Оглавление

[ОБЩИЙ ВЗГЛЯД НА DATA SCIENCE В КОНТЕКСТЕ БОЛЬШИХ ДАННЫХ. 1](#_Toc178067020)

[1. ОБЛАСТЬ ПРИМЕНЕНИЯ DATA SCIENCE И БОЛЬШИХ ДАННЫХ И ИХ ПРЕИМУЩЕСТВА 1](#_Toc178067021)

[2. КАТЕГОРИИ ДАННЫХ 2](#_Toc178067022)

[3. ПРОЦЕСС DATA SCIENCE 3](#_Toc178067023)

[4. ЭКОСИСТЕМА БОЛЬШИХ ДАННЫХ 4](#_Toc178067024)

[5. ПРИМЕР ИСПОЛЬЗОВАНИЯ HADOOP 7](#_Toc178067025)

[ИТОГ 8](#_Toc178067026)

## ОБЩИЙ ВЗГЛЯД НА DATA SCIENCE В КОНТЕКСТЕ БОЛЬШИХ ДАННЫХ.

В концепцию data science входит использование методов анализа огромных объемов данных и извлечения содержащейся в них информации. Связь между большими данными и data science такая же, как между сырой нефтью и нефтеперерабатывающим заводом. Data science и большие данные развивались на базе статистики и традиционного управления данными, но сейчас считаются разными дисциплинами.

Характеристики больших данных часто называются «тремя V»:

*Объем* (Volume) — сколько данных содержит набор?

*Разнообразие* (Variety) — насколько отличаются друг от друга разные типы данных?

*Скорость* (Velocity) — с какой скоростью генерируются новые данные?

Часто эти характеристики дополняются «четвертым V» — *достоверностью* (Veracity): насколько точны данные? Эти четыре свойства отличают большие данные от данных, встречающихся в традиционных средствах управления данными. Соответственно, привносимые ими изменения проявляются почти во всех аспектах: сборе данных, хранении и обслуживании данных, поиске, обмене, передаче и визуализации. Кроме того, большие данные требуют применения специализированных средств извлечения информации.

Data science — это расширение статистики, способное справляться с огромными объемами данных, производимыми в наши дни. Data science добавляет методы из computer science (информатики) в репертуар статистики.

По мере роста объема данных и потребностей в их использовании каждый специалист data science будет сталкиваться с проектами из области больших данных на протяжении своей карьеры.

# 1. ОБЛАСТЬ ПРИМЕНЕНИЯ DATA SCIENCE И БОЛЬШИХ ДАННЫХ И ИХ ПРЕИМУЩЕСТВА

Data science и большие данные сейчас встречаются почти повсеместно как в коммерческих, так и в некоммерческих средах.

Коммерческие компании почти во всех промышленных областях используют data science и большие данные для получения информации о клиентах, процессах, персонале, конкурентах и товарах.

Сбор информации об интернет-пользователях для показа контекстной рекламы. Персонализация рекламы в реальном времени. Профессионалы в области HR используют личностную аналитику (people analytics) и глубокий анализ текста для отбора кандидатов, отслеживания настроения работников и изучения неформальных связей среди коллег.

People analytics является центральной темой книги «Moneyball: The Art of Winning an Unfair Game» (В российском прокате фильм шел под названием «Человек, который изменил всё»). В книге (и фильме) показано, что традиционный процесс отбора спортсменов в бейсболе был случайным, и его замена коррелированными признаками изменила всё. Применение статистики позволило нанимать правильных игроков и выставлять их против тех соперников, над которыми возможно наибольшее преимущество. Финансовые учреждения используют data science для прогнозирования рынка ценных бумаг, вычисления риска предоставления ссуд и привлечения новых клиентов. Правительственные и неправительственные организации хорошо осведомлены о ценности данных.

# 2. КАТЕГОРИИ ДАННЫХ

В data science и области больших данных встречается много разных типов данных, для каждого из которых требуются свои инструменты и методы.

* Структурированные.
* Неструктурированные.
* На естественном языке.
* Машинные.
* Графовые.
* Аудио, видео и графика.
* Потоковые.

*СТРУКТУРИРОВАННЫЕ ДАННЫЕ*

Структурированные данные зависят от модели данных и хранятся в фиксированном поле внутри записи. Соответственно, структурированные данные часто бывает удобно хранить в таблицах, в базах данных или файлах Excel. SQL (Structured Query Language, язык структурированных запросов) является основным средством управления и обращения с запросами к данным, хранящимся в базах данных, т.е. данные прямого доступа. Также иногда встречаются структурированные данные, которые достаточно трудно сохранить в традиционной реляционной базе данных (один из примеров — иерархические данные, например, генеалогическое дерево), т.е. данные последовательного доступа.

