ-значение, сгенерированные на этапе Map,* ***группируются по ключу****, а затем* ***обрабатываются****, как правило, с помощью операции агрегирования.*

Пример: Предположим, что мы хотим определить релевантные слова в очень большом текстовом файле.

- ***MAP****: Каждая строка в файле разделяется на слова, и каждое слово становится парой ключ-значение* ***(слово, 1)****.*
- ***REDUCE****: Все пары с одинаковым ключом (одинаковым словом) группируются, а значения (количество вхождений) суммируются*

Hadoop: фреймворк для реализации MapReduce

- *Hadoop – это открытый фреймворк, который реализует модель MapReduce и предоставляет надежную платформу для обработки распределенных данных.*
- *Это де-факто стандарт для обработки больших данных.*

ОСНОВНЫЕ КОМПОНЕНТЫ:

- *Hadoop Distributed File System (HDFS): распределенная файловая система, оптимизированная для хранения больших объемов данных на компьютерных кластерах.*
- *YARN (Yet Another Resource Negotiator – Еще один переговорщик по ресурсам): Фреймворк управления ресурсами, который позволяет выполнять как приложения MapReduce, так и другие типы приложений.*
- *MapReduce: движок выполнения, реализующий модель MapReduce.*

Распределенная файловая система Hadoop (HDFS)

- *Распределенная файловая система Hadoop (подпроект Apache Hadoop) – это распределенная файловая система, разработанная для работы на базовом оборудовании.*

Особенности:

- *Высокая отказоустойчивость и возможность внедрения на недорогом оборудовании.*
- *Обеспечивает высокоскоростной доступ к данным приложения и подходит для приложений с большими наборами данных.*
- *Обеспечивает потоковый доступ к данным файловой системы.*
- *Первоначально был создан как инфраструктура для проекта веб-поисковой системы Apache Nutch.*
- *Адрес: https://hadoop.apache.org/hdfs/.*

Кластер HDFS

- Самая простая версия кластера HDFS имеет два типа демонов:
- NAMENODE и DATANODE в архитектуре типа «мастер-раб».
- HDFS предоставляет файлы, которые хранятся в виде блоков по 128 МБ на Datanodes.
- Обычно в кластере есть только один Namenode, чья роль заключается в хранении метаданных из файловой системы.
- Он регулирует доступ пользователей к файлам, а также такие операции, как открытие, закрытие и именование файлов.
- В то же время он поддерживает отображение блоков, находящихся на Datanodes.
- Datanodes хранят все блоки, из которых состоит файловая система, и отвечают за управление запросами на запись и чтение.

YARN

- *Компонент Hadoop, отвечающий за вычисления.*
- *YARN делит свою функциональность между тремя типами демонов:*
  a) *ResourceManager;*
  b) *NodeManager;*
  c) *ApplicationMaster.*
- *Архитектура YARN работает по принципу «мастер/раб»*
- *Кластер имеет един ResourceManager, который обслуживает потребности ресурсов (памяти и CPU) для всего кластера.*
- *NodeManager распределены по всем хостам кластера.*
  - *ResourceManager эквивалентен планировщику CPU ОС;*
  - *NodeManagers – процессы системы, которые требуют времени процессора, концепция, также известная как Datacenter as a Computer.*

Преимущества использования Hadoop и MapReduce

- ***Масштабируемость**: возможность обрабатывать произвольные наборы данных большого размера путем добавления новых узлов в кластер.*
- ***Надежность**: отказоустойчивость, гарантирующая, что данные не будут потеряны в случае сбоя одного из узлов.*
- ***Экономичность**: использование стандартного оборудования и программного обеспечения с открытым исходным кодом.*
- ***Простота использования**: простой и интуитивно понятный интерфейс программирования.*

Применение MapReduce и Hadoop

- ***Анализ веб-данных**: индексация поисковых систем, анализ настроений, рекомендации по продуктам.*
- ***Биоинформатика**: анализ генетических последовательностей, открытие новых лекарств.*
- ***Финансы**: обнаружение мошенничества, анализ рисков, оптимизация портфелей.*
- ***Социальные сети**: анализ настроений, выявление трендов, рекомендации по контенту.*

Где подходит

- *MapReduce подходит для обработки и анализа очень больших объемов данных (Big Data), распределенных по нескольким узлам в кластере.*
- *Он основан на двух основных этапах:*
  - *Map (фрагментация и обработка данных) и*
  - *Reduce (агрегирование и объединение результатов).*
- *Идеально подходит для больших, простых и параллельных пакетных обработок, которые масштабируются на сотни/тысячи узлов.*

Ситуации и области

- **Обработка данных в больших масштабах**
  - *Лог-файлы с веб-серверов;*
  - *Данные, генерируемые социальными сетями;*
  - *Массовые финансовые транзакции;*
  - *Биологические или геномные последовательности.*
- *Поиск и индексирование*
  - *Создание поисковых систем (Google первоначально ввел его для индексации веб-сайтов);*
  - *Генерация и обновление распределенных индексов.*
- *Агрегации и простые статистические данные*
  - *Подсчет вхождений (например, «Слова в тексте» → word count);*
  - *Расчет средних, минимальных, максимальных значений, распределений;*
  - *Генерация отчетов на уровне организации.*

- *Обработка неструктурированных или полуструктурированных данных*
  - *Текстовый майнинг, извлечение частотных терминов;*
  - *Анализ логов и кликстримов;*
  - *Преобразование больших файлов (JSON, XML, CSV).*
- *Распределенная и отказоустойчивая обработка*
  - *Задачи, при которых данные не могут быть обработаны на одном компьютере;*
  - *Когда требуется автоматическое масштабирование и восстановление в случае сбоя.*

Ограничения и соображения

- ***Задержка**: для приложений реального времени MapReduce может быть слишком медленным.*
- ***Сложность**: Настройка и управление кластером Hadoop может быть сложной задачей.*
- ***Ограниченная модель программирования**: MapReduce оптимизирован для определенных типов задач и может быть менее подходящим для других.*

Не подходит

- *Для обработки в реальном времени (пакетная обработка, медленная для потоковой передачи);*
- *Для сложных итеративных анализов (например, передовые алгоритмы ML или сложные графы)*
- *→ здесь более эффективны Apache Spark или специализированные системы (Neo4j, TensorFlow и т. д.).*

Сложный пример MapReduce

- ***Анализ журналов веб-сервера**: подробное описание сценария, в котором задание MapReduce анализирует журналы веб-сервера, чтобы определить самые посещаемые страницы или источники трафика.*
- *Поток **MapReduce** для этого примера:*
  - ***Map** может извлекать URL-адреса и IP-адреса из журналов, а **Reduce** может подсчитывать частоту этих адресов.*

Пример программы

```
pip install mrjob # Установка библиотеки MRJob
from mrjob.job import MRJob
from mrjob.step import MRStep
import re
# Определяем класс для задания MapReduce
class WordCount(MRJob):
   def steps(self):
     return [
         MRStep(mapper=self.mapper_get_words,
        reducer=self.reduc     er_count_words)
     ]
```

```
# Функция отображения, которая разбивает строки на слова
def mapper_get_words(self, _, line):
# Удаляем все символы, которые не являются буквами или цифрами
   words = re.findall(r"\w+", line.lower())
   for word in words:
      yield word, 1
# Функция сокращения, которая суммирует количество вхождений каждого слова
def reducer_count_words(self, word, counts):
      yield word, sum(counts)
# Запуск задания
if __name__ == '__main__’:
       WordCount.run()
```

- ***PageRank*** *(алгоритм Google для ранжирования веб-страниц)*
- *Map → распределяет «оценку» страницы по всем внешним ссылкам;*
- *Reduce → суммирует баллы, полученные каждой страницей, и обновляет рейтинг.*
  Используется в поисковых системах и анализе больших графов.

