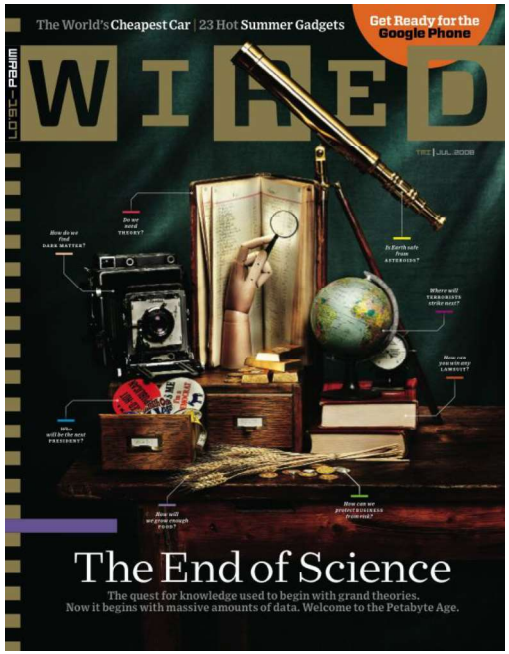
Khoa học dữ liệu Lớn

Phùng Quốc Định
Centre for Pattern Recognition and Data Analytics
Deakin University, Australia
Email: dinh.phung@deakin.edu.au
(published under **Dinh Phung**)

Outline

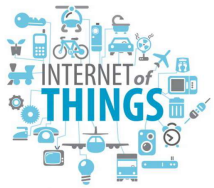
- Dữ liệu lớn (DLL) là gì?
  - Dữ liệu lớn từ đâu đến?
  - Cơ hội và thách thức của dữ liệu lớn
- Tiếp cận dữ liệu lớn như thế nào?
  - Quản lý dữ liệu lớn
  - Xử lý dữ liệu lớn
  - Tính toán phân tán và song song
  - Giới thiệu nền tảng công nghệ

Dữ liệu lớn là gì?



The quest for knowledge used to begin with grand theories. Now it begins with massive amounts of data. Welcome to the Petabyte Age.

"Chúng ta thường xây dựng lý thuyết trước khi khai phá kiến thức. Nhưng ngày nay, việc này thường lại bắt đầu từ dữ liệu trước. Thời đại dữ liệu lớn đã bắt đầu!"



Kết nối vạn vật

Tính toán đám mây

Dữ liệu lớn là gì?
What is big data

- DLL là các tập dữ liệu rất lớn và/hoặc rất phức tạp.
- Vượt quá khả năng kỹ thuật và lý thuyết truyền thống.
- DLL có ba đặc điểm quan trọng (3Vs).

Dòng dữ liệu không ngừng chuyển động

Dữ liệu đa dạng, khó điều khiển, từ cấu trúc đến không cấu trúc

.....
Zettabytes(10^{18})
Petabytes (10^{15})

Dữ liệu lớn là gì?
What is big data

Một zettabyte lớn thế nào?

Dự tính đến năm 2020 dữ liệu toàn cầu sẽ đạt khoảng 44 ZB.

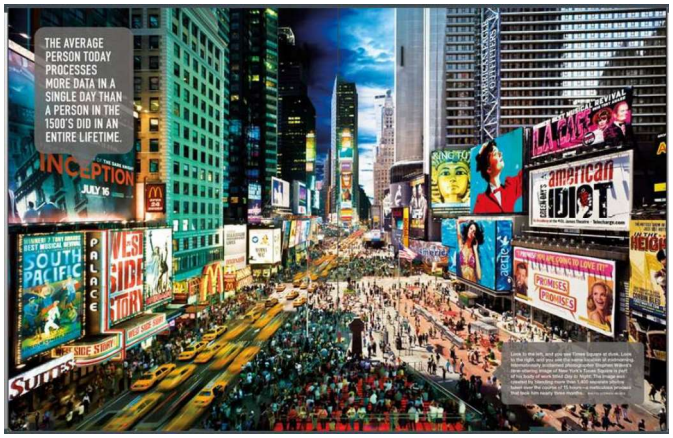
Nếu ta dùng nhiều máy iPad để chứa số dữ liệu này và chồng lên nhau, chúng sẽ lấp đầy hơn sáu lần khoảng cách từ trái đất đến mặt trăng.

[Nguồn: https://irfansalam.wordpress.com/tag/zetabyte/]

Dữ liệu lớn từ đâu đến?

- Dữ liệu lớn là gì?
  - Dữ liệu từ đâu đến?
  - Cơ hội và thách thức của DLL
- Tiếp cận dữ liệu lớn như thế nào?
  - Quản lý dữ liệu lớn
  - Xử lý dữ liệu lớn
  - Tính toán phân tán và song song
  - Giới thiệu nền tảng công nghệ

Dữ liệu lớn đến từ đâu?
Sources of data



"The average person today processes more data in a single day than a person in the 1500's did in an entire life time"

Every 60 seconds

- **98,000+** tweets
- **695,000** status updates
- **11 million** instant messages
- **698,445** Google searches
- **168 million+** emails sent
- **1,820TB** of data created
- **217** new mobile web users

[Nguồn: Smolan and Erwitt, The human face of big data, 2013]

Dữ liệu lớn đến từ đâu?
Sources of data

"Chỉ trong ngày đầu tiên một em bé sinh ra đời, số lượng dữ liệu thu thập được tương đương với 70 lần thông tin trong Thư viện Quốc hội Mỹ (The Library of Congress)"

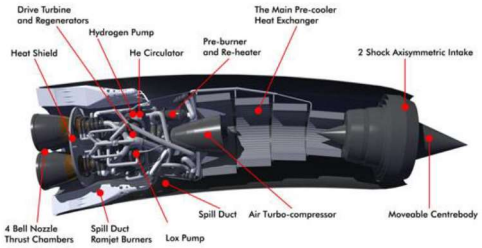
[Nguồn: Smolan and Erwitt, *The human face of big data*, 2013]

Dữ liệu lớn đến từ đâu?
Sources of data

Dữ liệu lớn đến từ đâu?
What drives big data

- "Lớn mà không to, to mà không lớn" [GS Hồ Tú Bảo]

(tương tác dữ liệu cảm biến)



Lớn mà không to

(đếm số lượng nhấp chuột trên toàn cầu)

To mà không lớn

Lean data vs big data?
Complexity or size?

