Chapter Two
Introduction to Data Science

Outline

2. Introduction to Data Science

Topics Covered

- Overview for Data Science
  - Definition of data and information
  - Data types and representation
- Data Value Chain
  - Data Acquisition
  - Data Analysis
  - Data Curating
  - Data Storage
  - Data Usage
- Basic concepts of Big data

2.1 Overview of Data Science

- **Defination:**
  - ***Data science*** is a multi-disciplinary field that uses scientific methods, processes, algorithms and systems to ***extract knowledge*** and insights from ***structured***, ***semi structured*** and ***unstructured data.***
- **Data science need**
  - a strong quantitative background in statistics and linear algebra
  - Programming knowledge with focuses in data warehousing, mining and modeling to build and analyze algorithms.
- It is one of the most promising and in-demand career paths for skilled professionals.
  - Data sceince professionals are called ***data scientists***

Overview of Data Science(2)

Data scientists need to

- be curious and result-oriented,
- have industry-specific knowledge and
- have good communication skills that allow them to explain highly technical results to their non-technical counterparts.

- This chapter cover basic definitions of data and information, data types and representation, data value change and basic concepts of big data.

2.1.1 Definition of data and information

What is data?

- Data can be defined as ***a representation of facts, concepts, or instructions*** in a formalized manner, which should be suitable for communication, interpretation, or processing by human or electronic machine.
- Data is represented with the help of characters such as alphabets (A-Z, a-z), digits (0-9) or special characters (+,-,/,*,<,>,= etc.)

What is Information?

- Information is organized or classified data, which has some meaningful values for the receiver.
  - It is the processed data on which decisions and actions are based.
- For the decision to be meaningful, the processed data must qualify for the following characteristics
  - **Timely** – Information should be available when required.
  - **Accuracy** – Information should be accurate.
  - **Completeness** – Information should be complete.

Summary

Summery: Data Vs. Information	
Data	Information
Described as unprocessed or raw facts and figures	Described as processed data
Cannot help in decision making	Can help in decision making
Raw material that can be organized, structured, and interpreted to create useful information systems.	Interpreted data; created from organized, structured, and processed data in a particular context.
‘groups of non-random’ symbols in the form of text, images, and voice representing quantities, action and objects'.	Processed data in the form of text, images, and voice representing quantities, action and objects'.

Data Processing Cycle

- ***Data processing*** is the re-structuring or re-ordering of data by people or machine to increase their usefulness and add values for a particular purpose.
- Data processing consists of the following basic steps
  - **Input step** – the input data is prepared in some convenient form for processing.
    - The form depends on the processing machine.
    - For example - when electronic computers are used – input medium options include magnetic disks, tapes, and so on.
  - **Processing step** – the input data is changed to produce data in a more useful form.
    - For example - pay-checks can be calculated from the time cards, or
    - a summary of sales for the month can be calculated from the sales orders.
  - **Output step** – the result of the preceeding processing step is collected.
    - The particular form of the output data depends on the use of the data.
    - For example - output data may be pay-checks for employees.

Data types and its representation – (based on programming language)

- Data type(type) is an attribute of data which tells the compiler or interpreter how the programmer intends to use the data.
- Common data types in programming languages include:
  - Integers, Booleans, Characters, floating-point numbers, alphanumeric strings (strings)
- A data type constrains the values that an expression, such as a variable or a function, might take.
- This data type defines
  - the operations that can be done on the data,
  - the meaning of the data, and
  - the way values of that type can be stored.
- On other hand, for the analysis of data, there are three common types of data types or structures: ***Structured data***, **unstructured *data***, and ***semi-structured data***.

Data types/structure – based on analysis of data

- **Structured Data, unstructured data, semi-structured data, and metadata**
- **Structured Data**
  - Structured data is data that adheres to a pre-defined data model and is therefore straightforward to analyze.
  - Usually, conforms to a tabular format with relationship between the different rows and columns.
    - Common examples are Excel files or SQL databases.
  - Structured data depends on the existence of a data model
    - a model of how data can be stored, processed and accessed.

Cont...

- **Structured Data**
  - Because of a data model, each field can be accessesed separately or jointly along with data from other fields.
  - It is possible to quickly aggregate data from various locations in the database.
  - Structured data is considered the most ‘traditional’ form of data storage.
    - since the earliest versions of database management systems (DBMS) were able to store, process and access structured data.

Data types/structure – based on analysis of data

Unstructured Data

- does not have a predefined data model or is not organized in a pre-defined manner.
- It is without proper formatting and alignment
- Unstructured information is typically ***text-heavy***, but may contain data such as dates, numbers, and facts as well.
- This results in irregularities and ambiguities that make it difficult to understand using traditional programs as compared to data stored in structured databases.
  - Common examples include: audio, video files or No-SQL databases.