Статистика по логам

- *Map → извлекает поля из файлов логов (IP, URL, время доступа и т. д.);*
- *Reduce → рассчитывает количество посещений на страницу, среднее время отклика, распределение трафика.*
- *Мониторинг веб-сайтов, кибербезопасность, оптимизация сетей.*

- **Распределенная сортировка** (Distributed Sort)
  - Map → считывает данные и генерирует пары (ключ, значение);
  - Shuffle + Reduce → перегруппировывает и сортирует на глобальном уровне.
    Необходимо для конвейеров предварительной обработки больших данных.

- **Распределенное соединение** (Compuneri pe seturi de date mari)
  - Map → выдает пары (общий ключ, значение) из двух больших таблиц;
  - Reduce → объединяет записи с одинаковым ключом.
    Анализ Big Data в стиле SQL, когда данные не помещаются в одну базу данных.

- **Inverted Index** (Обратный индекс для текстового поиска)
  - Map → для каждого слова в документе выдает (слово, ID_документа);
  - Reduce → создает список документов, в которых встречается это слово.

  - Основа для поисковых систем, систем рекомендаций, текстового

- *Геномный анализ / Биоинформатика*
- *Map → обрабатывает последовательности ДНК/РНК в фрагменты;*
- *Reduce → объединяет результаты для поиска паттернов, мутаций или сходств.*
- *Персонализированная медицина, сельское хозяйство, генетические исследования.*

Расширенные технические сведения

- *Настройка и оптимизация кластера Hadoop:*
- *Установка и настройка Hadoop: подробная информация об установке Hadoop на кластере,*
- *настройке HDFS и YARN, а также о том, как установить параметры для оптимизации производительности.*
- *Параметры конфигурации: объясняет критические настройки в файлах конфигурации Hadoop, такие как размер блоков HDFS, количество реплик и параметры памяти для задач Map и Reduce.*
- *Оптимизация заданий MapReduce: подробная информация о том, как оптимизировать количество задач Map и Reduce, как минимизировать затраты на ввод-вывод и как отслеживать производительность заданий с помощью таких инструментов, как Hadoop Job Tracker.*
- *Fault Tolerance: Механизмы отказоустойчивости: Описывает, как Hadoop обрабатывает аппаратные ошибки путем репликации данных на нескольких узлах, автоматического обнаружения сбоев и повторного запуска неудачных задач Map и Reduce.*
- *Примеры сбоев и способы их устранения в Hadoop: объясняет, что происходит при сбое DataNode или NameNode и как кластер автоматически перенастраивается для поддержания работы заданий.*

Лабораторная работа

- *Настройка локального кластера: пошаговое руководство по настройке кластера Hadoop на нескольких виртуальных машинах или в контейнерах Docker, чтобы студенты могли поэкспериментировать с реальным кластером.*
- *Практическое упражнение: Написание простого задания MapReduce, такого как подсчет слов или агрегирование данных из большого файла CSV, и его запуск на настроенном кластере Hadoop.*

Источники

- *Майкл Нолл. Написание программы Hadoop MapReduce на Python. michael-noll.com*
- *Hadoop Streaming с использованием Python – задача подсчета слов. GeeksforGeeks*
- *Hadoop – Streaming. tutorialspoint.com*
- *MapReduce 101: что это такое и как начать работу. talend.com*
- *Hadoop Streaming. Практическое введение в MapReduce с Python. Модесто Мас | Блог*

Расширенная обработка

с

APACHE SPARK

Контекст и развитие Apache Spark

- *Происхождение:*
  - *Apache Spark был разработан в Калифорнийском университете в Беркли в рамках проекта AMPLab в 2009 году с целью преодолеть ограничения модели MapReduce, особенно в отношении обработки в памяти.*
- *Spark vs. MapReduce:*
  - *Spark обеспечивает производительность, в 100 раз превышающую производительность MapReduce для определенных приложений, благодаря использованию оперативной памяти для промежуточного хранения данных.*
- *Сфера применения:*
  - *Общие сценарии: Apache Spark используется для обработки больших данных (Big Data), машинного обучения, анализа в реальном времени и обработки графиков, среди прочего.*
  - *Большие объемы данных и скорость: Spark предпочтителен для приложений, требующих быстрой и итеративной обработки больших данных.*

Архитектура Apache Spark

- *RDDs (Resilient Distributed Datasets) – базовая единица данных в Spark, которая является распределенной и устойчивой к ошибкам.*
- *DAG (Directed Acyclic Graph) – создает направленный ациклический граф для оптимизации выполнения заданий.*
- *Компоненты Spark: Spark состоит из нескольких модулей, таких как Spark SQL, Spark Streaming, MLlib и GraphX.*

RDD – Resilient Distributed Datasets

- *RDD – фундаментальная структура данных в рамках фреймворка Apache Spark.*
- *Представляют собой неизменяемые коллекции объектов, распределенных по нескольким узлам в кластере.*

Основные характеристики:

- *Неизменяемость: после создания RDD не может быть изменен.*
- *Распределение: данные распределяются по нескольким узлам для параллельной обработки.*
- *Устойчивость: в случае сбоя одного из узлов RDD можно восстановить из исходных данных.*
- *Линейность: Операции над RDD линеаризуются, что означает, что их можно эффективно параллелизовать.*

Создание RDD

RDD можно создать несколькими способами:

- ***Параллелизация коллекции:***
  - *Создание RDD из локальной коллекции объектов.*
- ***Чтение файла:*** *Создание RDD из данных, хранящихся в файле (текст, CSV, JSON и т. д.).*
- ***Преобразование другого RDD:*** *создание нового RDD путем применения функции к существующему RDD.*
- ***Операции над RDD*** *(два основных типа):*
- ***Преобразования:*** *создает новый RDD из существующего, не изменяя исходные данные.*
  - *Примеры: map, filter, flatMap, reduceByKey, join.*
- ***Действия:*** *возвращают результат или запускают операцию, предполагающую фактический вычисление.*
  - *Примеры: count, collect, saveAsTextFile.*