Cơ hội và thách thức

- Dữ liệu lớn là gì?
  - Dữ liệu lớn từ đâu đến?
  - Cơ hội và thách thức của dữ liệu lớn
- Tiếp cận dữ liệu lớn như thế nào?
  - Quản lý dữ liệu lớn
  - Xử lý dữ liệu lớn
  - Tính toán phân tán và song song
  - Giới thiệu nền tảng công nghệ

DLL có thể đem lại những cơ hội gì?

- Dữ liệu lớn và lợi ích chiến lược của quốc gia
  - BQP Mỹ dành khoảng 250 triệu mỗi năm để khai khác DLL, nhằm nâng cao khả năng ra quyết định.

CINDER (Cyber-Insider Threat)

"Năm 2012, văn phòng chính sách khoa học và công nghệ của Mỹ thuộc Văn phòng điều hành của Tổng thống Mỹ đã công bố 84 chương trình về dữ liệu lớn thuộc 6 Bộ của Chính quyền Liên bang. Những chương trình này đề cập đến thách thức và cơ hội của cuộc cách mạng dữ liệu lớn và xem việc tìm lời giải cho vấn đề dữ liệu lớn là sứ mệnh của các cơ quan chính phủ cũng như của việc cách tân và khám phá khoa học"

DLL có thể đem lại những cơ hội gì?

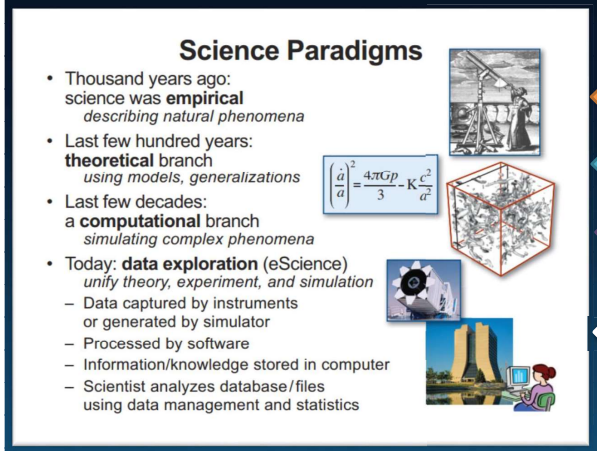
- Dữ liệu lớn thay đổi diện mạo doanh nghiệp, công ty công nghiệp và khởi nghiệp
- Các doanh nghiệp đã có thể truy cập tới các nguồn dữ liệu lớn:

  dữ liệu độc quyền = tài nguyên
- Ngành công nghiệp cũng sẽ thay đổi mạnh mẽ.
  Ví dụ: **advanced manufacturing process optimization**
- Cùng với sự phát triển mạnh mẽ của nghành khoa học dữ liệu (KHDL) là cơ hội khởi nghiệp

startups = ideas + KHDL + $$$?

DLL mang lại lợi ích gì?