*НЕСТРУКТУРИРОВАННЫЕ ДАННЫЕ*

Неструктурированные данные трудно подогнать под конкретную модель данных, потому что их содержимое зависит от контекста или имеет переменный характер, например, обычные сообщения электронной почты

*ДАННЫЕ НА ЕСТЕСТВЕННОМ ЯЗЫКЕ*

Данные на естественном языке составляют особую разновидность неструктурированных данных; обработка таких данных достаточно сложна, потому что она требует знания как лингвистики, так и специальных методов data science, связанных с распознаванием сущностей, тематических областей, обобщением, завершением текста и анализом эмоциональной окраски. Модели, адаптированные для одной предметной области, плохо обобщаются для других областей. Даже самые современные методы не смогут расшифровать смысл произвольного фрагмента текста. У людей возникают проблемы с восприятием естественного языка. Он неоднозначен по своей природе. Сама концепция смысла выглядит спорно и плохо определяема. Если два человека слушают один разговор; вынесут ли они одинаковый смысл из него? Даже смысл отдельных слов может изменяться в зависимости от настроения говорящего.

*МАШИННЫЕ ДАННЫЕ*

К машинным данным относится информация, автоматически генерируемая компьютером, процессом, приложением или устройством без вмешательства человека, например, *Интернет вещей*.

*ГРАФОВЫЕ, ИЛИ СЕТЕВЫЕ, ДАННЫЕ*

Под «графом» имеется в виду понятие из математической теории графов — математическая структура для моделирования попарных отношений между объектами. Графовые структуры данных используют узлы, ребра и свойства для представления и хранения данных. Графовые данных гармонично описывают социальные сети. Для хранения графовых данных используются графовые базы данных, а для построения запросов к ним — такие специализированные языки запросов, как SPARQL (https://ru.wikipedia.org/wiki/SPARQL).

*АУДИО, ВИДЕО И ГРАФИКА*

Аудио, видео и графика — типы данных, ставящие задачи, тривиальные с точки зрения человека (например, распознавание объекта на картинке), но сложные для компьютера.

Левая часть мозга человека выстраивает простейшие схемы взаимосвязей, но простейшие не в смысле их общего представления, а в смысле возможности количественного представления. Образное мышление более информационно насыщенно, потому, что не имеет полного цифрового образа. Хороший пример односторонние и двусторонние рынки. Одна сторона из двустороннего взаимодействия не имеет возможности отказаться от взаимодействия с другой при наличии субъективного представления о взаимодействии.

*ПОТОКОВЫЕ ДАННЫЕ*

Потоковые данные могут принимать почти любую из перечисленных форм, однако у них имеется одно дополнительное свойство. Данные поступают в систему при возникновении некоторых событий, а не загружаются в хранилище данных большими массивами.

# 3. ПРОЦЕСС DATA SCIENCE

* *НАЗНАЧЕНИЕ ЦЕЛИ ИССЛЕДОВАНИЯ*
* *СБОР ДАННЫХ*

Источники.

* *ПОДГОТОВКА ДАННЫХ*

Очистка, интеграция, преобразование.

* *ИССЛЕДОВАНИЕ ДАННЫХ*

Понимание данных их распределение и исследование аномалий.

* *МОДЕЛИРОВАНИЕ ДАННЫХ ИЛИ ПОСТРОЕНИЕ МОДЕЛИ*

Построение модели является итеративным процессом, в ходе которого исследователь выбирает переменные для модели, применяет модель и проводит диагностику модели.