Примеры

- *Scalaval textFile = sc.textFile("file.txt") // Создает RDD из текстового файла val counts = textFile.flatMap(line => line.split(" ")) .mapToPair(word => (word, 1)) .reduceByKey(_ + _) // Подсчитывает количество вхождений слов counts.saveAsTextFile("wordcounts.txt") Используйте код с осторожностью.*
- *Преимущества использования RDD:*
- *Производительность: RDD позволяют параллельно обрабатывать данные, что значительно повышает производительность.*
- *Устойчивость: RDD устойчивы к сбоям, что обеспечивает непрерывность обработки данных.*
- *Гибкость: RDD можно использовать для широкого спектра приложений, от пакетной обработки до потоковой передачи данных и машинного обучения.*
- *Ограничения: Неизменность: для некоторых приложений необходимость создавать новые RDD при каждом преобразовании может быть неэффективной.*
- *Сложность: Программирование с использованием RDD может быть более сложным, чем другие модели распределенного программирования. RDD являются основополагающими для понимания Spark и представляют собой прочную основу для изучения других концепций и приложений.*

Расширенные технические сведения

- *Настройка и оптимизация Apache Spark:*
- *Установка и настройка Spark: инструкции по установке Spark на кластере, настройке автономного режима или интеграции с Hadoop YARN и настройке производительности.*
- *Оптимизация заданий Spark: как настроить количество исполнителей, оптимизировать параллелизм и использовать кэширование для повышения производительности заданий.*
- *Отказоустойчивость и постоянство: как Spark восстанавливает RDD из их линии происхождения (lineage) в случае сбоя узла.*
- *Сохранение в памяти и на диске: как и когда сохранять RDD в памяти (с репликацией) или на диске для защиты данных и оптимизации выполнения.*

Практические аспекты и лабораторные работы

- *Лабораторная работа:*
  - *Настройка кластера Spark: руководство по установке и настройке локального или облачного кластера Spark.*
- *Практическое упражнение: Предложение по написанию и запуску задания Spark для обработки большого набора данных, такого как файл журналов, и применению преобразований и действий к RDD.*
- *Групповые проекты:*
  - *Предложения по проектам: Идея проекта по разработке приложения машинного обучения с помощью Spark MLlib или анализу больших наборов данных с помощью Spark SQL.*
  - *Презентация проекта: Требование к студентам представить архитектуру, реализацию и результаты своего проекта, объяснив технические проблемы и решения.*

MapReduce против Spark и Flink

MapReduce

- *Модель программирования: простая и понятная, основанная на модели map-reduce.*
- *Скорость: относительно низкая из-за промежуточной записи данных на диск.*
- *Использование: Подходит для пакетных задач, таких как анализ логов, индексирование поисковых систем.*

- ***Spark***

- *Модель программирования: RDD (Resilient Distributed Datasets), которые позволяют обрабатывать данные в памяти и повторно использовать промежуточные результаты.*
- *Скорость: Гораздо быстрее, чем MapReduce, благодаря обработке в памяти.*
- *Использование: Подходит для широкого спектра задач, включая пакетную обработку, потоковую передачу данных, машинное обучение, графы.*
- *Другие характеристики: Поддерживает SQL, машинное обучение, графы, потоковую передачу данных.*

- ***Flink***

- *Модель программирования: Непрерывные потоки данных, оптимизированные для потоковой обработки.*
- *Скорость: очень быстрый, с низкой задержкой, идеально подходит для приложений в реальном времени.*
- *Использование: Особенно подходит для потоковой обработки, такой как анализ датчиков, приложения IoT.*
- *Другие характеристики: Поддерживает как пакетную, так и потоковую обработку, оконный режим, время события.*

Пример лабораторных работ

- *Сравнение производительности с другими парадигмами обработки: Apache Spark vs. MapReduce:*
- *Представление преимуществ и недостатков двух технологий.*
- *Spark известен своей скоростью и поддержкой операций в памяти, в то время как MapReduce ценится за простоту и надежность при обработке данных на диске.*
- *Когда использовать MapReduce: Объясните ситуации, в которых MapReduce предпочтительнее (например, когда данные слишком велики для хранения в памяти) по сравнению с более современными альтернативами, такими как Spark.4.*

Групповой проект

- *Предложения по проектам:*
  - *Идеи для более крупных проектов, таких как анализ данных из социальных сетей (например, анализ настроений) или обработка и визуализация большого объема климатических данных.*
- *Презентация проекта:*
  - *Студенты представят результаты и обсудят проблемы и решения, с которыми они столкнулись при реализации MapReduce.*

MapReduce против Spark и Flink

Caracteristică	MapReduce	Spark	Flink
Model de programare	Map-reduce	RDD-uri	Fluxuri de date continue
Viteză	Lentă	Rapidă	Foarte rapidă
Utilizări	Batch	Batch,streaming, ML, grafuri	Streaming, batch
Latentă	Înaltă	Medie	Scăzută
Complexitate	Simplu	Mai complex	Complex

MapReduce против Apache Spark

Характеристика	MapReduce	Apache Spark
Модель	Пакетная обработка, 2 этапа: Map + Reduce	Распределенная обработка в памяти, DAG (Directed Acyclic Graph)
Скорость	Более медленная – чтение/запись на диск после каждого шага	Намного быстрее – хранит данные в памяти
Сложность	Требует подробного кода для каждого задания	API высокого уровня (PySpark, SparkSQL, MLlib)
Итерации (ML, графы)	Неэффективно (повторная запись в HDFS)	Эффективно (данные остаются в памяти между шагами)
Потоковая обработка	Нет (только пакетная)	Да (Spark Streaming)
Языки	Java, Python (через Hadoop Streaming) и т. д.	Python (PySpark), Scala, Java, R
Когда подходит	Простые пакетные обработки, отчеты, индексирование веб-сайтов	Машинное обучение, обработка в реальном времени, графы, сложная аналитика

Обработка потоков данных

Интеграция данных

Управление интеграцией данных

- *Процесс, необходимый для обеспечения того, чтобы данные из различных источников были объединены в единую и полезную для анализа и принятия решений форму.*
- *Учитывая характеристики данных в Big Data («V»), управление интеграцией данных требует передовых подходов и технологий для гармонизации данных и обеспечения их аналитической ценности.*
- *Хорошо спланированное управление интеграцией данных имеет решающее значение, поскольку дает организациям возможность получать ценную информацию и принимать решения в режиме реального времени.*

Гетерогенные источники данных

- *В Big Data данные поступают из широкого спектра источников.*
- *Эти источники данных имеют разные форматы и структуры.*
- *Гетерогенность данных требует сложных процессов преобразования и интеграции.*
- *Структурированные данные (например, реляционные базы данных, электронные таблицы).*
- *Неструктурированные данные (электронные письма, текстовые документы, изображения, видео).*
- *Полуструктурированные данные (JSON, XML, серверные логи).*
- *Потоки данных в реальном времени (от датчиков IoT, сенсорных сетей, финансовых транзакций).*

