

## Scalable and Emerging Information System Techniques

### 8.1. Techniques for voluminous data

1. **A/B testing:** A technique in which a control group is compared with a variety of test groups in order to determine what treatments (i.e., changes) will improve a given objective variable, e.g., marketing response rate. This technique is also known as split testing or bucket testing.
2. **Association rule learning:** A set of techniques for discovering interesting relationships, i.e., “association rules,” among variables in large databases.
3. **Classification.** A set of techniques to identify the categories in which new data points belong, based on a training set containing data points that have already been categorized
4. **Cluster analysis.** A statistical method for classifying objects that splits a diverse group into smaller groups of similar objects, whose characteristics of similarity are not known in advance. An example of cluster analysis is segmenting consumers into self-similar groups for targeted marketing
5. **Crowdsourcing.** A technique for collecting data submitted by a large group of people or community (i.e., the “crowd”) through an open call, usually through networked media such as the Web
6. **Data fusion and data integration.** A set of techniques that integrate and analyze data from multiple sources in order to develop insights in ways that are more efficient and potentially more accurate than if they were developed by analyzing a single source of data.
7. **Data mining.** A set of techniques to extract patterns from large datasets by combining methods from statistics and machine learning with database management. These techniques include association rule learning, cluster analysis, classification, and regression
8. **Ensemble learning.** Using multiple predictive models (each developed using statistics and/or machine learning) to obtain better predictive performance than could be obtained from any of the constituent models.
9. **Genetic algorithms.** A technique used for optimization that is inspired by the process of natural evolution or “survival of the fittest.” In this technique, potential solutions are encoded as “chromosomes” that can combine and mutate.

- 10. Machine learning.** A subspecialty of computer science (within a field historically called “artificial intelligence”) concerned with the design and development of algorithms that allow computers to evolve behaviors based on empirical data
- 11. Natural language processing (NLP).** A set of techniques from a subspecialty of computer science (within a field historically called “artificial intelligence”) and linguistics that uses computer algorithms to analyze human (natural) language.
- 12. Neural networks.** Computational models, inspired by the structure and workings of biological neural networks (i.e., the cells and connections within a brain), that find patterns in data. Neural networks are well-suited for finding nonlinear patterns. They can be used for pattern recognition and optimization.
- 13. Network analysis.** A set of techniques used to characterize relationships among discrete nodes in a graph or a network. In social network analysis, connections between individuals in a community or organization are analyzed, e.g., how information travels, or who has the most influence over whom.
- 14. Optimization.** A portfolio of numerical techniques used to redesign complex systems and processes to improve their performance according to one or more objective measures (e.g., cost, speed, or reliability).
- 15. Pattern recognition.** A set of machine learning techniques that assign some sort of output value (or label) to a given input value (or instance) according to a specific algorithm. Classification techniques are an example.
- 16. Predictive modeling.** A set of techniques in which a mathematical model is created or chosen to best predict the probability of an outcome. An example of an application in customer relationship management is the use of predictive models to estimate the likelihood that a customer will “churn” (i.e., change providers) or the likelihood that a customer can be cross-sold another product.
- 17. Regression.** A set of statistical techniques to determine how the value of the dependent variable changes when one or more independent variables is modified.
- 18. Sentiment analysis.** Application of natural language processing and other analytic techniques to identify and extract subjective information from source text material.
- 19. Signal processing.** A set of techniques from electrical engineering and applied mathematics originally developed to analyze discrete and continuous signals, i.e.,

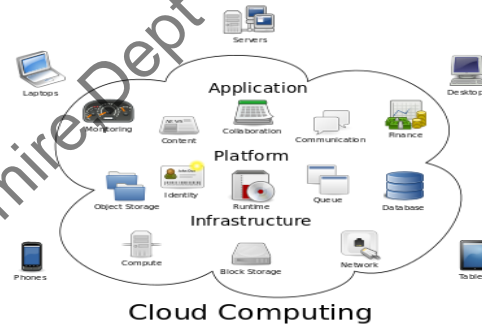
representations of analog physical quantities (even if represented digitally) such as radio signals, sounds, and images