- Khám phá khoa học dựa vào dữ liệu lớn

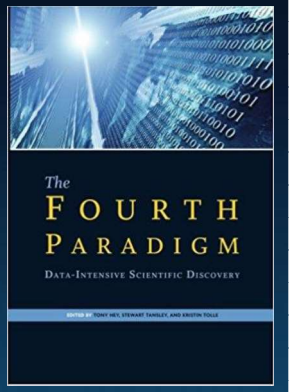


Thực nghiệm

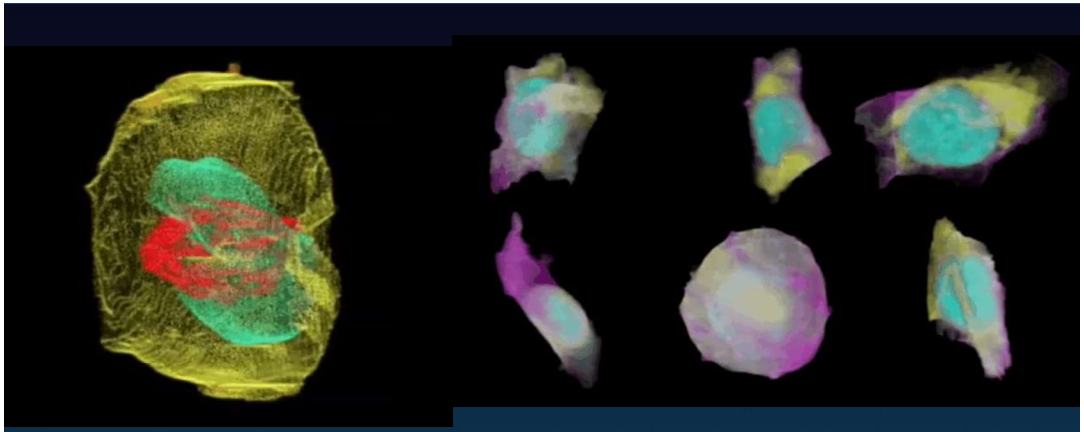
Lý thuyết

Tính toán và mô phỏng

Khám phá dữ liệu



Data-intensive Scientific Discovery

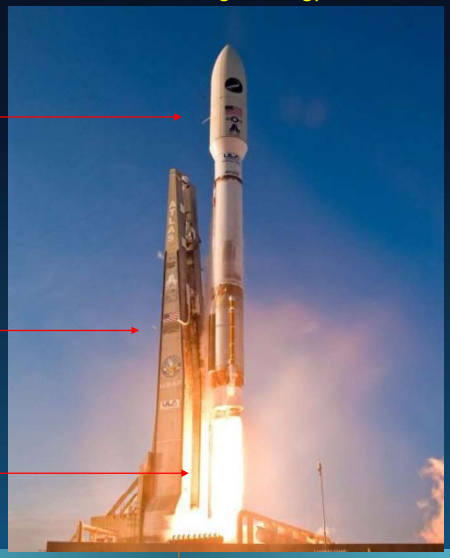


Machine learning predicts the look of stem cells, *Nature News*, April 2017
The Allen Cell Explorer Project

"No two stem cells are identical, even if they are genetic clones …. Computer scientists analysed thousands of the images using deep learning programs and found relationships between the locations of cellular structures. They then used that information to predict where the structures might be when the program was given just a couple of clues, such as the position of the nucleus. The program 'learned' by comparing its predictions to actual cells"

DLL mang lại ích gì?

Andrew Ng's Analogy



Mô hình tính toán lớn, e.g., deep learning/AI

Cơ sở hạ tầng tính toán, nhà nghiên cứu, môi trường và chính sách như bệ phóng

Dữ liệu lớn như nguồn năng lượng

Thách thức và vấn đề của DLL

Key challenges and issues with big data

- Data and storage overgrow computation!
  - Web, mobile, sensor, scientific, etc.
    - Facebook's daily logs: 60TB
    - 1,000 Genomes Projects: 200%B
    - Google Web index: 10+ PB
    - Cost of 1TB of disk: ~ $50
  - Storage getting cheaper
    - Size doubling every 18 months
  - Stalling CPU speeds and storage bottlenecks
    - Time to read 1TB from disk: 3 hours (100MB/S)

Cách tiếp cận và phương pháp phân tích dữ liệu trở thành chìa khóa quan trọng!

Thách thức và vấn đề của DLL

Key challenges and issues with big data

- Ethical issues (đạo đức)
  - breach of privacy, collection of data without informed consent
- Security and privacy (bảo mật và thông tin cá nhân)
  - the ease of stealing, including identity theft, the stealing of national security information
- Issue of exploitation (trục lợi bất hợp pháp)
  - commercial mining of information; targeting for commercial gain
- Issues of power and politics (quyền lực và chính trị)
  - the use of data to perpetuate particular views, ideologies
- Issues of truth (sự thật)
  - the perpetuation of falsehoods; propaganda
- Issues of social justice (công bằng trong xã hội)
  - information is overwhelmingly skewed towards certain groups and leaves others out of the 'digital revolution'. [Radika Gorur]

Thách thức và vấn đề của DLL

Key challenges and issues with big data

- Có phải cứ có nhiều dữ liệu thì càng tốt không?
- Điều này chưa chắc:
  - Nhầm lẫn noise/artefact với thông tin thật (more false positives).
  - Tăng giá thành lưu trữ dữ liệu và tính toán không hiệu quả.
  - Phức tạp hóa vấn đề không đúng cách.
  - Những mô hình phân tích dữ liệu tinh vi và hiệu quả có thể không ứng dụng được nữa.

Tóm tắt về DLL

- Dữ liệu lớn (DLL) xử lý tập dữ liệu rất lớn hoặc (đồng thời) rất phức tạp vượt quá giới hạn của công nghệ và kỹ thuật cổ điển.
- DLL có ba đặc tính quan trọng:
  - Kích thước rất lớn: petabytes, zettabytes
  - Dòng dữ liệu không ngừng chuyển động
  - Dữ liệu đa dạng, khó điều khiển, di chuyển từ cấu trúc (structured) sang không cấu trúc (unstructured data).
- DLL đến từ nhiều nguồn khác nhau và không ngừng lớn lên
  - Dữ liệu online, mạng xã hội.
  - Kết nối vạn vật (IoT) và thiết bị thông minh (smart devices, sensors).
  - Các giao dịch và dữ liệu trong doanh nghiệp.
- DLL đem lại nhiều cơ hội
  - Lợi ích chiến lược quốc gia
  - Doanh nghiệp và khởi nghiệp
  - Khám phá khoa học
- Nhưng cũng đặt ra nhiều thách thức và lắm cạm bẫy

Tọa đàm | Panel discussion

Câu hỏi 1: Có cần quan tâm đến dữ liệu lớn không?

Câu hỏi 2: Có phải cứ có nhiều dữ liệu là tốt không?

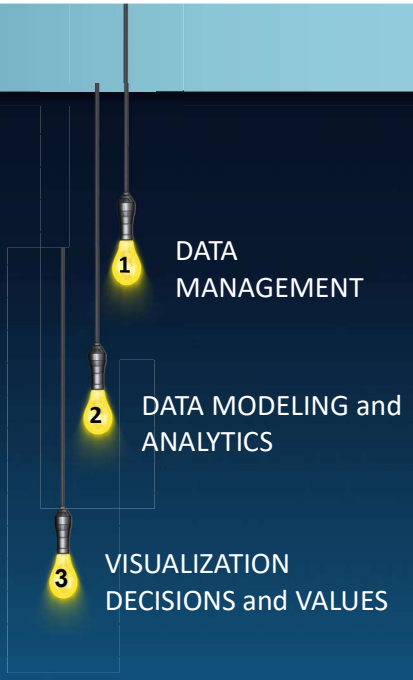
Câu hỏi 3: Dữ liệu lớn hay dữ liệu nhỏ? Lean data or big data?