ETL – ELT

- *В Big Data традиционный процесс ETL может быть адаптирован и заменен ELT (извлечение, загрузка и преобразование) из-за больших объемов данных и разнообразия источников.*
- *В ELT данные сначала загружаются в платформу Big Data (например, Hadoop, Apache Spark), где они впоследствии преобразуются.*
- ***Извлечение**: получение данных из различных источников, часто осуществляемое в режиме реального времени для управления потоками данных.*
- ***Преобразование**: очистка, форматирование и стандартизация данных, включая такие процессы, как удаление несоответствующих значений или дубликатов.*
- ***Загрузка**: хранение данных в архитектуре Big Data для последующего анализа, например, в системах типа Data Lake или Data Warehouse.*

Семантическая интеграция и стандартизация данных

- *Данные из Big Data должны быть стандартизированы для эффективной интеграции:*
  - ***Онтологическое отображение:***

  Использование онтологий и семантических моделей для сопоставления схожих концепций из разных источников.
  - ***Стандартизация и нормализация:***

  Обеспечение согласованности значений в данных (например, единиц измерения, форматов данных, структур имен) для правильного анализа.

Очистка и качество данных

- *Очистка данных имеет решающее значение, учитывая, что данные в Big Data могут быть разного качества:*
  - *Удаление отсутствующих значений и ошибок: обработка отсутствующих и неверных данных.*
  - *Консолидация и дедупликация: выявление и удаление дубликатов данных.*
  - *Валидация и проверка: проверка согласованности данных и их правильности в соответствии с бизнес-правилами.*

Хранение и архитектура интегрированных данных

- ***Data Lake**: система хранения различных типов данных, не ограниченная фиксированной схемой, подходящая для хранения больших объемов необработанных данных.*
- ***Data Warehouse**: используется для хранения структурированных данных, где применяются преобразования для анализа.*
- ***Гибридные архитектуры**: интеграция между Data Lake и Data Warehouse, где необработанные данные обрабатываются в Data Lake, а преобразованные результаты перемещаются в Data Warehouse для быстрого анализа.*

Инструменты и технологии для интеграции больших данных

- *Apache NiFi: автоматизирует потоки данных, облегчая обработку в режиме реального времени между различными системами.*
- *Apache Kafka: система потоковой передачи данных, которая облегчает интеграцию данных в реальном времени из нескольких источников.*
- *Apache Spark: обеспечивает распределенную и масштабируемую обработку, идеально подходит для работы с большими объемами данных и быстрых ETL-операций.*
- *NoSQL: базы данных, такие как MongoDB, Cassandra, HBase, Neo4j, которые позволяют хранить неструктурированные и полуструктурированные данные*

Качество и управление метаданными

- *Метаданные и каталогизация данных:*
  - *Использование метаданных для описания и отслеживания источников и преобразований, через которые проходят данные,*
  - *Это позволяет обеспечить более эффективную интеграцию и отслеживаемость.*
- *Профилирование данных:*
  - *Постоянный мониторинг качества данных для обеспечения их согласованности и точности.*

Безопасность и конфиденциальность

- *Контролируемый доступ и шифрование:*
  - *Управление доступом к данным и внедрение шифрования для защиты конфиденциальных данных.*
- *Соответствие требованиям:*
  - *Обеспечение соответствия интеграции данных нормативным требованиям (например, GDPR, HIPAA).*

Автоматизация и непрерывная интеграция

- *Управление интеграцией данных можно оптимизировать с помощью автоматизации, что позволяет:*
- *Автоматизация потоков данных:*
  - ✓*Использование инструментов для управления обновлениями и обеспечения постоянной актуальности интегрированных данных.*
  - ✓*Рабочие процессы CI/CD (непрерывная интеграция/непрерывное развертывание):*
- *Интеграция изменений данных в непрерывный процесс обновления данных и приложений, которые их используют.*

Интегрированный анализ и визуализация

- *Цель интеграции данных заключается в том, чтобы обеспечить возможность их анализа в унифицированном виде:*
  - *Агрегирование данных: объединение данных в формате, доступном для анализа.*
  - *Визуализация: использование инструментов визуализации данных, таких как Tableau, Power BI или модули визуализации Spark, для интерпретации результатов анализа.*

Примеры интеграционных потоков

- *Конкретным примером потока интеграции в Big Data может служить проект «умного города»:*

1. **Сбор**: *Данные о трафике собираются с датчиков на дорогах и камер наблюдения.*
2. **Преобразование**: *данные очищаются, а пропущенные значения дополняются.*
3. **Интеграция**: *данные о дорожном движении интегрируются с данными о погоде и общественном транспорте.*
4. **Анализ**: *Проводятся прогнозные анализы для выявления и управления зонами с интенсивным дорожным движением.*
5. **Визуализация**: *данные визуализируются на интерактивной карте в режиме реального времени, доступной для органов власти и граждан для принятия обоснованных решений.*

Предварительная информация об управлении данными

- *Представляет собой первоначальную или подготовительную информацию о том, как данные будут собираться, организовываться, храниться и обрабатываться в рамках проекта по управлению данными, обеспечивая основу для планирования и внедрения эффективного управления данными.*
- *PDMI используется исследователями и проектными группами для эффективного планирования управления данными и обеспечения того, чтобы все аспекты хранения, доступа и защиты данных были задокументированы и согласованы с целями проекта.*
- *PDMI может использоваться на начальных этапах проектирования проекта для создания основы эффективного управления данными.*

Элементы PDMI

- ***Описание наборов данных**: Указывает, какие типы данных будут собираться, их форматы и предполагаемый объем.*
- ***Структура и организация данных**: информация о том, как будут структурированы данные, включая использование баз данных или других решений для хранения.*
- ***Защита данных и конфиденциальность**: определяет предварительные меры по защите данных, включая защиту конфиденциальности и соблюдение правил защиты данных.*
- ***Стратегия обмена данными и доступа к ним**: как данные будут распространяться и становиться доступными, например, путем хранения в облаке или на платформах для обмена данными.*
- ***Планирование целостности и качества данных**: процедуры обеспечения качества для предотвращения ошибок и поддержания целостности данных во время их сбора и обработки.*

Просмотр данных

Визуализация данных в Big Data

- *Процесс, необходимый для преобразования огромных объемов информации в полезные знания.*
- *Используя подходящие технологии и инструменты, организации могут получить более глубокое понимание своих данных и принимать более эффективные решения.*

Этапы:

1) *Сбор и интеграция данных*
2) *Предварительная обработка данных*
3) *Анализ данных*
4) *Визуализация данных*

1. Сбор и интеграция данных

- *Множественные источники: данные собираются из различных источников, как внутренних (базы данных, приложения), так и внешних (социальные сети, датчики, устройства IoT). Стандартизация: данные очищаются, преобразуются и стандартизируются для обеспечения согласованности и точности.Интеграция: данные из разных источников интегрируются в единый формат, готовя их к анализу.*