- 20. Spatial analysis.** A set of techniques, some applied from statistics, which analyze the topological, geometric, or geographic properties encoded in a data set.
- 21. Statistics.** The science of the collection, organization, and interpretation of data, including the design of surveys and experiments.
- 22. Supervised learning.** The set of machine learning techniques that infer a function or relationship from a set of training data. Examples include classification and support vector machines
- 23. Simulation.** Modeling the behavior of complex systems, often used for forecasting, predicting and scenario planning. Monte Carlo simulations, for example, are a class of algorithms that rely on repeated random sampling, i.e., running thousands of simulations, each based on different assumptions.
- 24. Time series analysis.** Set of techniques from both statistics and signal processing for analyzing sequences of data points, representing values at successive times, to extract meaningful characteristics from the data. Examples of time series analysis include the hourly value of a stock market index or the number of patients diagnosed with a given condition every day.
- 25. Unsupervised learning.** A set of machine learning techniques that finds hidden structure in unlabeled data. Cluster analysis is an example of unsupervised learning (in contrast to supervised learning).
- 26. Visualization. Techniques** used for creating images, diagrams, or animations to communicate, understand, and improve the results of big data analyses

## 8.2. Cloud computing technologies and their types

- **Cloud computing**, also known as on-demand computing, is a kind of Internet-based computing, where shared resources, data and information are provided to computers and other devices on-demand. It is a model for enabling ubiquitous, on-demand access to a shared pool of configurable computing resources.

- Cloud computing and storage solutions provide users and enterprises with various capabilities to store and process their data in third-party data centers. It relies on sharing of resources, similar to a utility over a network.
- Cloud computing, or in simpler shorthand just "the cloud", also focuses on maximizing the effectiveness of the shared resources. Cloud resources are usually not only shared by multiple users but are also dynamically reallocated per demand.
- This can work for allocating resources to users. For example, a cloud computer facility that serves European users during European business hours with a specific application (e.g., email) may reallocate the same resources to serve North American users during North America's business hours with a different application (e.g., a web server).
- This approach helps maximize the use of computing power while reducing the overall cost of resources by using less power, air conditioning, rack space, etc. to maintain the system. With cloud computing, multiple users can access a single server to retrieve and update their data without purchasing licenses for different applications.



Cloud computing is usually described in one of two ways. Either based on the cloud location, or on the service that the cloud is offering. Based on a cloud location, we can classify cloud as: public,

- private,
- hybrid
- community cloud

Based on a service that the cloud is offering, we are speaking of either:

- IaaS (Infrastructure-as-a-Service)
- PaaS (Platform-as-a-Service)
- SaaS (Software-as-a-Service)
- or, Storage, Database, Information, Process, Application, Integration, Security, Management, Testing-as-a-service

**Public cloud**, we mean that the whole computing infrastructure is located on the premises of a cloud computing company that offers the cloud service. The location remains, thus, separate from the customer and he has no physical control over the infrastructure.

As public clouds use shared resources, they do excel mostly in performance, but are also most vulnerable to various attacks.

**Private cloud** means using a cloud infrastructure (network) solely by one customer/organization. It is not shared with others, yet it is remotely located. If the cloud is externally hosted. The companies have an option of choosing an on-premise private cloud as well, which is more expensive, but they do have a physical control over the infrastructure.

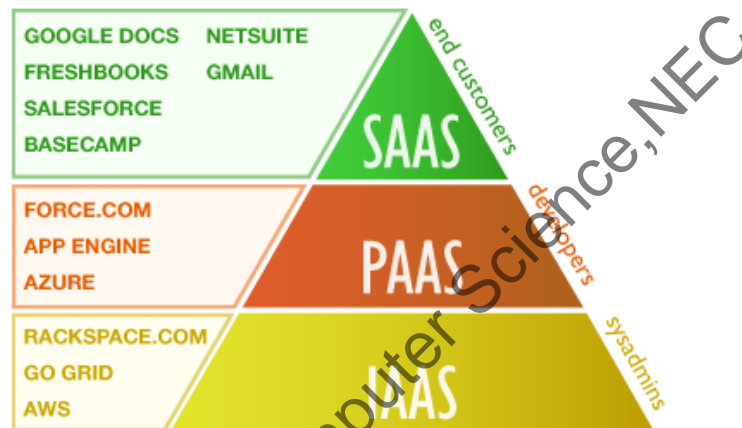
The security and control level is highest while using a private network. Yet, the cost reduction can be minimal, if the company needs to invest in an on-premise cloud infrastructure.