- The ability to store and process unstructured data has greatly grown in recent years, with many new technologies

Cont...

- Tools that are able to store unstructured data.
  - For example:
    - ***MongoDB*** is optimized to store documents.
    - ***Apache Graph*** - is optimized for storing relationships between nodes.
- The ability to analyze unstructured data is especially relevant in the context of Big Data, since a large part of data in organizations is unstructured.
  - Think about pictures, videos or PDF documents.
- The ability to extract value from unstructured data is one of main drivers behind the quick growth of Big Data.

Data types/structure – based on analysis of data

Semi-structured Data

- Semi-structured data is a form of structured data that does not conform with the formal structure of data models associated with relational databases or other forms of data tables
- Contain tags or other markers to separate semantic elements and enforce hierarchies of records and fields within the data.
  - Therefore, it is also known as self-describing structure.
- Examples of Semi-Structured data
  - JSON and XML are forms of semi-structured data.
- Semi-structured data is considerably easier to analyze than unstructured data.
- Many Big Data solutions and tools have the ability to ‘read’ and process either JSON or XML.

Example of Semi-Structured and Structured data

```
[
    {
        first_name  : "Kathleen",
        last_name   : "Jo",
        order_id    : "123456",
        order_total : "12.34"
    },
    {
        first_name  : "John",
        last_name   : "Doe",
        order_id    : "098765",
        order_total : "98.76
    }
]
```

First Name	Last_Name	Order_id	Order_total
Kathleen	Jo	123456	12.34
John	Doe	098765	98.76

Metadata – Data about Data

- Metadata is data about data.
- From a technical point of view, metadata is not a separate data structure, but it is one of the most important elements for Big Data analysis and big data solutions.
- It Just provides additional information about a specific set of data.
- **Example:**
  - In a set of photographs, metadata could describe when and where the photos were taken.
  - The metadata then provides fields for dates and locations which, by themselves, can be considered structured data.
- Because of this reason, metadata is frequently used by Big Data solutions for initial analysis.

2.2 Data value Chain

- DVC describe the information flow within a big data system as a series of steps needed to generate value and useful insights from data.
- The Big Data Value Chain identifies the following key high-level activities:
  - Data acquisition
  - Data analysis
  - Data curation
  - Dat usage

Data Value Chain Activities

Data Acquisition

- It is the process of **gathering**, **filtering**, and **cleaning** data before it is put in a data storage (database, warehouse etc)
- Data acquisition is one of the major big data *challenges in terms of infrastructure requirements.*
- The infrastructure required to support the acquisition of big data must
  - deliver low, predictable latency in both capturing data and in executing queries;
  - be able to handle very high transaction volumes, often in a distributed environment; and
  - support flexible and dynamic data structures.

.

Data Value Chain Activities

Data Analysis

- Data analysis involves exploring, transforming, and modelling data with the goal of highlighting relevant data, synthesizing and extracting useful hidden information with high potential from a business point of view.
- It is concerned with making the raw data acquired amenable to use in decision-making as well as domain-specific usage.
- Related areas include data mining, business intelligence, and machine learning (covered in Chapter 4)

Data Value Chain Activities

Data Curation

- It is the active management of data over its life cycle to ensure it meets the necessary data quality requirements for its effective usage.
- Data curation processes can be categorized into different activities such as content creation, selection, classification, transformation, validation, and preservation.
- Data curation is performed by expert curators that are responsible for improving the accessibility and quality of data.
- Data curators (also known as scientific curators, or data annotators) hold the responsibility of ensuring that data are trustworthy, discoverable, accessible, reusable, and fit their purpose.
- A key trend for the curation of big data utilizes community and crowd sourcing approaches.

Data Value Chain Activities

Data Storage

- It is the persistence and management of data in a scalable way that satisfies the needs of applications that require fast access to the data.
- Relational Database Management Systems (RDBMS) have been the main, and almost unique, solution to the storage paradigm for nearly 40 years.
- Relational DBMS lacks flexibility with regard to schema changes and the performance and fault tolerance when data volumes and complexity grow, making them unsuitable for big data scenarios.
- NoSQL technologies have been designed with the scalability goal in mind and present a wide range of solutions based on alternative data models.

Data Value Chain Activities

Data Usage

- Data usage in business decision-making can enhance competitiveness through
  - reduction of costs, increased added value, etc
- It covers the data-driven business activities that need access to data, its analysis, and the tools needed to integrate the data analysis within the business activity.
- While the problem of working with data that exceeds the computing power or storage of a single computer is not new, the pervasiveness, scale, and value of this type of computing has greatly expanded in recent years.
- Big data is a blanket term for the non-traditional strategies and technologies needed to gather, organize, process, and gather insights from large datasets.

Big Data

What Is Big Data?