2. Предварительная обработка данных

- ***Очистка**: удаляются дубликаты данных, ошибки, пропущенные значения и аномалии.*
- ***Преобразование**: данные преобразуются в формат, подходящий для анализа, например, путем нормализации или дискретизации.*
- ***Обогащение**: данные обогащаются дополнительной информацией для более глубокого понимания.*

3. Анализ данных

- ***Исследование**: Используются статистические методы и методы интеллектуального анализа данных для выявления скрытых в данных закономерностей, тенденций и взаимосвязей.*
- ***Моделирование**: Создаются прогнозные модели для предсказания будущих событий или поведения.*

4. Визуализация данных

- ***Выбор данных**: Выбираются данные, необходимые для ответа на бизнес-вопросы.*
- ***Выбор типа визуализации**: Выбирается тип визуализации, наиболее подходящий для передачи информации (графики, диаграммы, карты и т. д.).*
- ***Создание визуализации**: используются инструменты визуализации для создания четких и информативных визуальных представлений.*
- ***Интерпретация**: интерпретируются визуализации для извлечения инсайтов и принятия обоснованных решений.*

Типы визуализаций

- ***Статические визуализации:***
  - ***Графики, диаграммы, карты.***
- ***Интерактивные визуализации:***
  - ***Панели инструментов, 3D-визуализации, инфографика.***
- ***Геопространственные визуализации:***
  - ***Тематические карты, пространственный***

Используемые технологии

- *Инструменты визуализации*:
- *Tableau, Power BI, Qlik Sense, Google Data Studio.*
- *Платформы Big Data*: *Hadoop, Spark, Cloud Dataflow*.
- *Языки программирования*: *Python, Go, R, SQL, Cypher*.
- *Базы данных NoSQL*: *Neo4j, HBase, Cassandra, MongoDB*;
- *Реляционные базы данных (SQL)*.

Преимущества визуализации данных в Big Data

- *Лучшее понимание данных: визуализация делает данные более понятными и интерпретируемыми.*
- *Быстрое выявление тенденций и аномалий: визуализация позволяет быстро выявлять необычные модели и поведения.*
- *Эффективная коммуникация: визуализация – это эффективный способ донести сложную информацию до широкой аудитории.*
- *Поддержка принятия решений: визуализации обеспечивают прочную основу для принятия обоснованных решений.*

Looker Studio

- *Ранее Google Data Studio*
- *Онлайн-инструмент для преобразования данных в настраиваемые, информативные отчеты и панели мониторинга.*
- *Looker Studio входит в пакет Google Analytics 360 для предприятий, а бесплатная версия доступна для частных лиц и небольших команд.*

ИИ в больших данных

- *Введение в ИИ и его связь с большими данными.*
- *Интеллектуальный анализ данных*
- *Текстовый майнинг*
- *NLP*
- *Машинное обучение*

Интеллект

Общее и всеобъемлющее определение интеллекта:

Интеллект представляет собой способность решать новые и/или сложные проблемы с помощью гибких и творческих методов, основываясь на предыдущих знаниях и опыте.

Определение интеллекта

- *Интеллект представляет собой способность легко и точно понимать*
  - *легко и точно,*
  - *эффективно и глубоко,*
  - *улавливать суть,*
  - *решать новые ситуации или проблемы на основе ранее накопленного опыта.*
- *Также это способность обнаруживать свойства окружающих объектов и явлений,*
  - *а также взаимосвязей между ними,*
  - *в сочетании с возможностью решать новые проблемы*

Искусственный интеллект (ИИ)

Относится к

- *Способность машин имитировать человеческие функции, такие как мышление, обучение, планирование и творчество.*
- *С помощью ИИ технические системы могут воспринимать окружающую среду,*
- *обрабатывать эту информацию и решать проблемы для достижения определенной цели.*
- *Аналогично области ИИ используется концепция вычислительного интеллекта (ВИ), определяемая как:*
  - *Моделирование биологического интеллекта, в отличие от ИИ, который основан на понятии знаний.*

Искусственный интеллект

- *Применение ИИ для больших данных.*
- *Техники ИИ для анализа и интерпретации больших данных.*
- *Использование нейронных сетей и глубокого обучения в больших данных.*
- *Среди используемых интеллектуальных технологий можно выделить*
  - *Data Mining (DM), Text Mining (TM),*
  - *Natural Language Processing (NLP),*
  - *машинное обучение (ML), глубокое обучение (DL),*
  - *Transfer Learning (TL), Reinforce Learning (RL).*

Data Mining

- *Нетривиальный процесс извлечения полезной, ранее неизвестной, интересной и потенциально полезной информации из данных,*
- *как правило, в виде моделей и шаблонов знаний, с целью обнаружения новых взаимосвязей и обобщения данных*
- *таким образом, чтобы они были понятны и полезны владельцу данных.*
- *Методы и алгоритмы извлечения знаний из массивных наборов данных.*
- *Внедрение и оптимизация техник Data Mining на массивах данных.*
- *Платформы и инструменты для Data Mining в Big Data*

Алгоритмы DM

- ***Алгоритмы ассоциации**: выявляют связи между элементами набора данных. Например, «клиенты, которые покупают хлеб, покупают и молоко». (Пример: алгоритм Apriori);*
- ***Алгоритмы классификации**: распределяют элементы по одному или нескольким заранее определенным классам.*
- *Например, классификация электронных писем как спам или нет.*
  - *(Алгоритмы: SVM, Naïve Bayes, деревья решений . . .)*
- ***Алгоритмы регрессии**: предсказывают непрерывное числовое значение. Например, предсказание цены дома в зависимости от площади, количества комнат и т. д.*
- *(Пример: линейная регрессия, логистическая регрессия)*
- ***Алгоритмы кластеризации**: Группируют похожие данные в кластеры. Например, сегментация клиентов в зависимости от поведения при покупке.*

Машинное обучение (ML)

Определение:

- *ML – это подраздел ИИ, который фокусируется на разработке алгоритмов, позволяющих компьютерам учиться на данных и делать прогнозы, принимать или предлагать решения без явного программирования для каждой задачи.*
- *Масштабируемые алгоритмы машинного обучения для больших данных.*
- *Использование ML в больших данных для прогнозирования и предсказания.*

ML

Процесс:

- Обучение модели на основе исторических данных (под наблюдением, без наблюдения или с частичным наблюдением).
- Валидация и тестирование модели для оценки ее эффективности.
- Применение модели для прогнозирования или классификации новых наборов данных.

Цель:

- Машинное обучение сосредоточено на создании моделей, которые могут обобщать и применять знания, полученные из данных, к новым задачам.