Tiếp cận dữ liệu lớn như thế nào?

- Dữ liệu lớn là gì?
  - Dữ liệu lớn từ đâu đến?
  - Cơ hội và thách thức của dữ liệu lớn
- Tiếp cận dữ liệu lớn như thế nào?
  - Quản lý dữ liệu lớn
  - Xử lý dữ liệu lớn
  - Tính toán phân tán và song song
  - Giới thiệu nền tảng công nghệ

Chìa khóa của dữ liệu lớn

- Đâu là chìa khóa khoa học và công nghệ của DLL?
  - Quản trị dữ liệu, tức lưu trữ, bảo trì và truy nhập các nguồn dữ liệu lớn.
  - Phân tích dữ liệu, tức tìm cách hiểu được dữ liệu và tìm ra các thông tin hoặc tri thức quý báu từ dữ liệu.
  - Trao đổi, hiển thị dữ liệu và kết quả phân tích dữ liệu để tạo ra sản phẩm hay giá trị.



Chìa khóa của dữ liệu lớn

- Hỏi gì khi tiếp cận DLL?

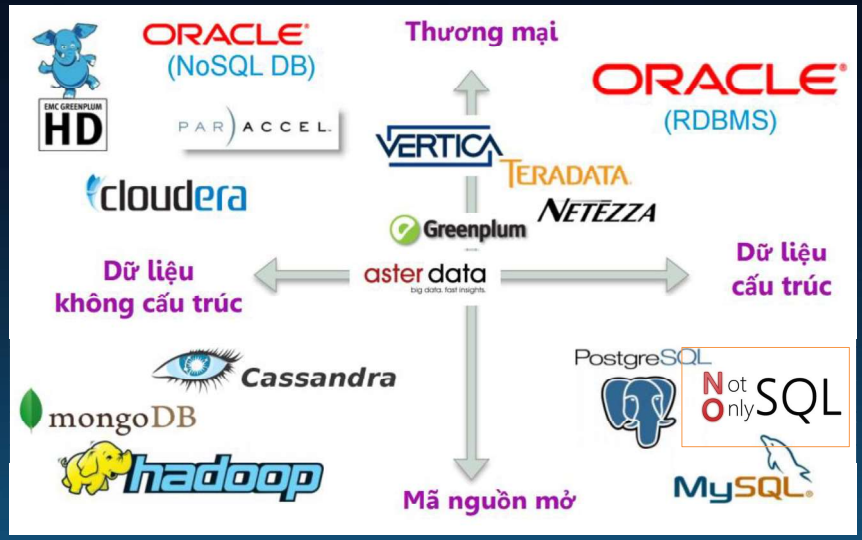
FUNDAMENTAL CONCERNS
How quickly do we need to get the results?
How big is the data to be processed?
Does the model building require several iterations or a single iteration?

SYSTEM CONCERNS
Will there be a need for more data processing capability in the future?
Is the rate of data transfer critical for this application?
Is there a need for handling hardware failures within the application?

TECHNOLOGY CONCERNS
What are the infrastructures (cloud/physical systems) to be used?
What are the technologies to be used for distributed/parallel processing?
Is there a need to invest into researching a new model?

[Reddy and Singh, A Survey on platforms for big data analytics, *Journal of Big Data*, 2014]

Quản lý dữ liệu lớn
Big data management



[Nguồn: Cisco]

Xử lý dữ liệu lớn
Big data processing

- Key technology to process big data
  - Scaling out
  - Distributed computing
  - Parallel computing
- Important open source technologies:
  - MapReduce
  - Hadoop
  - Spark
  - TensorFlow

Xử lý dữ liệu lớn
Những thuật ngữ quan trọng

- Scalability
  - the ability of the system to cope with the growth of data, computation and complexity without compromising the services and its core functionalities.
- Data I/O performance
  - the rate at which the data is transferred to/from a peripheral device.
- Fault tolerance
  - the capability of continuing operating properly in the event of a failure of one or more components.

[Reddy and Singh, A Survey on platforms for big data analytics, *Journal of Big Data*, 2014]

Xử lý dữ liệu lớn
Những thuật ngữ quan trọng

- Real-time processing
  - the ability to process the data and produce the results strictly within certain time constraints.
- Data size supported
  - the size of the dataset that a system can process and handle efficiently.
- Iterative tasks support
  - the ability of a system to efficiently support iterative tasks.

[Reddy and Singh, A Survey on platforms for big data analytics, *Journal of Big Data*, 2014]

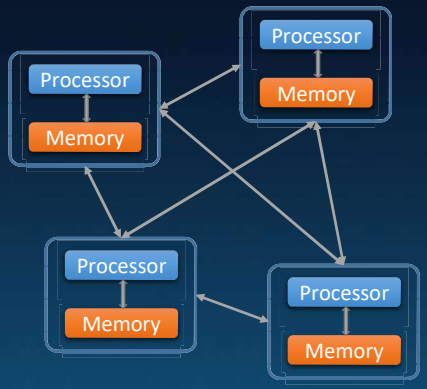
Tính toán phân tán và song song

Distributed vs Parallel computation

- Dữ liệu lớn là gì?
  - Dữ liệu lớn từ đâu đến?
  - Cơ hội và thách thức của DLL
- Tiếp cận dữ liệu lớn như thế nào?
  - Quản lý dữ liệu lớn
  - Xử lý dữ liệu lớn
  - Tính toán phân tán và song song
  - Giới thiệu nền tảng công nghệ