- An exact definition of “big data” is difficult to nail down because projects, vendors, practitioners, and business professionals use it quite differently. With that in mind, generally speaking, big data is:
  1. large datasets
  2. the category of computing strategies and technologies that are used to handle large datasets
- In this context, “large dataset” means a dataset too large to reasonably process or store with traditional tooling or on a single computer.
- This means that the common scale of big datasets is constantly shifting and may vary significantly from organization to organization.

Big Data

- The basic requirements for working with big data are the same as the requirements for working with datasets of any size.
- However, the massive scale, the speed of ingesting and processing, and the characteristics of the data that must be dealt with at each stage of the process present significant new challenges when designing solutions.
- The goal of most big data systems is to surface insights and connections from large volumes of heterogeneous data that would not be possible using conventional methods.
- In 2001, Gartner's Doug Laney first presented what became known as **the "three Vs of big data"** to describe some of the characteristics that make big data different from other data processing:

Big Data

The "three Vs of big data"/Characterstics of Big Data/

1. **Volume**

- The sheer scale of the information processed helps define big data systems.
- These datasets can be orders of magnitude larger than traditional datasets, which demands more thought at each stage of the processing and storage life cycle.
- Often, because the work requirements exceed the capabilities of a single computer, this becomes a challenge of pooling, allocating, and coordinating resources from groups of computers.
- Cluster management and algorithms capable of breaking tasks into smaller pieces become increasingly important.

2. Velocity

- Another way in which big data differs significantly from other data systems is the speed that information moves through the system.
- Data is frequently flowing into the system from multiple sources and is often expected to be processed in real time to gain insights and update the current understanding of the system.
- This focus on near instant feedback has driven many big data practitioners away from a batch-oriented approach and closer to a real-time streaming system.
- Data is constantly being added, massaged, processed, and analyzed in order to keep up with the influx of new information and to surface valuable information early when it is most relevant.
- These ideas require robust systems with highly available components to guard against failures along the data pipeline.

3. Variety

- Big data problems are often unique because of the wide range of both the sources being processed and their relative quality.
- Data can be ingested from internal systems like application and server logs, from social media feeds and other external APIs, from physical device sensors, and from other providers.
- Big data seeks to handle potentially useful data regardless of where it's coming from by consolidating all information into a single system.
- The formats and types of media can vary significantly as well. Rich media like images, video files, and audio recordings are ingested alongside text files, structured logs, etc.
- While more traditional data processing systems might expect data to enter the pipeline already labeled, formatted, and organized, big data systems usually accept and store data closer to its raw state.
- Ideally, any transformations or changes to the raw data will happen in memory at the time of processing.

Other Characterstics of Big Data

Other V's

- The other V's which tended to describe challenges rather than qualities of big data. The additions include:

- **Veracity**: The variety of sources and the complexity of the processing can lead to challenges in evaluating the quality of the data (and consequently, the quality of the resulting analysis)
- **Variability**: Variation in the data leads to wide variation in quality. Additional resources may be needed to identify, process, or filter low quality data to make it more useful.
- **Value**: The ultimate challenge of big data is delivering value. Sometimes, the systems and processes in place are complex enough that using the data and extracting actual value can become difficult.

Big Data Life Cycle

- The general categories of activities involved with big data processing are:
  - *Ingesting data into the system*
    - taking raw data and adding it to the system.
    - Requires Extraction Transformation Loading(ETL) process
  - *Persisting the data in storage*
    - *Store ingested data onto disks (Distributed file system)*
  - *Computing and Analyzing data*
    - *Process the data to surface insights*
    - *Approchs:* ***batch processing, Realtime processing***
  - *Visualizing the results*
    - *Spot trends and make sense of large number of data points*
    - The data changes frequently and large deltas *are indicator of the health of the system/organization*

Clustered Computing

- Because of the quantities of big data, individual computers are often inadequate
  - Therefore, to address the high storage and computational needs of big data, **computer clusters** are a better fit.
- Big data clustering software combines the resources of many smaller machines, to provide a number of benefits:

1. **Resource Pooling**: Combining the available **storage space** ,**CPU** and **memory** is extremely important.
   - Processing large datasets requires large amounts these three resources.
2. **High Availability**: Clusters can provide better fault tolerance and availability to prevent hardware or software failures from affecting access to data and processing.
   - This is increasingly important to real-time analytics.

Clustered Computing(2)

3. **Easy Scalability**: Clusters make it easy to scale horizontally by adding additional machines to the group. This means the system can react to changes in resource requirements without expanding the physical resources on a machine.