KDD против ML

Подход:

KDD *– это сложный и комплексный процесс обнаружения знаний в данных, который включает в себя несколько этапов;*

ML *– одна из техник, используемых на этапе DM в KDD.*

Область применения:

KDD *– это более широкий термин, охватывающий весь процесс анализа данных;*

ML *– это конкретная область, занимающаяся разработкой и внедрением алгоритмов машинного обучения.*

Фокус:

KDD *фокусируется на извлечении полезных знаний и их интерпретации,*

ML *сосредоточено на оптимизации и создании моделей, которые обучаются на данных для прогнозирования.*

Обучение с помощью вознаграждения (усиление)
(RL - Reinforcement Learning)

- *Раздел ML, который фокусируется на том, как программные агенты должны действовать в среде, чтобы максимизировать понятие кумулятивного вознаграждения.*
- *Центральная концепция заключается в обучении политике принятия решений, которая указывает агенту, какие действия следует предпринимать в различных ситуациях, чтобы получить наибольшее вознаграждение в долгосрочной перспективе.*

Ключевые концепции:

- ***Агент**: Субъект, который учится и принимает решения (например, робот или программное обеспечение).*
- ***Среда**: все, что находится вне агента и с чем агент взаимодействует.*

RL (продолжение)

- ***Состояния (States)***: *Представления возможных ситуаций, в которых может оказаться агент в среде.*
- ***Действия (Actions)***: *Решения или движения, которые агент может совершать.*
- ***Вознаграждение (Reward)***: *обратная связь, получаемая агентом после совершения определенного действия в определенном состоянии. Вознаграждение может быть положительным или отрицательным и используется для оценки эффективности действия.*
- ***Политика (Policy)***: *Стратегия, которую агент использует для принятия решения о том, какое действие предпринять в определенном состоянии. Политика может быть детерминированной или стохастической.*
- ***Ценность (Value)***: *Мера того, насколько хорош состояние или действие с точки зрения ожидаемых будущих вознаграждений.*

Обучение путем переноса (*TL – Transfer learning*)

Глубокое обучение

- *Подраздел ML, использующий многослойные искусственные нейронные сети.*
- *Искусственные нейронные сети: модели, вдохновленные человеческим мозгом.*
- *Сверточные нейронные сети (CNN): в основном используются для обработки изображений.*
- *Рекуррентные нейронные сети (RNN): используются для обработки последовательностей, таких как текст или временные ряды.*
- *Генеративно-состязательные сети (GAN): используются для генерации новых данных, таких как изображения или тексты.*

LLM — большая языковая модель

- *LLM – это модель искусственного интеллекта, обученная на больших объемах текстовых данных для понимания, генерации и интерпретации естественного языка.*
- *Эти модели основаны на передовых архитектурах нейронных сетей, таких как Transformers, которые используются в таких приложениях, как автоматический перевод, автозаполнение текста, чат-боты и резюмирование текста.*

Характеристики LLM

1. *Масштабируемость: обучены на основе больших данных, включая книги, статьи, веб-сайты и т. д.*
2. *Многофункциональность: могут отвечать на вопросы, резюмировать тексты, писать код, выполнять переводы и генерировать творческий контент.*
3. *Архитектура Transformer: используют механизмы самовнимания для понимания контекста каждого слова в тексте. Примеры включают GPT (Generative Pre-trained Transformer), BERT (Bidirectional Encoder Representations from Transformers) и PaLM.*
4. *Предварительное обучение и адаптация: Модели сначала обучаются на общих наборах данных (предварительное обучение), а затем дорабатываются для конкретных задач (точная настройка).*

Приложения

- *Виртуальные помощники (например, ChatGPT, Google Bard)*
- *Текстовые системы рекомендаций*
- *Анализ настроения*
- *Автоматическое заполнение кода (например, GitHub Copilot)*

LLM – Логическая языковая модель

- *Логическая языковая модель – это специализированный класс моделей, которые интегрируют формальную логику (такую как пропозициональная логика или логика первого порядка) в обработку и генерацию естественного языка.*
- *Эти модели используются для объединения логического мышления и естественного языка, предоставляя возможности дедукции, проверки утверждений или генерации текстов на основе логических правил.*

Характеристики логических языковых моделей

1. *Интеграция формальной логики: модели сочетают естественный язык с четко определенными логическими правилами для ответа на вопросы или проверки утверждений.*
2. *Применение в символическом ИИ: используют методы символического ИИ для точных выводов и рассуждений.*
3. *Способность проверки: позволяет моделям проверять достоверность предложений или решать проблемы на основе логических правил.*

Примеры использования

- *Экспертные системы: генерация рекомендаций на основе заранее определенных правил.*
- *Проверка данных: логическая проверка согласованности данных в базах знаний.*
- *Дедуктивное мышление: решение математических или научных задач, связанных со сложными логическими отношениями.*

Различия

Внешний вид	*Режим большого языка*	*Логическая языковая модель*
Цель	*Понимание и генерация естественного языка*	*Рассуждения на основе формальной логики*
Данные для обучения	*Неструктурированные данные (текст, разговоры)*	*Логические правила и формальный язык*
Приложения	*Чат-боты, переводы, резюме*	*Экспертные системы, логическое мышление*
Гибкость	*Общая и контекстуальная*	*Строго основанный на правилах*

Обработка естественного языка

- *NLP – подраздел ИИ, междисциплинарная область на стыке ИИ, лингвистики и информатики, занимающаяся обработкой и пониманием естественного языка компьютерами.*
- *Цель NLP – разработать приложения, которые позволяют машинам интерпретировать, анализировать, генерировать и взаимодействовать с человеческим языком максимально естественным образом.*
- *Естественный язык и логика: интегрирует логическое мышление для проверки и понимания текста.*
- *Анализ больших наборов текстов – Big Data.*

Цели НЛП

1. *Понимание языка: интерпретация значения слов, предложений или общего контекста.*
2. *Генерация языка: создание связного текста, например, автозаполнение или генерация ответов.*
3. *Анализ языка: извлечение информации, анализ настроения или классификация текста.*
4. *Взаимодействие человека и компьютера: разработка естественных интерфейсов, таких как виртуальные помощники.*

Поддомены NLP

1. Синтаксическая обработка:
   1. Анализ грамматической структуры (парсинг).
   2. Идентификация частей речи (POS-тегирование).
2. Семантическая обработка:
   1. Понимание значения слов и фраз (лексическая семантика).
   2. Идентификация отношений между концептами (онтологии и семантические сети).
3. Прагматическая обработка:
   1. Понимание контекста и намерения говорящего.
4. Распознавание и генерация речи:
   1. Преобразование речи в текст (speech-to-text).
   2. Генерация речи из текста (text-to-speech).
5. Анализ настроения и мнений:
   1. Идентификация полярности (положительная, отрицательная, нейтральная) в отзывах, комментариях и т. д.

