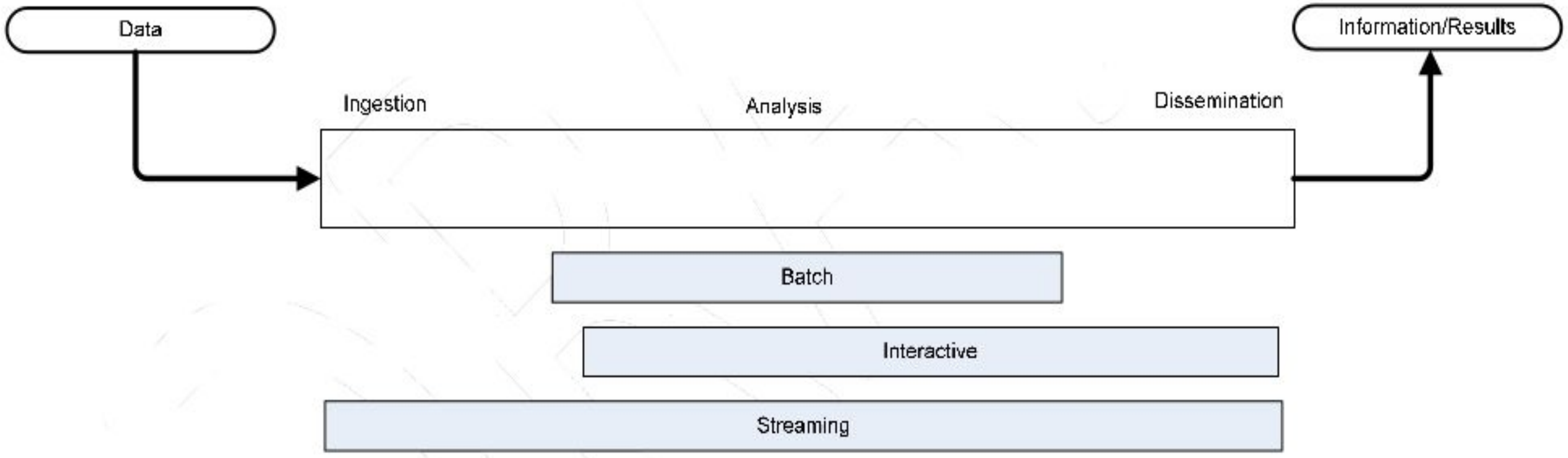
Big Data Processing

Breaking up computation to scale it

MONASH University

Overview: Processing



Interactive:	Bringing humans into the loop
Streaming:	Massive data streaming through system with little storage
Batch:	Data stored and analysed in large blocks, “batches” easier to develop and analyse

Processing Background Concepts

in-memory: In RAM, i.e., not going to disk

parallel processing: Performing tasks in parallel

distributed computing: Across multiple machines

scalability: To handle a growing amount of work; to be enlarged to accommodate growth (not just “big”)

data parallel: Processing can be done independently on separate chunks of data

RAM (Random Access Memory): store computer programs and data that CPU needs in real time. RAM data is volatile and is erased once computer is switched off.
HDD (Hard Disk Drive) : has permanent storage and it is used to store user specific data and operating system files.
Be sure to check out the SSD (Solid-State drive) if you are curious about their difference with HDD ;)

MONASH University

Distributed Analysis

Legacy systems provide powerful statistical tools on the desktop, e.g.