- Using clusters requires a solution for
  - managing cluster membership,
  - coordinating resource sharing, and
  - scheduling actual work on individual nodes. Solution for cluster membership and resource allocation include:
- software like Hadoop's YARN (which stands for Yet Another Resource Negotiator) or Apache Mesos.
- The machines involved in the computing cluster are also typically involved with the management of a distributed storage system (*discuss in data persistence).*

Step 1: Ingesting Data into the System(2)

- Data ingestion is the process of taking raw data and adding it to the system.
- Complexity of this operation depends -
  - heavily on the format and quality of the data sources and
  - how far the data is from the desired state prior to processing.
- Dedicated ingestion tools that can add data to a big data system are.
  - **Apache Sqoop** – technologies that can take existing data from relational databases and add it to a big data system.
  - Similarly, **Apache Flume and Apache Chukwa** are projects designed to aggregate and import application and server logs.
  - Queuing systems like **Apache Kafka** can also be used as an interface between various data generators and a big data system.
  - Ingestion frameworks like **Gobblin** can help to aggregate and normalize the output of these tools at the end of the ingestion pipeline.

Step 1: Ingesting Data into the System

- In the ingestion process - some level of analysis, sorting, and labelling usually takes place.
- This process is sometimes called ETL (stands for extract, transform, and load).
  - This term conventionally refers to legacy data warehousing processes
- Typical operations might include
  - modifying the incoming data to format it,
  - categorizing and labelling data,
  - filtering out unneeded or bad data, or potentially validating that it adheres to certain requirements.

Step 2: Persisting the Data in Storage

- The ingestion processes typically hand the data off to the components that manage storage, so that it can be reliably persisted to disk.
- Although looks simple operation, the volume of incoming data, the requirements for availability, and the distributed computing layer make more complex storage systems necessary.
- This usually means leveraging a distributed file system for raw data storage.
- Solutions like Apache Hadoop's HDFS filesystem allow large quantities of data to be written across multiple nodes in the cluster.
- This ensures that the data can be accessed by compute resources, can be loaded into the cluster's RAM for in-memory operations, and can gracefully handle component failures.

Step 2: Persisting the Data in Storage(2)

- Other distributed filesystems can be used in place of HDFS including **Ceph** and **GlusterFS**.
- Data can also be imported into other distributed systems for more structured access.
- Distributed databases, especially NoSQL databases, are well-suited for this role.
  - Because they are often designed with fault tolerant considerations and can handle heterogeneous data.
- Many different types of distributed databases available to choose from depending on how you want to organize and present the data.

Step 3: Computing and Analyzing Data

- Once the data is available, the system can begin processing the data to surface actual information.
- The computation layer is perhaps the most diverse part of the system.
  - the requirements and best approach can vary significantly depending on what type of insights desired.
- Data is often processed repeatedly - either iteratively by a single tool or by using a number of tools to surface different types of insights.
- **Two main method of processing: Batch and Real-time**
- **Batch processing** is one method of computing over a large dataset.
- The process involves: breaking work up into smaller pieces, scheduling each piece on an individual machine, reshuffling the data based on the intermediate results, and then calculating and assembling the final result.
- These steps are often referred: splitting, mapping, shuffling, reducing, and assembling, or collectively as a distributed map reduce algorithm.
- This is the strategy used by Apache Hadoop's MapReduce.
- Batch processing is most useful when dealing with very large datasets that require quite a bit of computation.

- **Real-time processing** - demands that information be processed and made ready immediately and requires the system to react as new information becomes available.
- One way of achieving this is **stream processing**
  - which operates on a continuous stream of data composed of individual items.
- Apache Storm, Apache Flink, and Apache Spark provide different ways of achieving real-time or near real-time processing.
  - There are trade-offs with each of these technologies
- In general, real-time processing is best suited for analyzing smaller chunks of data that are changing or being added to the system rapidly.

Step 4: Visualizing the Results

- Recognizing trends or changes in data over time is often more important than the values themselves.
- Visualizing data is one of the most useful ways to spot trends and make sense of a large number of data points.
- Real-time processing is frequently used to visualize application and server metrics.
- The data changes frequently and large deltas in the metrics typically indicate significant impacts on the health of the systems or organization.
- Projects like **Prometheus** can be useful for processing the data streams as a time-series database and visualizing that information.
- **Elastic Stack** – is one popular way of visualizing data, formerly known as the ELK stack.
  - Composed of Logstash for data collection, Elasticsearch for indexing data, and Kibana for visualization, the Elastic stack can be used with big data systems to visually interface with the results of calculations or raw metrics.

Step 4: Visualizing the Results (2)

- A similar stack can be achieved using Apache Solr for indexing and a Kibana fork called Banana for visualization.
  - The stack created by these is called Silk.
- Another visualization technology typically used for interactive data science work is a data "**notebook**".
- These projects allow for interactive exploration and visualization of the data in a format conducive to sharing, presenting, or collaborating.
- Popular examples of this type of visualization interface are Jupyter Notebook and Apache Zeppelin.

Next Time!

Chapter 3: Intoroduction to Artificial Intillegence (AI)