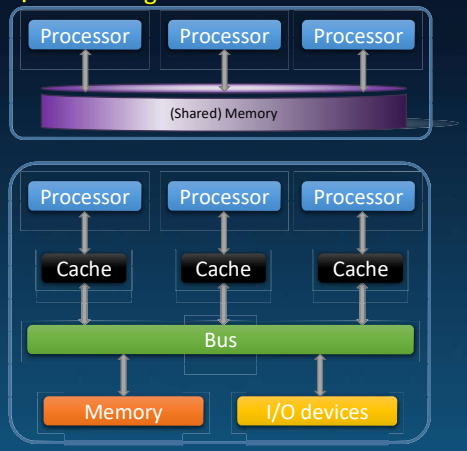
Tính toán phân tán và song song
Distributed and Parallel computation

Tính toán phân tán: bài toán được chia nhỏ thành cụm và phân tán vào nhiều máy khác nhau; mỗi máy có một bộ nhớ riêng.



VS

Tính toán song song: bài toán có cấu trúc tính toán song song, được chia nhỏ vào nhiều bộ xử lý để tính song song có cùng bộ nhớ chung.



Key difference: private vs shared memory

Tính toán phân tán và song song
Distributed and Parallel computation

Phân tính dữ liệu = Mô hình + Dữ liệu

Song song hóa dữ liệu (data parallelism):

Dữ liệu được chia thành cụm và chạy song song trên cùng một mô hình

Song song hóa mô hình (model parallelism):

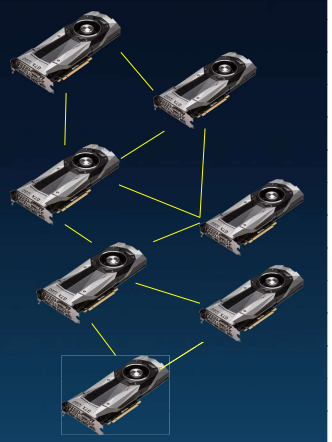
Nhiều mô hình chạy trên cùng một tập dữ liệu (e.g., Gibbs sampling)

Tính toán phân tán và song song
Tính song song bằng phần cứng với GPU và CPU đa nhân

- Multicore CPU
  - machine with dozens of processing cores
  - parallelism achieved through multithreading
  - Drawbacks:
    - limited number of processing cores.
    - limited memory few hundred gigabytes.

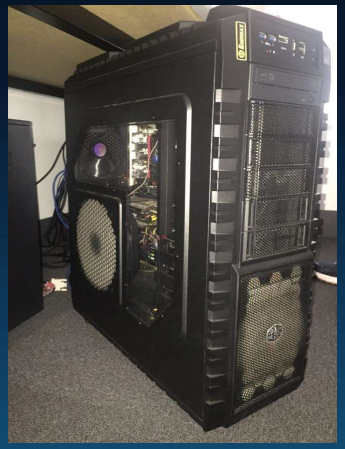
Tính toán phân tán và song song
Tính song song bằng phần cứng với GPU và CPU đa nhân

- Hardware: GPUs
  - highly parallel simple processors
  - large number of processing cores (2K+)
  - orders of magnitude speedup compared with multicore CPU.
  - Drawbacks:
    - limited memory (12GB memory per GPU)
    - few software and algorithms that are available for GPUs.



Tính toán phân tán và song song
Tính song song bằng phần cứng với GPU và CPU đa nhân

- Some examples we used (~25K AUD)
  - 1 x Ubuntu machine
    - CPU: Intel® Core™ i7-2600K @ 3.40GHz (8M Cache, up to 3.80Ghz, 4 cores, 8 threads)
    - RAM: 16GB
    - HDD: 2TB + 12TB (Qnap-NAS)
    - GPU: 3x NVIDIA GeForce GTX 590 (1024 cores, 3GB memory)

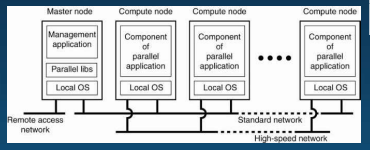


Tính toán phân tán và song song
Tính song song bằng phần cứng với GPU và CPU đa nhân

- Some examples we used (~70K AUD, ARC Grant)
  - 5 NVIDIA DEVCUBE (20K USD)
    - RAM: 128GB
    - SSD: 256GB + 480GB
    - HDD: 8TB mirror
    - GPU: 4x NVIDIA TITAN X (3072 cores, 1000 MHz, 12GB memory)
    - Design specially for Deep Learning
    - Theano, Tensorflow, Torch, Caffe
    - Python
    - Linux packages and libraries
    - Develop and experiment parallel algorithms, such as for DL, on single GPU or multiple GPUs

Tính toán phân tán và song song
Xử lý phân tán với hệ thống cluster

- Cluster Computing Systems
  - a collection of similar workstations or PCs, closely connected by a high-speed LAN, each node runs the same operating system
- Advantages
  - Economical: 15x cheaper than traditional supercomputers with the same performance
  - Scalability: Easy to upgrade and maintain
  - Reliability: continuing to operate even in case of partial failures
- Disadvantages
  - Difficult to manage and organize a large number of computers
  - Low data I/O performance
  - Not suitable for real-time processing



http://csis.pace.edu/~marchese/CS865/Lectures/Chap1/Chapter1a_files/image007.jpg

Tính toán phân tán và song song
Xử lý phân tán với hệ thống cluster

- Some examples we used (~120K AUD, ARC Grant)

- CPU: Intel® Xeon® E5-26700 @ 2.60GHz (8 cores, 16 threads)
- Processing cards: Intel Xeon Phi coprocessor cards (60 cores)
- RAM: 128GB
- HDD: 24TB