Общие приложения NLP

1. *Поисковые системы: понимание намерений пользователя и улучшение результатов поиска.*
2. *Виртуальные помощники: Alexa, Google Assistant, Siri, которые интерпретируют голосовые команды.*
3. *Чат-боты: автоматические разговоры с пользователями, используемые в службах поддержки клиентов.*
4. *Анализ текста: классификация, резюмирование и извлечение информации из документов.*
5. *Автоматический перевод: Google Translate и другие подобные системы.*
6. *Грамматическая коррекция: инструменты, такие как Grammarly.*
7. *Обнаружение спама: идентификация нежелательных сообщений в электронной почте.*

Методы и технологии в НЛП

1. Языковые модели:
   1. Лингвистические правила: системы, основанные на грамматике и правилах (более старые).
   2. Машинное обучение (ML): алгоритмы, обучающиеся на данных, такие как Naive Bayes или SVM.
   3. Глубокое обучение: передовые модели, такие как рекуррентные нейронные сети (RNN), LSTM и Transformers (например, GPT, BERT).
2. Предварительная обработка текста:
   1. Токенизация: Разделение текста на слова или фразы.
   2. Стеминг и лематизация: сокращение слов до их базовой формы.
   3. Удаление неважных слов: фильтрация стоп-слов (например, «и», «в», «от»).
3. Техники представления текста:
   1. Bag of Words (BoW) и TF-IDF: простые числовые представления.
   2. Word Embeddings: распределенные представления, такие как Word2Vec, GloVe или вложения, сгенерированные моделями Transformers.

Проблемы в НЛП

1. *Неоднозначность языка: многие слова или фразы могут иметь несколько значений (например, полисемия).*
2. *Культурный и лингвистический контекст: значения могут варьироваться в зависимости от региона, культуры или диалекта.*
3. *Стилистические фигуры: понимание метафор, иронии или сарказма.*
4. *Обработка сложных языков: языки со сложной грамматической структурой или уникальными характеристиками (например, тональные языки).*

Текстовый майнинг

- *Обработка и анализ больших объемов текстовых данных.*
- *Масштабируемые алгоритмы текстового майнинга.*
- *Применение текстового майнинга в анализе больших данных (например, анализ настроений, классификация текстов).*

Определение ТМ

- *Отрасль искусственного интеллекта и компьютерной лингвистики;*
- *Процесс автоматического анализа больших объемов текстовых данных и извлечения полезной информации, моделей и приобретения скрытых релевантных знаний.*
- *Основная цель – преобразовать неорганизованные и неструктурированные текстовые данные в структурированную форму, которую легко анализировать и интерпретировать.*
- *ТМ широко используется в различных областях: маркетинге, здравоохранении, безопасности и национальной разведке, науке... для преобразования больших объемов текстовых данных в информацию, которую можно использовать.*

Категории алгоритмов ТМ

- ***Обучение с контролем (классификация)**: Модель обучается на помеченных данных.*
  - *Примеры: классификация, регрессия.*
- ***Неконтролируемое обучение (кластеризация)**: Модель выявляет закономерности в немеченых данных.*
  - *Примеры: кластеризация, ассоциация.*
- ***Обучение с подкреплением**: агент учится принимать решения в среде, чтобы максимизировать вознаграждение.*

Основные этапы текстового майнинга

- *Предварительная обработка;*
- *Токенизация;*
- *Лемматизация и стемминг;*
- *Преобразование данных;*
- *Извлечение характеристик;*
- *Анализ и интерпретация.*

Большие данные и **Iot** и **IIoT**

Большие данные и IoT

- *Практическое применение обработки в реальном времени в контексте IoT.*
- *Большие данные и IoT (Интернет вещей) имеют широкое применение в анализе и обработке данных в режиме реального времени.*
- *IoT можно считать «движущей силой» для Big Data, то есть важным источником данных, который требует инфраструктуры Big Data для эффективного управления и анализа.*
- *Данные из IoT обрабатываются и анализируются в режиме реального времени, укрепляя таким образом логические связи между Strimtg и IoT.*

Искусственный интеллект вещей (AIoT)

- *AIoT – это сочетание ИИ и IoT.*
- *AIoT объединяет возможности подключения и сбора данных устройств IoT с мощью анализа, обучения и автономного принятия решений ИИ.*
- *Эта интеграция позволяет устройствам IoT не только собирать и отправлять данные, но и обрабатывать и анализировать эти данные в режиме реального времени для принятия интеллектуальных и автоматических решений.*

Характеристики и преимущества AIoT

1) ***Автономность и автоматизация***: *AIoT позволяет системам работать автономно без вмешательства человека, повышая эффективность и снижая операционные затраты.*
2) ***Анализ в реальном времени***: *благодаря интеграции ИИ в устройства IoT анализ данных может выполняться мгновенно, что крайне важно для приложений, требующих быстрой реакции, таких как автономные автомобили или системы безопасности.*
3) ***Автоматическое обучение***: *AIoT использует алгоритмы машинного обучения для обучения на основе собранных данных, постоянно улучшая свою производительность и адаптируясь к меняющимся условиям.*
4) ***Проактивные решения***: *устройства AIoT могут предвидеть проблемы и принимать превентивные меры, такие как прогнозное техническое обслуживание на заводах или оптимизация энергопотребления в интеллектуальных зданиях.*

Приложения AIoT

- ***Умные города***: *оптимизация трафика, управление энергопотреблением и мониторинг загрязнения.*
- ***Промышленные системы***: *автоматизация производственных процессов, прогнозное техническое обслуживание оборудования.*
- ***Умные дома***: *виртуальные помощники, передовые системы безопасности, автоматический контроль освещения и температуры.*
- ***Здравоохранение***: *носимые устройства для мониторинга здоровья, автоматическая диагностика и оптимизация лечения.*
- ***Автономные автомобили***: *использование данных с датчиков для навигации и принятия решений в режиме*

Экономика 4.0

Точная экономика

Экономика 5.0

Регенеративная экономика

(Е. Интеллектуальная)

Экономика 4.0 и 5.0 и роль *Big Data*

- *Экономика 4.0 и 5.0 представляют собой эволюционные этапы промышленной революции, характеризующиеся глубокой интеграцией цифровых технологий во все аспекты жизни и экономики.*
- *Большие данные играют центральную роль в этих изменениях, обеспечивая необходимый импульс для инноваций и преобразований.*

Основные компоненты Экономики 4.0 и 5.0

Значительная эволюция способа производства и потребления товаров и услуг, подпитываемая технологиями:

ИИ, IoT, большие данные, блокчейн

Общество		
4.0	Промышленность	5.0
4.0	Сельское хозяйство	5
4.0	Биотехнологии	5
4	Услуги	5
4	Управление	5
5G		

Экономика 4.0: Цифровые основы

- *Характеристики:*
  - *Цифровизация, автоматизация, взаимосвязь, Интернет вещей (IoT).*
  - *Роль больших данных: оптимизация процессов, прогнозирование, персонализация услуг, открытие новых бизнес-моделей.*
  - *Примеры: интеллектуальная фабрика, автономные автомобили, интеллектуальные города.*

Промышленность 4.0

Автоматизация и эффективность

- ***Интеллектуальное производство***: использование коллаборативных роботов, 3D-принтеров и автономных систем для оптимизации производства.
- ***Профилактическое техническое обслуживание***: использование датчиков и анализа данных для прогнозирования отказов оборудования.
- ***Интеллектуальные цепочки поставок***: интеграция цифровых технологий для более эффективного управления потоком материалов и информации.