- SAS
- R
- Matlab

Often-times without distributed or multi-processor support

Supporting distributed/multi-processor computation requires special redesign of algorithms

Map-Reduce

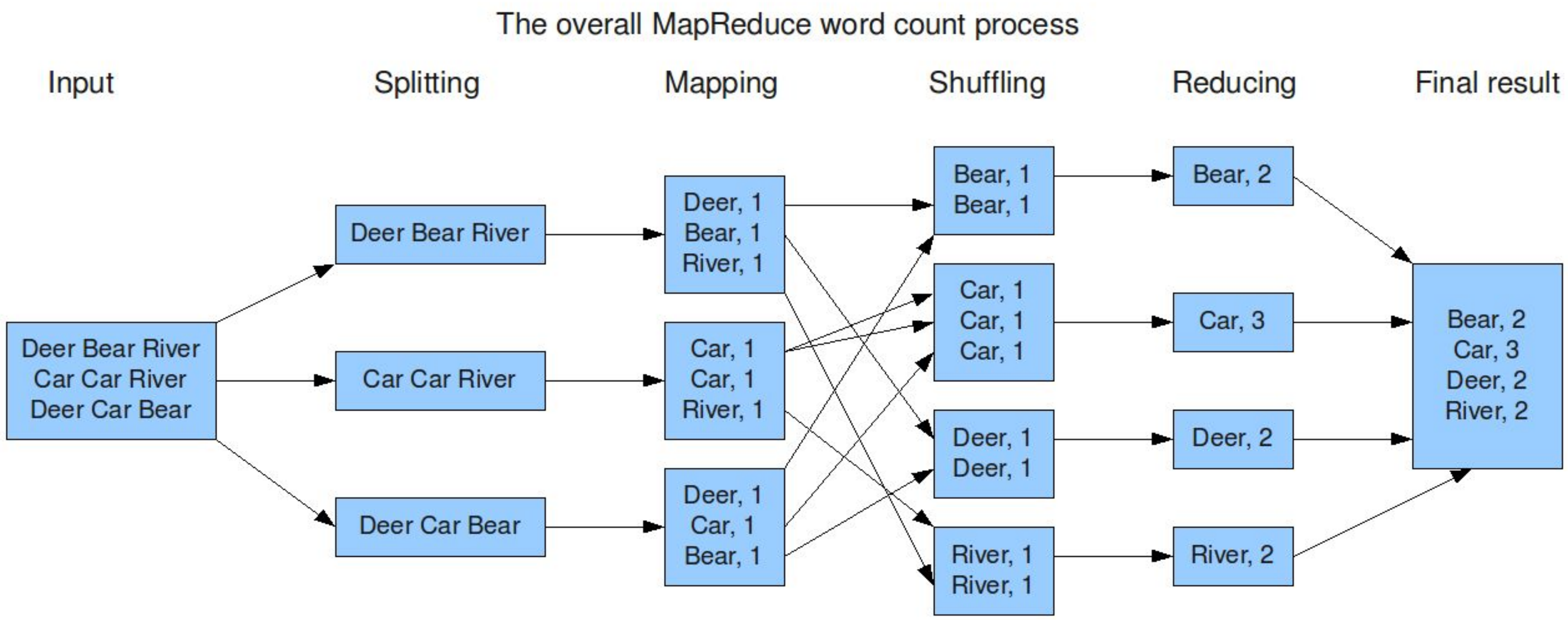
Simple distributed processing framework developed at Google

- published by Dean and Ghemawat of Google in 2004
- intended to run on commodity hardware; so has fault-tolerant infrastructure
- from a distributed systems perspective, is quite simple

Commodity hardware: Computer **hardware** that is affordable and easy to obtain. Typically it is a low-performance system that is IBM PC-compatible and is capable of running Microsoft Windows, Linux, or MS-DOS without requiring any special devices or equipment.

Map-Reduce

Example



For a simple word-count task: (1) divide data across machines

(2) `map()` to key-value pairs (3) `sort()` and `merge()` identical keys

Map-Reduce

Requires simple data parallelism followed by some merge ("reduce") process

- Stopped using by Google probably in 2005

- Google now uses "Cloud Dataflow", available commercially, as open source

Hadoop

Open-source Java implementation of Map-Reduce

- Originally developed by Doug Cutting while at Yahoo!
- Architecture:

  Common: Java libraries and utilities

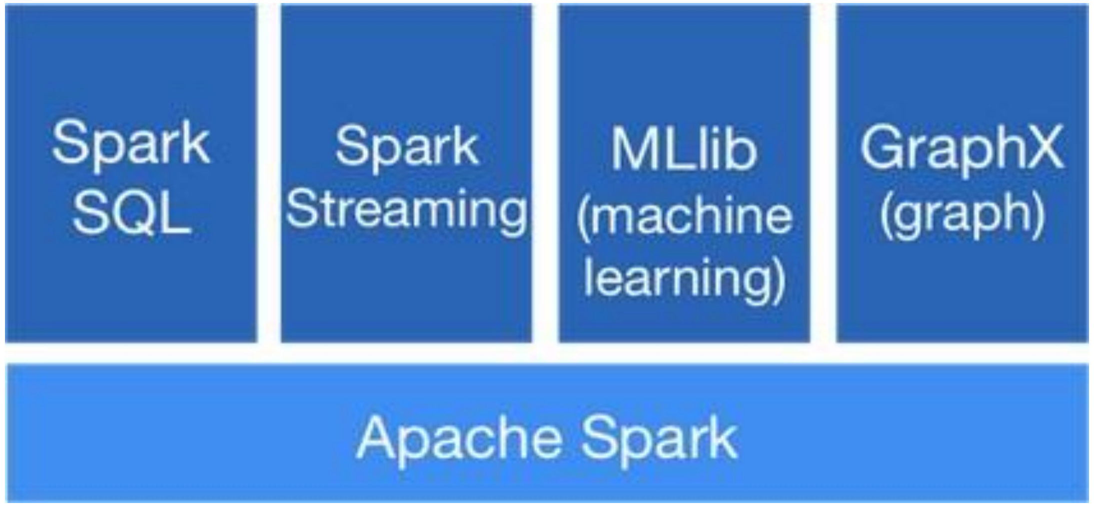
  MapReduce: Core paradigm
- Huge tool ecosystem
- Well passed the peak of the hype curve (referring to Gardner’s Hype Curve)

MONASH University

Spark

Another (open source) Apache top-level project at Apache Spark

- Developed at AMPLab at UC Berkeley
- Builds on Hadoop infrastructure
- Interfaces in Java, Scala, Python, R
- Provides in-memory analytics
- Works with some of the Hadoop ecosystem