Clusters of 8 CentOS machines

Server name	IP
gandalf-1.it.deakin.edu.au	10.120.0.241
gandalf-2.it.deakin.edu.au	10.120.0.242
gandalf-3.it.deakin.edu.au	10.120.0.243
gandalf-4.it.deakin.edu.au	10.120.0.244
gandalf-5.it.deakin.edu.au	10.120.0.245
gandalf-6.it.deakin.edu.au	10.120.0.246
gandalf-7.it.deakin.edu.au	10.120.0.247
gandalf-8.it.deakin.edu.au	10.120.0.248

Tính toán phân tán và song song
Xử lý phân tán trên cloud

- Provided by large IT companies
  - Google Cloud Platform
  - Amazon Web Services
  - Microsoft Azure
- Advantages
  - low investing and maintaining cost
  - anywhere and at anytime accessibility
  - high scalability
- Disadvantages
  - data security
  - dependency on the provider
  - a constant internet connection
  - migration issue

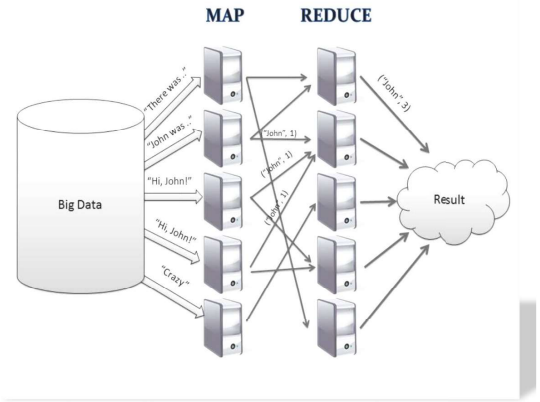
Giới thiệu một số công nghệ quan trọng

MapReduce, Hadoop, Spark, TensorFlow, MLlib

- Dữ liệu lớn là gì?
- Dữ liệu lớn từ đâu đến?
- Cơ hội và thách thức của dữ liệu lớn
- Tiếp cận dữ liệu lớn như thế nào?
  - Quản lý dữ liệu lớn
  - Xử lý dữ liệu lớn
  - Tính toán phân tán và song song
  - Giới thiệu nền tảng công nghệ

Công nghệ MapReduce
Parallel processing with MapReduce

- Nhu cầu xử lý DLL tăng nhanh
- MapReduce
  - invented by Google in 2004 and used in Hadoop.
  - breaking the entire task into two parts: mappers and reducers.
  - mappers: read the data from HDFS, process it and generate some intermediate results.
  - reducers: aggregate the intermediate results to generate the final output.
- Key Limitations
  - inefficiency in running iterative algorithms.
  - Mappers read the same data again and again from the disk.

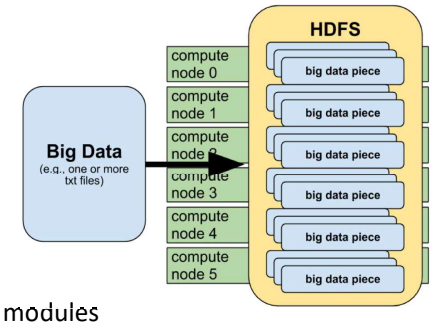


Data **parallelization** | **Distributed** execution | Outcome **aggregation**

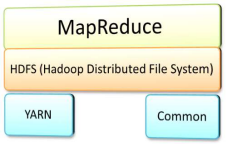
[Nguồn: https://www.supinfo.com/articles/single/2807-introduction-to-the-mapreduce-life-cycle]

Công nghệ Hadoop
Apache Hadoop

- Apache Hadoop:
  - an open source framework for storing and processing large datasets using clusters of commodity hardware
  - highly fault tolerant: (dữ liệu nhân thành 3 bản)
  - scaling up to 100s or 1000s of nodes



- Hadoop components:
  - **Common**: utilities that support the other Hadoop modules
  - **YARN**: a framework for job scheduling and cluster resource management.
  - **HDFS**: a distributed file system
  - **MapReduce**: computation model for parallel processing of large datasets.

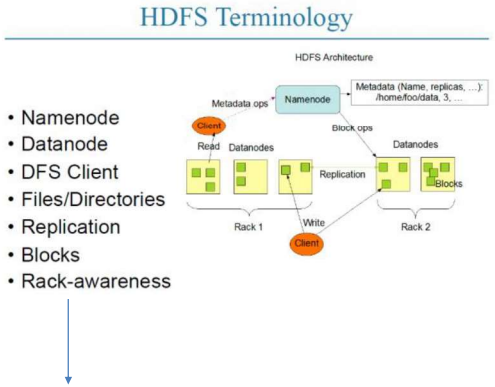


- Key limitation: not suitable for iterative-convergent algorithm!

[Nguồn: opensource.com]

Công nghệ Hadoop
Apache Hadoop

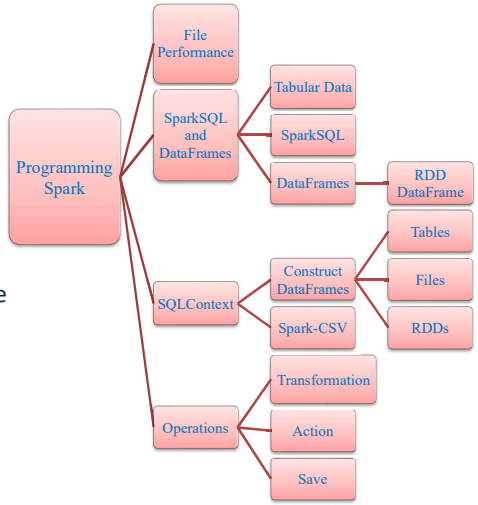
- Hadoop Distributed File System (HDFS)
  - a distributed file-system that stores data on the commodity machines, providing very high aggregate bandwidth across the cluster.
  - designed for large-scale distributed data processing under frameworks such as MapReduce.
  - store big data (e.g., 100TB) as a single file (we only need to deal with a single file)
  - fault tolerance: each block of data is replicated over DataNodes. The redundancy of data allows Hadoop to recover should a single node fail -> reminiscent to RAID architecture



"With a rack-aware file system, the JobTracker knows which node contains the data, and which other machines are nearby. If the work cannot be hosted on the actual node where the data resides, priority is given to nodes in the same rack. This reduces network traffic on the main backbone network".