* *ОТОБРАЖЕНИЕ И АВТОМАТИЗАЦИЯ*

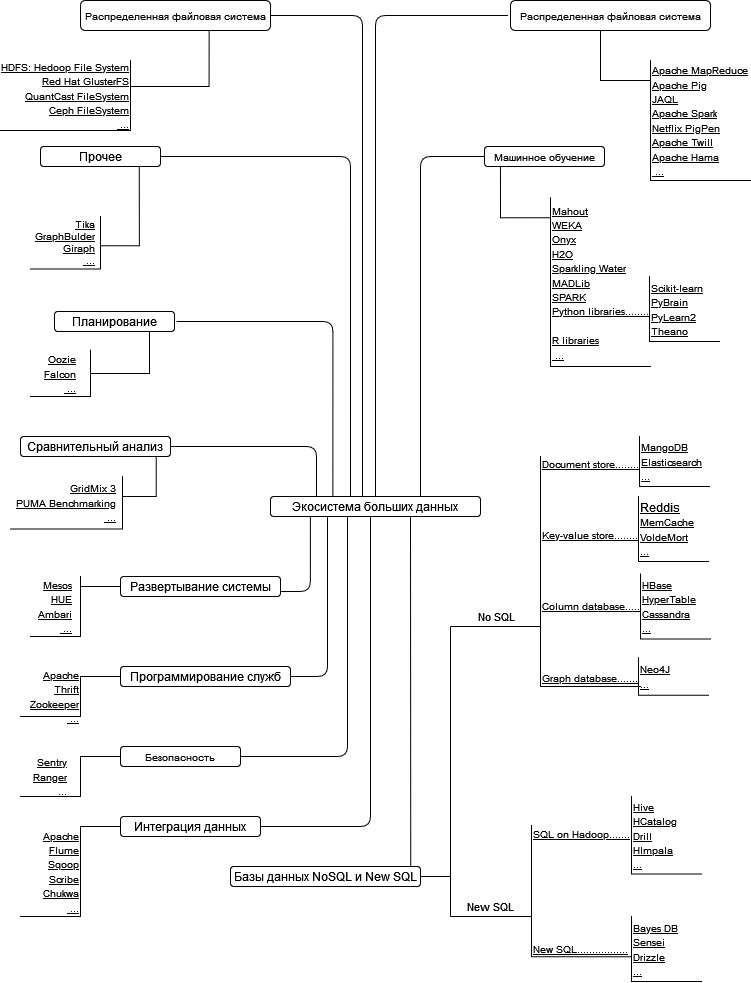
Представление результатов заказчику (бизнесу, или научному сообществу).

***Процесс data science является итеративным.***

**Мы решили найти, что-то общее в большом объеме данных без предположений о моделях взаимосвязей и здесь у человека возникает немалая проблема в обработке большого количества данных, мало того обработки в реальном времени, что недоступно сенсорам и механизмам действия интеллектуального агента в виде реального человека. И здесь мы обращаемся к компьютерам для обработки таких потоков данных.**

**В подходе, основанном на технологиях больших данных много допускающего критики, в частности мы можем нехотя сформировать само сбывающиеся прогнозы основываясь на оценке корреляции и подстройке данных под полученные корреляции. (Это отсюда можно выбросить)**

# 4. ЭКОСИСТЕМА БОЛЬШИХ ДАННЫХ



*РАСПРЕДЕЛЕННЫЕ ФАЙЛОВЫЕ СИСТЕМЫ*

Распределенная файловая система похожа на обычную файловую систему, но, в отличие от последней, она работает на нескольких серверах сразу.

Распределенные файловые системы обладают рядом важных преимуществ:

* Они способны хранить файлы, размер которых превышает размер диска отдельного компьютера.
* Файлы автоматически реплицируются на нескольких серверах для создания избыточности или выполнения параллельных операций, при этом все сложности технической реализации скрываются от пользователя.
* Система легко масштабируется: пользователь не ограничивается объемом памяти или дискового пространства одного сервера.

*ИНФРАСТРУКТУРА РАСПРЕДЕЛЕННОГО ПРОГРАММИРОВАНИЯ*

Один из важных аспектов работы с распределенным жестким диском заключается в том, что вы не перемещаете данные к программе, а скорее перемешаете программу к данным. Вам придется иметь дело со всеми сложностями, присущими распределенному программированию, например, перезапуском сбойных заданий, отслеживанием результатов из других субпроцессов и т. д.

*ИНФРАСТРУКТУРА ИНТЕГРАЦИИ ДАННЫХ*

Apache Sqoop и Apache Flume.

*ИНФРАСТРУКТУРЫ МАШИННОГО ОБУЧЕНИЯ*

Самая популярная библиотека машинного обучения для Python называется Scikit-learn.

Это отличный инструмент машинного обучения, и мы будем использовать ее. Конечно, существуют и другие библиотеки для Python:

* *PyBrain* для нейронных сетей - термином «нейронные сети» обозначаются алгоритмы обучения, моделирующие человеческий мозг в отношении механики обучения и сложности. Нейронные сети принято считать областью сложной и недоступной непосвященных.
* *NLTK* (Natural Language Toolkit) - как подсказывает название, библиотека предназначена для работы с данными на естественном языке. К этой обширной библиотеке прилагается ряд текстовых корпусов, упрощающих моделирование ваших данных.
* *Pylearn2* - другой инструментарий машинного обучения, немного уступающций Scikit-learn.
* *TensorFlow* - библиотека Python для глубокого обучения, предоставленная компанией Google.