**Hybrid cloud**, of course, means, using both private and public clouds, depending on their purpose. For example, public cloud can be used to interact with customers, while keeping their data secured through a private cloud

Based upon the services offered, clouds are classified in the following ways:

1. **Infrastructure as a service (IaaS)** involves offering hardware related services using the principles of cloud computing. These could include some kind of storage services (database or disk storage) or virtual servers. Leading vendors that provide Infrastructure as a service are Amazon EC2, Amazon S3, Rackspace Cloud Servers and Flexi scale.
2. **Platform as a Service (PaaS)** involves offering a development platform on the cloud. Platforms provided by different vendors are typically not compatible. Typical players in PaaS are Google™s Application Engine, Microsoft Azure, force.com

3. **Software as a service (SaaS)** includes a complete software offering on the cloud. Users can access a software application hosted by the cloud vendor on pay-per-use basis. This is a well-established sector. The pioneer in this field has been Salesforce. coms offering in the online Customer Relationship Management (CRM) space. Other examples are online email providers like Googles gmail and Microsofts hotmail, Google docs and Microsofts online version of office called BPOS (Business Productivity Online Standard Suite)



### 8.3. Map Reduce and Hadoop systems

#### MAP REDUCE

- Map Reduce is a framework for processing parallelizable problems across huge datasets using a large number of computers (nodes), collectively referred to as a cluster (if all nodes are on the same local network and use similar hardware) or a grid (if the nodes are shared across geographically and administratively distributed systems, and use more heterogeneous hardware).
- Processing can occur on data stored either in a file system (unstructured) or in a database (structured). Map Reduce can take advantage of locality of data, processing it on or near the storage assets in order to reduce the distance over which it must be transmitted.
- Map Reduce is the heart of Hadoop. It is this programming paradigm that allows for massive scalability across hundreds or thousands of servers in a Hadoop cluster. The Map

Reduce concept is fairly simple to understand for those who are familiar with clustered scale-out data processing solutions.

- ✓ Map" step: Each worker node applies the "map ()" function to the local data, and writes the output to a temporary storage. A master node orchestrates that for redundant copies of input data, only one is processed.
- ✓ "Shuffle" step: Worker nodes redistribute data based on the output keys (produced by the "map ()" function), such that all data belonging to one key is located on the same worker node.
- ✓ "Reduce" step: Worker nodes now process each group of output data, per key, in parallel.
- Map Reduce allows for distributed processing of the map and reduction operations. Provided that each mapping operation is independent of the others, all maps can be performed in parallel – though in practice this is limited by the number of independent data sources and/or the number of CPUs near each source.
- Similarly, a set of 'reducers' can perform the reduction phase, provided that all outputs of the map operation that share the same key are presented to the same reducer at the same time, or that the reduction function is associative. While this process can often appear inefficient compared to algorithms that are more sequential, Map Reduce can be applied to significantly larger datasets than "commodity" servers can handle – a large server farm can use Map Reduce to sort a petabyte of data in only a few hours.
- The parallelism also offers some possibility of recovering from partial failure of servers or storage during the operation: if one mapper or reducer fails, the work can be rescheduled – assuming the input data is still available.

### **HADOOP MAP REDUCE**

- Open source project written in Java
- Large scale distributed data processing
- Based on Google's Map Reduce framework and Google file system
- Works on commodity hardware
- Used by a Google, Yahoo, Facebook, Amazon, and many other startups

<http://wiki.apache.org/hadoop/PoweredBy>

## HADOOP CORE

- Hadoop Distributed File System (HDFS)
  - Distributes and stores data across a cluster (brief intro only)
- Hadoop Map Reduce (MR)
  - Provides a parallel programming model
  - Moves computation to where the data is
  - Handles scheduling, fault tolerance
  - Status reporting and monitoring

```
function map(String name, String document):
```

```
  // name: document name
```

```
  // document: document contents
```

```
  for each word w in document:
```

```
    emit (w, 1)
```

```
function reduce(String word, Iterator partialCounts):
```

```
  // word: a word
```

```
  // partialCounts: a list of aggregated partial counts
```

```
  sum = 0
```

```
  for each pc in partialCounts:
```

```
    sum += pc
```

```
  emit (word, sum)
```

Here, each document is split into words, and each word is counted by the map function, using the word as the result key. The framework puts together all the pairs with the same key and feeds them to the same call to reduce. Thus, this function just needs to sum all of its input values to find the total appearances of that word.