Công nghệ Spark
Apache Spark

- Key motivation: suitable for iterative-convergent algorithms!
- Inherits all features of Hadoop
- Spark key features:
  - Resilient Distributed Datasets (RDD)
    - Read-only, partitioned collection of records distributed across cluster, stored in memory or disk.
    - Data processing = graph of transforms where nodes = RDDs and edges = transforms.
  - Benefits:
    - Fault tolerant: highly resilient due to RDDs
    - Cacheable: store some RDDs in RAM, hence faster than Hadoop MR for iteration.
    - Support MapReduce.



Spark vs MapReduce

[Nguồn: adato]

Spark vs Hadoop

Sorted 100 TB of data on disk in 23 minutes; Previous world record set by Hadoop MapReduce used 2100 machines and took 72 minutes.

This means that Apache Spark sorted the same data 3X faster using 10X fewer machines.

"Winning this benchmark as a general, fault-tolerant system marks an important milestone for the Spark project".

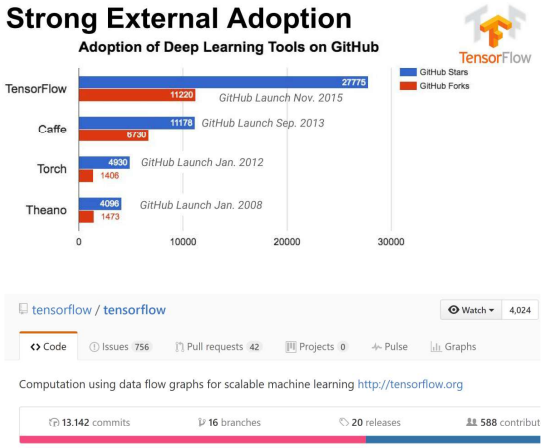
	Hadoop MR Record	Spark Record	Spark 1 PB
Data Size	102.5 TB	100 TB	1000 TB
Elapsed Time	72 mins	23 mins	234 mins
# Nodes	2100	206	190
# Cores	50400 physical	6592 virtualized	6080 virtualized
Cluster disk throughput	3150 GB/s (est.)	618 GB/s	570 GB/s
Sort Benchmark Daytona Rules	Yes	Yes	No
Network	dedicated data center, 10Gbps	virtualized (EC2) 10Gbps network	virtualized (EC2) 10Gbps network
Sort rate	**1.42 TB/min**	**4.27 TB/min**	**4.27 TB/min**
Sort rate/node	**0.67 GB/min**	**20.7 GB/min**	**22.5 GB/min**

[Nguồn: https://databricks.com]

Công nghệ TensorFlow

Parallel processing with TensorFlow

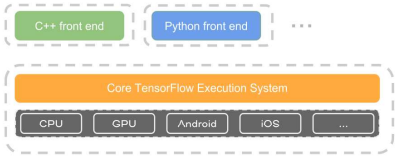
- TensorFlow
  - open-source framework for deep learning, developed by the GoogleBrain team.
  - provides primitives for defining functions on tensors and automatically computing their derivatives.



Công nghệ TensorFlow

Parallel processing with TensorFlow

- Back-end in C++: very low overhead
- Front-end in Python or C++: friendly programming language



- … and even in more platforms

Linux CPU	Linux GPU	Mac OS CPU	Windows CPU	Android
build passing	build passing	build passing	build passing	build passing

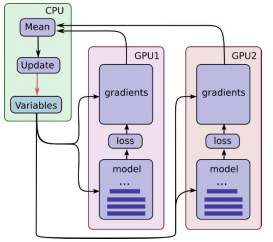
- Switchable between CPUs and GPUs

```
import tensorflow as tf
with tf.device('/cpu:'):
    a = tf.constant(1.0)
```

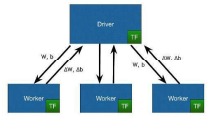
⇔

```
import tensorflow as tf
with tf.device('/gpu:0'):
    a = tf.constant(1.0)
```

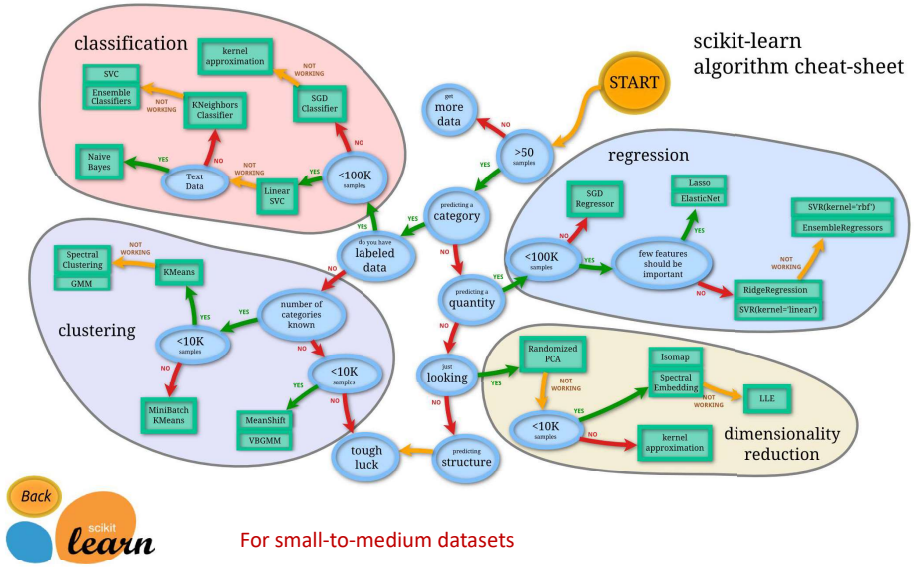
- Multiple GPUs in one machine or distributed over multiple machines




```
import tensorflow as tf
with tf.device('/gpu:0'):
    a = tf.constant(1.0)
with tf.device('/gpu:1'):
    b = tf.constant(2.1)
```