Сельское хозяйство 4.0

- ***Точное земледелие:***
  - ***Использование дронов, датчиков и анализа данных для оптимизации использования сельскохозяйственных ресурсов.***
  - ***Умные фермы: автоматизация сельскохозяйственных процессов, от орошения до сбора урожая.***
  - ***Устойчивые пищевые цепочки: отслеживаемость пищевых продуктов и сокращение отходов.***

Услуги 4.0

- **Здравоохранение**: телемедицина, диагностика с помощью искусственного интеллекта, индивидуализация лечения.
- **Образование**: персонализированное обучение, виртуальная и дополненная реальность, интерактивные онлайн-платформы.
- **Финансы**: цифровой банкинг, криптовалюты, робо-консультанты.
- **Транспорт**: автономные транспортные средства, интеллектуальная логистика, городская мобильность.
- **Энергетика**: интеллектуальные сети, возобновляемые источники энергии, энергоэффективность.

Управление 4.0

- Электронное правительство:
  - Упрощение взаимодействия между гражданами и администрацией.
  - Умные города: использование цифровых технологий для улучшения качества жизни в городах.
  - Кибербезопасность: защита критически важных инфраструктур и персональных данных.

Экономика 5.0

- *Новая концепция, объединяющая такие передовые технологии, как:*
  - *Искусственный интеллект (ИИ),*
  - *Интернет вещей (IoT),*
  - *анализ больших данных*
- *для создания более интеллектуальной, устойчивой и ориентированной на человека экономики.*

Экономика 5.0: интеллектуальный подход

Характеристики

- ***Устойчивость**: интеграция принципов устойчивого развития во все секторы экономики.*
- ***Этика**: Разработка и использование технологий ответственным и этичным образом.*
- ***Инклюзивность**: массовая персонализация. Обеспечение доступа к преимуществам технологий для всех.*
- ***Сотрудничество человека и машины**: создание синергии между людьми и машинами для решения сложных задач.*

Роль больших данных:

Создание персонализированных продуктов и услуг, оптимизация цепочек поставок, разработка устойчивых решений.

- *Примеры: персонализированная медицина, производство по запросу, циркулярная экономика.*

Промышленность 5.0

- *Сосредоточена на сотрудничестве между людьми и интеллектуальными машинами, привнося индивидуализацию и креативность в производство.*
- *Более тесно интегрирует человеческий фактор в производственные процессы, ценя креативность и человеческие навыки.*
- *Массовая персонализация: предлагает персонализированные продукты и услуги в широком масштабе, используя такие технологии, как 3D-печать.*
- *Использование передовых технологий для усиления человеческих способностей и повышения устойчивости.*

Промышленность 5.0 (продолжение)

- *Устойчивость: использует переработанные материалы, возобновляемые источники энергии и сокращает углеродный след.*
- *Интеллектуальные энергетические сети: интегрирует возобновляемые источники энергии, хранение и потребителей в эффективную и гибкую систему.*
- *Энергоэффективность: сокращает потребление энергии во всех секторах за счет использования интеллектуальных технологий и поведения потребителей.*
- *Доступная энергия: Обеспечивает доступ к энергии для всех сообществ, включая маргинализированные.*

Сельское хозяйство 5.0

- *Интегрирует передовые технологии: роботы, дроны, прогнозные анализы, чтобы создать более точное, эффективное и экологичное сельское хозяйство.*
- *Цель состоит в оптимизации сельскохозяйственного производства, снижении воздействия на окружающую среду и повышении продовольственной безопасности.*
- *Регенеративное сельское хозяйство: улучшение здоровья почвы, биоразнообразия и устойчивости экосистем.*
- *Функциональные продукты питания: производство продуктов питания с улучшенными питательными свойствами, адаптированных к индивидуальным потребностям.*
- *Городское сельское хозяйство: выращивание продуктов питания в городах с использованием вертикальных пространств и*

Биотехнологии

- *Bt 4.0, биотехнологии сосредоточены на использовании передовых технологий, таких как биоинформатика, синтетическая биология и генная инженерия, для оптимизации биологических процессов в различных отраслях, таких как сельское хозяйство, медицина и энергетика.*
- *Bt 5.0, акцент делается на более глубокой интеграции между биотехнологиями и ИИ, большими данными и IoT с целью создания более устойчивых, персонализированных и этичных решений.*
- *Они могут включать разработку новых персонализированных медицинских методов лечения на основе углубленного анализа генетических данных и использование ИИ для оптимизации биологических процессов*

Услуги 5.0

- *Цифровизация и крайняя персонализация услуг с помощью ИИ и больших данных.*
- *Медицинские услуги становятся более доступными и адаптированными к индивидуальным потребностям, предлагая пользователям более высокий уровень обслуживания благодаря:*
  - *Прогностическое здравоохранение: использование данных для прогнозирования заболеваний и предоставления персонализированных методов лечения.*
  - *Психическое здоровье: цифровые решения для психического здоровья, такие как онлайн-терапия и виртуальная реальность.*
  - *Глобальное здравоохранение: усилия, направленные на искоренение инфекционных заболеваний и улучшение доступа к медицинской помощи в развивающихся странах.*

Услуги 5.0 (продолжение)

- *Образование 5.0 – Обучение на протяжении всей жизни: предоставляет возможности для непрерывного обучения, адаптированного к индивидуальным потребностям и потребностям рынка труда.*
- *Персонализированное образование: использует технологии для создания учебных программ, адаптированных к стилю обучения каждого ученика.*
- *Образование для глобального гражданства: готовит молодежь к решению глобальных проблем, таких как изменение климата и неравенство.*

Управление 5.0

- *Использует цифровые технологии для создания более прозрачного, эффективного и ориентированного на граждан правительства.*
- *Внедрение ИИ, блокчейна и IoT в управление позволяет лучше собирать и анализировать данные;*
- *Облегчает принятие решений на основе данных и улучшает качество государственных услуг.*

Связь *больших данных* с этими экономиками

- ***Генерация данных***: *IoT, датчики, мобильные устройства генерируют огромные объемы данных.*
- ***Хранение данных***: *облачные инфраструктуры и распределенные базы данных управляют этими большими объемами данных.*
- ***Анализ данных***: *алгоритмы машинного обучения и искусственного интеллекта извлекают ценную информацию из данных.*
- ***Применение данных***: *данные используются для улучшения процессов, принятия более эффективных решений и создания новых продуктов и услуг.*

Примеры использования больших данных в экономике 4.0 и 5.0

- ***Здравоохранение**: раннее диагностирование заболеваний, разработка индивидуальных методов лечения, оптимизация управления запасами лекарственных средств.*
- ***Промышленность**: оптимизация производственных процессов, прогнозное техническое обслуживание, разработка индивидуальных продуктов.*
- ***Розничная торговля**: индивидуальные рекомендации по продуктам, оптимизация цен, эффективное управление запасами.*
- ***Финансы**: выявление мошенничества, оценка кредитных рисков, разработка индивидуальных финансовых продуктов*