Конечно, варианты не ограничиваются библиотеками Python. Spark - новое ядро машинного обучения с лицензией Apache, специализирующееся на машинном обучении в реальном времени. Проект заслуживает внимания; дополнительную информацию можно найти по адресу <http://spark.apache.org/>.

*БАЗЫ ДАННЫХ NoSQL*

«No» в этом контексте означает «не только».

Существует много разновидностей баз данных, но их можно разделить на следующие типы:

*Столбцовые базы данных* — данные организуются в столбцы, что позволяет алгоритмам существенно повышать скорость обработки запросов. В более современных технологиях используется принцип хранения по ячейкам. Табличные структуры продолжают играть важную роль в обработке данных.

*Хранилища документов* — хранилища документов не используют таблицы, но хранят полную информацию о документе. Их отличительной особенностью является чрезвычайно гибкая схема данных.

*Потоковые данные* — сбор, преобразование и агрегирование данных осуществляются не по пакетному принципу, а в реальном времени. Хотя мы выделили потоковые данные в категорию баз данных, чтобы упростить выбор инструмента, скорее они образуют особую разновидность задач, породившую такие технологии, как Storm.

*Хранилища «ключ—значение»* — данные не хранятся в таблицах; с каждым значением связывается ключ: например, org.marketing.sales.2015:20000. Такое решение хорошо масштабируется, но реализация почти полностью возлагается на разработчика.

*SQL в Hadoop* — пакетные запросы в Hadoop пишутся на SQL-подобном языке, во внутренней реализации которого используется инфраструктура отображения-свертки (map-reduce).

*Обновленный SQL* — этот тип сочетает масштабируемость баз данных NoSQL с преимуществами реляционных баз данных. Все эти базы данных используют интерфейс SQL и реляционную модель данных.

*Графовые базы данных* — не для всех задач табличный формат является оптимальным. Некоторые задачи более естественно подходят для представления в виде графа и хранения в графовых базах данных. Классический пример такого рода — социальная сеть.

*ИНСТРУМЕНТЫ ПЛАНИРОВАНИЯ*

Инструменты планирования упрощают автоматизацию повторяющихся операций и запуск заданий по событиям (например, при появлении нового файла в папке).

Они похожи на такие традиционные программы, как CRON в Linux, но разработаны специально для больших данных. Например, такие инструменты могут запускать задачу MapRеducе при появлении нового набора данных в каталоге.

*ИНСТРУМЕНТЫ СРАВНИТЕЛЬНОГО АНАЛИЗА*

Этот класс инструментов был разработан для оптимизации установки больших данных за счет предоставления стандартизированных профилей. Профили строятся на основании представительного множества операций с большими данными.

Задачи сравнительного анализа и оптимизации инфраструктуры больших данных и настройки конфигурации часто относятся к компетенции не специалистов data science, а профессионалов, специализирующихся на организации IТ-инфраструктуры.

Использование оптимизированной инфраструктуры приводит к существенной экономии. Например, экономия 10% в кластере из 100 серверов равна стоимости 10 серверов.

*РАЗВЕРТЫВАНИЕ СИСТЕМЫ*

Подготовка инфраструктуры больших данных - непростая задача. Инструменты развертывания системы проявляют себя при развертывании новых приложений в кластерах больших данных. Они в значительной степени автоматизируют установку и настройку компонентов больших данных. К числу основных задач специалиста data science эта область не относится.

*ПРОГРАММИРОВАНИЕ СЛУЖБ*

Предположим, вы создали приложение для прогнозирования результатов футбольных матчей и теперь хотите предоставить всем желающим доступ к прогнозам вашего приложения. Однако вы понятия не имеете, какие архитектуры или технологии будут использоваться в их системах. Инструменты программирования служб обеспечивают доступ к приложениям больших данных как к сервису. Специалистам data science иногда приходится открывать доступ к своим моделям через службы.

Самым известным примером такого рода являются REST·службы; сокращение REST обозначает «Representational State Transfer», т. е. «передача состояния представления. Эти службы часто используются для передачи данных веб-сайтам.

*БЕЗОПАСНОСТЬ*

Вы хотите, чтобы все желающие могли получить доступ к вашим данным? Вероятно, при этом нужно организовать механизм точного управления доступом к данным, но делать это на уровне каждого отдельного приложения не хочется. Средства безопасности больших данных позволяют создать централизованную и высокоточную систему управления доступом к данным. Безопасность больших данных превратилась в самостоятельную область исследований, и специалисты data science обычно сталкиваются с ней в роли потребителей данных, редко, когда им приходится реализовывать средства безопасности самостоятельно.