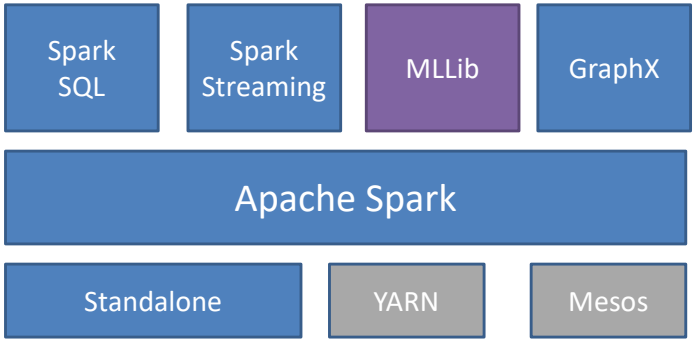
Mô hình lớn cho dữ liệu lớn
Scikit-learn cho dữ liệu vừa và nhỏ



For small-to-medium datasets

Mô hình lớn cho dữ liệu lớn
Scaling up ML and statistical models

- Most of machine learning and statistical algorithms are iterative-convergent!
- This is because most of them are optimization-based methods (e.g., Coordinate Descent, SGD) or statistical inference algorithms (e.g, MCMC, Variational, SVI).
- And, these algorithms are iterative in nature!



Mô hình lớn cho dữ liệu lớn
Scaling up ML and statistical models

Spark MLlib

- MLlib history
  - A platform on Spark providing scalable machine learning and statistical modelling algorithms.
  - Developed from AMPLab, UC Berkeley and shipped with Spark since 2013.
- MLlib algorithms
  - Classification
    - Linear models (linear SVMs, logistic regression)
    - Naïve Bayes
    - Least squares
    - Classification tree
    - Ensembles of trees (Random Forests and Gradient-Boosted Trees)
  - Regression
    - Generalized linear models (GLMs)
    - Regression tree
    - Isotonic regression
  - Clustering
    - K-means
    - Gaussian mixture
    - Power iteration clustering (PIC)
    - Latent Dirichlet Allocation (LDA)
    - Streaming k-means
  - Collaborative filtering (recommender system)
    - Alternating least squares (ALS),
    - Non-negative matrix factorization (NMF)
  - Dimensionality reduction
    - Singular value decomposition (SVD)
    - Principal component analysis (PCA)
  - Optimization
    - SGD, L-BFGS

Mô hình lớn cho dữ liệu lớn
Scaling up ML and statistical models

- Machine learning and statistical models for big data
  - Why traditional methods might fail on big data?
  - Three approaches to scale up big models
    - Scale up Gradient Descent Methods
    - Scale up Probabilistic Inference
    - Scale up Model Evaluation (bootstrap)
  - Open sources for big models
    - Mllib, Tensorflow
- Three approaches to build your own big models
  - Data augmentation
  - Stochastic Variational Inference for Graphical Models
  - Stochastic Gradient Descent and Online Learning

Bài Giảng Tiếp Theo

Tóm tắt những thông điệp chính

- Dữ liệu lớn là gì?
  - Dữ liệu từ đâu đến?
  - Cơ hội và thách thức của DLL
- Tiếp cận dữ liệu lớn như thế nào?
  - Quản lý dữ liệu lớn
  - Xử lý dữ liệu lớn
  - Tính toán phân tán và song song
  - Giới thiệu nền tảng công nghệ

- DLL ngày càng phổ biến và có ba đặc tính quan trong: Kích thước rất lớn, Dữ liệu đa dạng khó điều khiển và Dòng dữ liệu không ngừng chuyển động.
- DLL đem lại nhiều cơ hội nhưng cũng đặt ra nhiều thách thức.
- Khi tiếp cận DLL có ba bước ta cần quan tâm:
  - Lưu trữ dữ liệu như thế nào?
  - Dùng công nghệ gì để xử lý dữ liệu?
  - Dùng công cụ và mô hình gì để phân tích dữ liệu?
- Những nền tảng công nghệ mã nguồn mở chính cho DLL để tham khảo bao gồm:
  - Hadoop + MapReduce
  - Spark
  - TensorFlow (chủ yếu cho deep learning)

Acknowledgement and References

Tài liệu tham khảo

- Một số hình ảnh minh họa được download từ images.google.com theo cài đặc mặc định của search engine này.
- Eric Xing and Qirong Ho, *A New Look at the System Algorithm and Theory Foundations of Distributed Machine Learning*, KDD Tutorial 2015.
- Michael Jordan, *On the Computational and Statistical Interface and "Big Data"*, ICML Keynote Speech, 2014.
- Hồ Tú Bảo, *Dữ liệu lớn: Cơ hội và thách thức*, Tia Sáng, 2012
- Dilpreet Singh and Chandan Reddy, *A survey on platforms for big data analytics*, Journal of Big Data, 2014.

Xin cảm ơn các anh, chị đã lắng nghe!