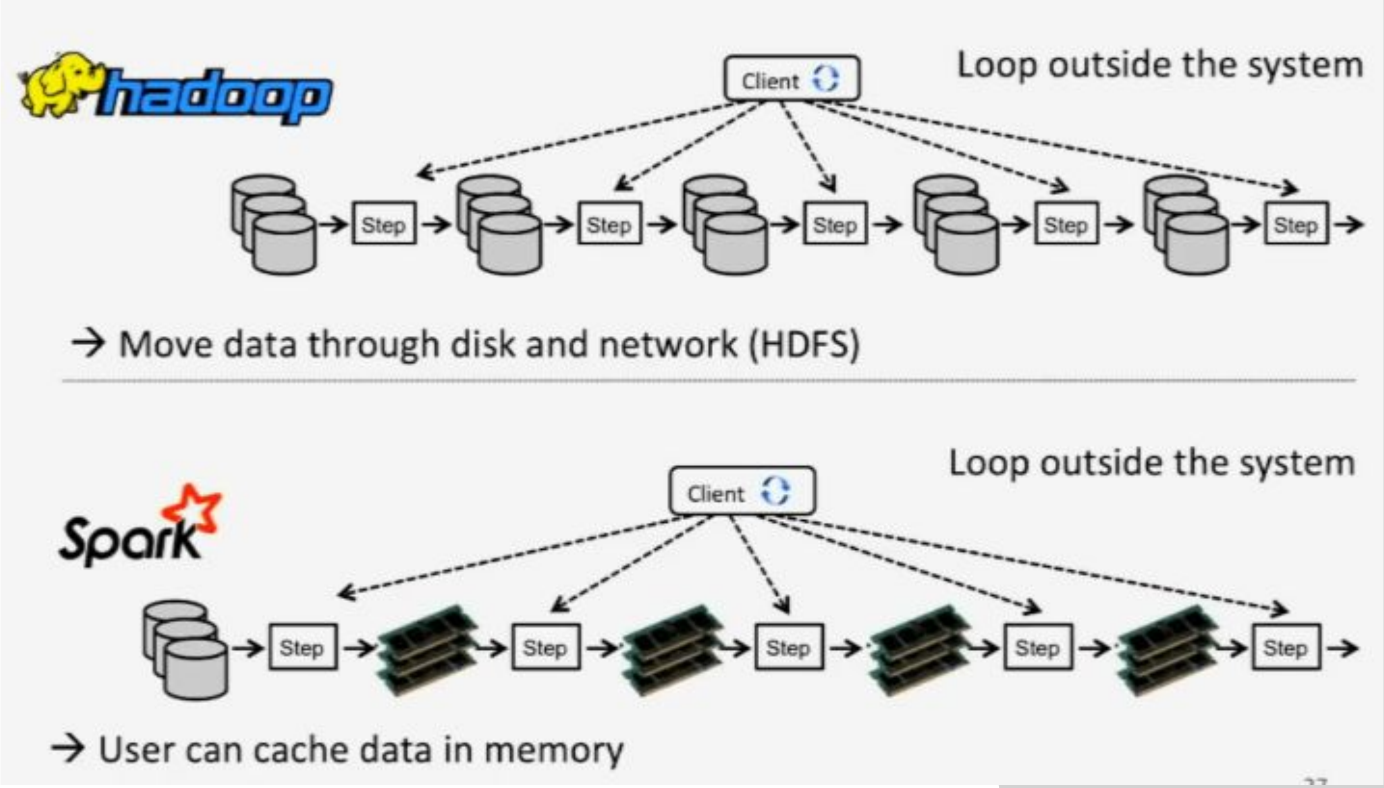
Summary: Hadoop and Spark

Hadoop provides an inexpensive and open source platform for parallelising processing:

- based on a simple Map-Reduce architecture
- not suited to streaming (suitable for offline processing)

Spark is a more recent development than Hadoop

- includes Map-Reduce capabilities
- provides real-time, in-memory processing
- much faster than Hadoop



MONASH University

flux.qa/GMBXRB

Which one of the following is suitable for real-time data processing?

A. Hadoop
B. Spark
C. Excel

Big Data Processing

Netflix Journey

MONASH University

Evolution of the Netflix Data Pipeline

Here are some statistics about Netflix data pipeline:

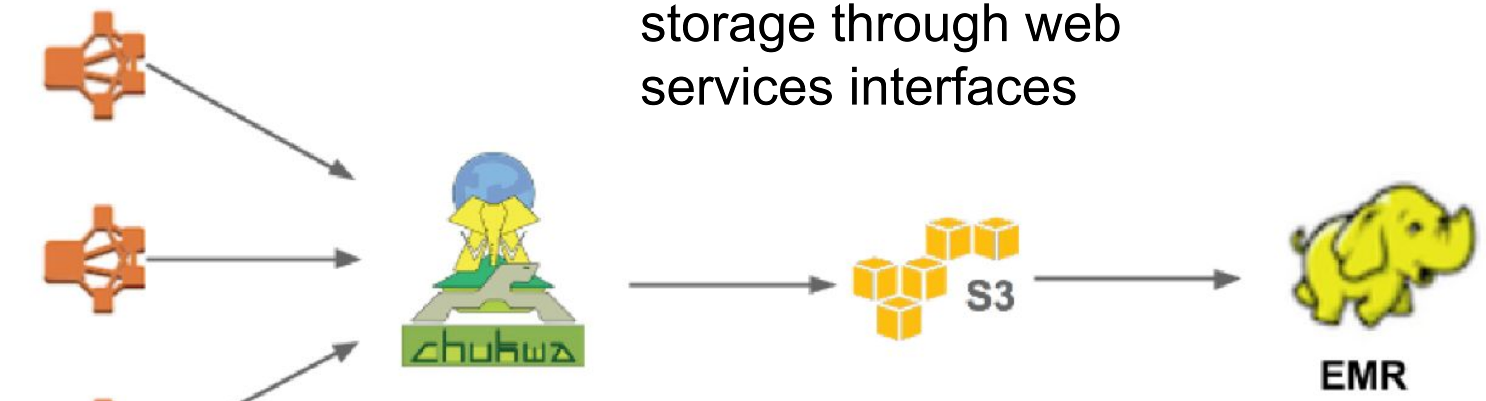
- ~500 billion events and ~1.3 PB per day
- ~8 million events and ~24 GB per second during peak hours