# 5. ПРИМЕР ИСПОЛЬЗОВАНИЯ HADOOP

*Hortonworks Sandbox* — виртуальная машина, созданная *Hortonwork*s для тестирования приложений больших данных на локальной машине.

Воспользуемся средой *Hortonworks Sandbox* в *VirtualBox*. *VirtualBox* — инструмент виртуализации, позволяющий запустить другую операционную систему в вашей операционной системе.

[Установка Hadoop на Windows – Блог программиста (pro-prof.com)](https://pro-prof.com/forums/topic/%D1%83%D1%81%D1%82%D0%B0%D0%BD%D0%BE%D0%B2%D0%BA%D0%B0-hadoop-%D0%BD%D0%B0-windows?ysclid=m0wjw1hlxz571799627)

https://pro-prof.com/forums/topic/установка-hadoop-на-windows?ysclid=m0wjw1hlxz571799627

Для запуска Sandbox в Virtualbox необходимо выполнить несколько простых шагов:

1. Загрузите виртуальный образ по адресу http://hortonworks.com/products/hortonworks-sandbox/#install.

2. Запустите управляющую виртуальную машину. VirtualBox можно загрузить по адресу https://www.virtualbox.org/wiki/Downloads.

3. Нажмите Ctrl+I и выберите виртуальный образ Hortonworks.

4. Щелкните на кнопке Next.

5. Щелкните на кнопке Import; через непродолжительное время ваш образ будет успешно импортирован.

6. Выберите свою виртуальную машину и щелкните на кнопке Run.

7. Немного подождите запуска CentOS с установкой Hadoop. Обратите внимание: на иллюстрации используется Sandbox версии 2.1. В других версиях возможны небольшие изменения.

На запущенную машину можно войти напрямую или воспользоваться SSH. Мы воспользуемся веб-интерфейсом. Введите в браузере адрес http://127.0.0.1:8000

В поставку Hortonworks также включены два учебных набора данных, находящиеся на уровне HCatalog. Щелкните на кнопке HCat, чтобы просмотреть список доступных таблиц.

Чтобы просмотреть данные, щелкните на кнопке Browse Data рядом с пунктом sample\_07 для перехода к следующему экрану.

На первый взгляд таблица кажется совершенно обычной. Hive — инструмент, позволяющий работать с этими данными как с обычной базой данных на языке SQL.

Да, все верно: в Hive для получения результатов используется HiveQL — диалект традиционного языка SQL. Чтобы открыть редактор Beeswax HiveQL, щелкните на кнопке Beeswax в меню.

Для получения результатов выполните следующий запрос:

Select description, avg(salary) as average\_salary from sample\_07 group by description order by average\_salary desc.

Щелкните на кнопке Execute. Hive преобразует запрос HiveQL в задание MapReduce и выполняет его в среде Hadoop.

Если это ваш первый запрос, выполнение может занять до 30 секунд — Hadoop известен своей долгой инициализацией.

Через некоторое время на экране появляется результат. Его анализ подтверждает, что среди самых высокооплачиваемых профессий немало врачей (по крайней мере по данным США).

# ИТОГ

Большие данные — обобщающий термин для любых наборов данных, достаточно больших и сложных, чтобы их можно было обработать традиционными средствами работы с данными. Большие данные характеризуются «четырьмя V»: объемом, разнообразием, скоростью и достоверностью.

Основным содержанием data science являются методы анализа наборов данных, от совсем небольших до гигантских.

Процесс data science сам по себе нелинеен, но его можно разделить на несколько шагов:

1. Назначение цели исследования.

2. Сбор данных.

3. Подготовка данных.

4. Исследование данных.

5. Моделирование.

6. Отображение и автоматизация.

Набор технологий больших данных вовсе не сводится к Hadoop. Он состоит из множества разных технологий, которые можно разбить на следующие категории:

• Файловая система.

• Инфраструктуры распределенного программирования.

• Интеграция данных.

• Базы данных.

• Машинное обучение.

• Безопасность.

• Планирование.

• Сравнительный анализ.

• Развертывание.

• Программирование служб.

Не все категории больших данных интенсивно используются специалистами data science. В основном они занимаются файловыми системами, инфраструктурами распределенного программирования, базами данных и машинным обучением. Конечно, им приходится иметь дело с другими компонентами, и все же эти предметные области относятся к сфере деятельности других профессий.

Данные могут существовать во многих формах. Основные формы:

• Структурированные данные.

• Неструктурированные данные.

• Данные на естественном языке.

• Машинные данные.

• Графовые данные.

• Потоковые данные