There are several hundred event streams flowing through the pipeline. For example:

- Video viewing activities
- UI activities
- Error logs
- Performance events
- Troubleshooting & diagnostic events

Netflix Data Pipeline: V1.0 Chukwa pipeline

Amazon S3 provides object storage through web services interfaces

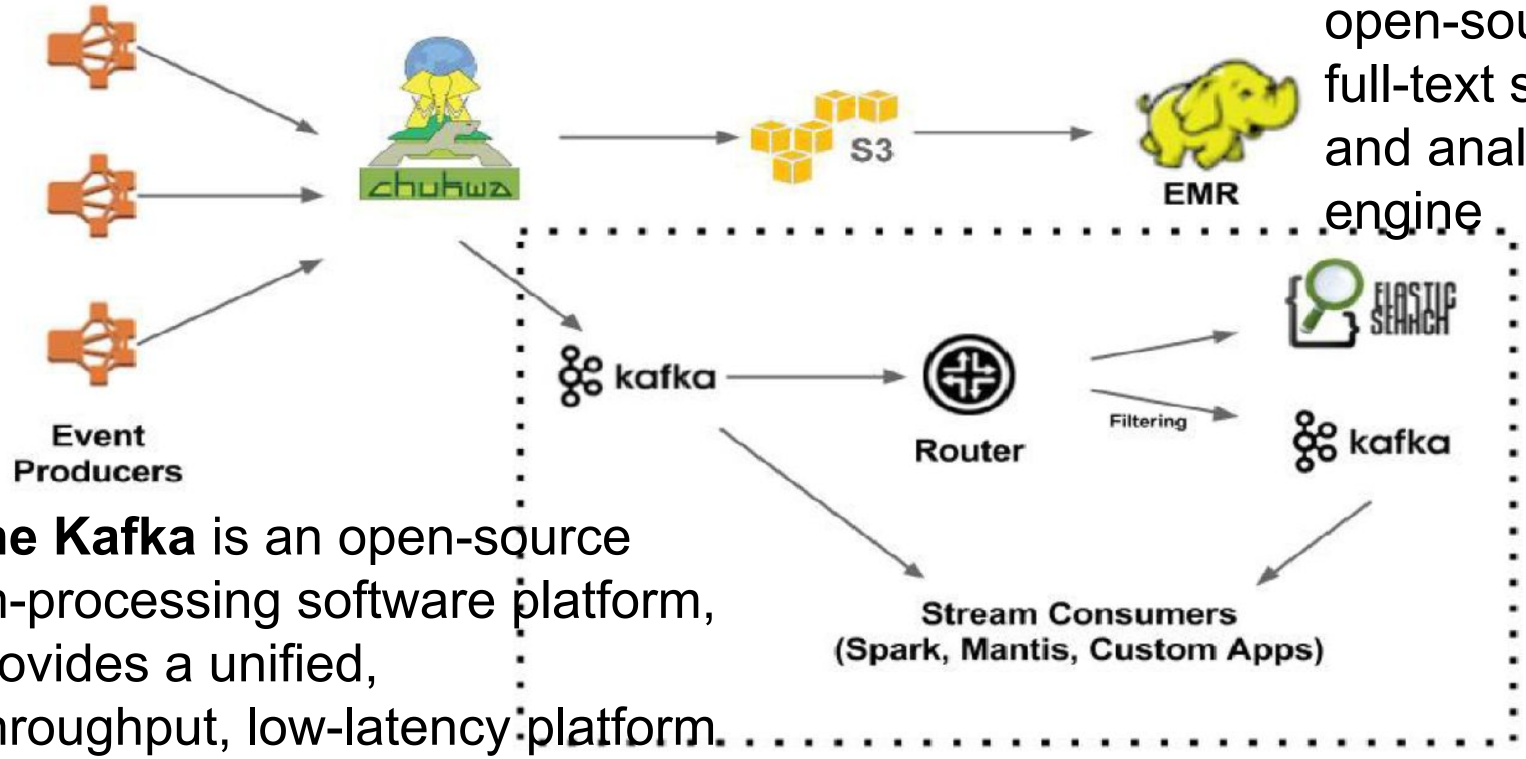


Apache Chukwa is an open source data collection system for monitoring large distributed systems

Amazon Elastic MapReduce (EMR) is a AWS tool for big data processing and analysis

Netflix Data Pipeline: V1.5 Chukwa pipeline with real-time branch

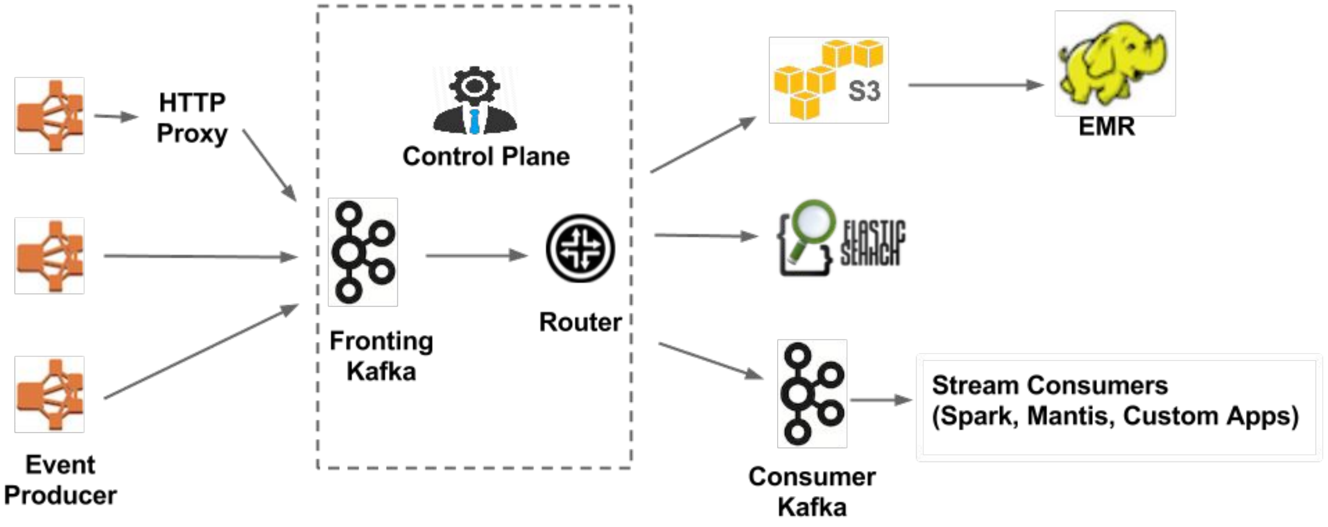
Elasticsearch is a highly scalable open-source full-text search and analytics engine



Apache Kafka is an open-source stream-processing software platform, that provides a unified, high-throughput, low-latency platform for handling real-time data feeds.

Netflix Data Stack

Simplified view using Apache Kafka, Elastic Search, AWS S3, Apache Spark, Apache Hadoop, and EMR.



see [Architecture of Giants: Data Stacks]()

Current Hot Topics

MONASH University

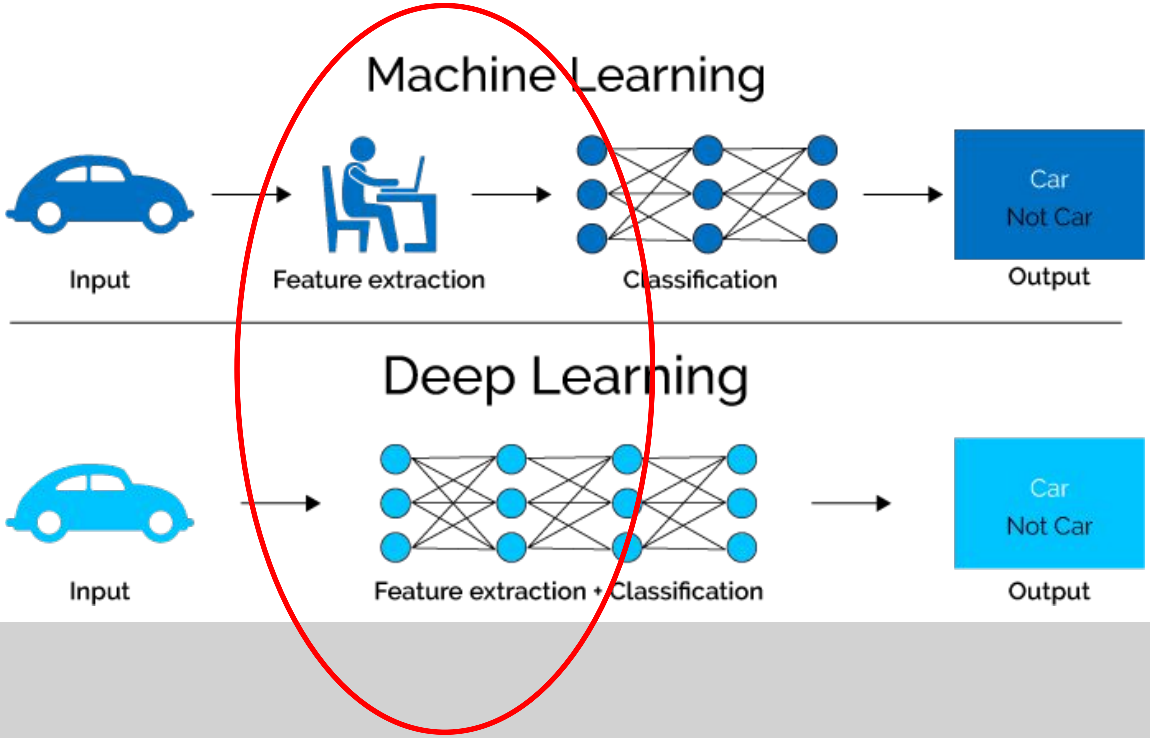
The Machine Learning Renaissance

Mike Olson (co-founded Cloudera in 2008) says without big data and a platform to manage big data, machine learning and artificial intelligence just don't work.

See *[the machine learning renaissance](#)* starting at 60 seconds.

Deep Learning

- A machine learning subfield of learning representations of data. Exceptional effective at learning patterns.
- Deep learning algorithms attempt to learn (multiple levels of) representation by using a hierarchy of multiple layers
- If you provide the system tons of information, it begins to understand it and respond in useful ways.



Reinforce Learning

- A machine learning and deep learning subfield.
- Ability for the machine to learn through trial and error once given the objective.
- DeepMind, bought by Google, responsible for AlphaGo. (We have already seen this earlier)

Reinforce Learning

Summary

Databases

- SQL vs NoSQL

Distributed Processing

- Hadoop
- Spark
- Map-Reduce

Learning Outcomes

Week 10

By the end of this week you should be able to:

- Characterize different database types
- Differentiate between SQL and NoSQL databases
- Define what distributed processing is
- Analyse the Map-Reduce framework
- Differentiate between Hadoop and Spark
- *Apply R/shell commands to read/manipulate big data files*

MONASH University