## HADOOP SYSTEMS



- This open source software platform managed by the Apache Software Foundation has proven to be very helpful in storing and managing vast amounts of data cheaply and efficiently.
- Hadoop, and what makes it so special? Basically, it's a way of storing enormous data sets across distributed clusters of servers and then running "distributed" analysis applications in each cluster.
- It's designed to be robust, in that your Big Data applications will continue to run even when individual servers — or clusters — fail. And it's also designed to be efficient, because it doesn't require your applications to shuttle huge volumes of data across your network.

**Apache Hadoop** is an open-source software framework written in Java for distributed storage and distributed processing of very large data sets on computer clusters built from commodity hardware. All the modules in Hadoop are designed with a fundamental assumption that hardware failures are common and should be automatically handled by the framework.

- The core of Apache Hadoop consists of a storage part, known as Hadoop Distributed File System (HDFS), and a processing part called Map Reduce. Hadoop splits files into large blocks and distributes them across nodes in a cluster.
- To process data, Hadoop transfers packaged code for nodes to process in parallel based on the data that needs to be processed. This approach takes advantage of data locality— nodes manipulating the data they have access to— to allow the dataset to be processed faster and more efficiently than it would be in a more conventional supercomputer architecture that relies on a parallel file system where computation and data are distributed via high-speed networking.

The base Apache Hadoop framework is composed of the following modules:

Hadoop Common – contains libraries and utilities needed by other Hadoop modules;

- Hadoop Distributed File System (HDFS) – a distributed file-system that stores data on commodity machines, providing very high aggregate bandwidth across the cluster;
- Hadoop YARN – a resource-management platform responsible for managing computing resources in clusters and using them for scheduling of users' applications, and

- Hadoop Map Reduce – an implementation of the Map Reduce programming model for large scale data processing.

The term Hadoop has come to refer not just to the base modules above, but also to the ecosystem or collection of additional software packages that can be installed on top of or alongside Hadoop, such as Apache Pig, Apache Hive, Apache HBase, Apache Phoenix, Apache Spark, Apache ZooKeeper, Cloudera Impala, Apache Flume, Apache Sqoop, Apache Oozie, Apache Storm

Apache Hadoop's MapReduce and HDFS components were inspired by Google papers on their MapReduce and Google File System.

## MAP REDUCE ADVANTAGES

### Locality

Job tracker divides tasks based on location of data: it tries to schedule map tasks on same machine that has the physical data

### Parallelism

Map tasks run in parallel working different input data splits

Reduce tasks run in parallel working on different intermediate keys

Reduce tasks wait until all map tasks are finished

### Fault tolerance






Job tracker maintains a heartbeat with task trackers

Failures are handled by re-execution

If a task tracker node fails then all tasks scheduled on it (completed or incomplete) are re-executed on another node

## 8.4. DATA MANAGEMENT IN THE CLOUD

### Cloud Characteristics:

-  On-demand capabilities:
-  Broad network access
-  Resource pooling
-  Rapid elasticity
-  Measured service

Negative part of cloud characteristics

- ✚ Data is stored at an untrusted host.
- ✚ Data is replicated, often across large geographic distances
- ✚ Only parallelizable workload can utilize elastic computer power

### **Data management applications in the cloud**

- Two largest components of data management market:
  - Transactional Data Management
  - Analytical Data Management
- Which one will benefit from moving to the cloud?

#### **Transactional Data Management**

- Banks, airline reservation, online e-commerce
- ACID, write-intensive
- Not ready to move to the cloud for the following reasons:
  - Don't use shared-nothing architecture
  - Hard to maintain ACID when data replication are all over the world
  - Enormous risks in storing transactional data on an untrusted host

#### **Analytical Data Management**

- Business planning, decision support
- Well-suited to run in a cloud environment:
  - Shared-nothing architecture is a good match
  - ACID guarantees are typically not needed
  - Particularly sensitive data can be left out of the cloud.

### **8.5. Information Retrieval in the Cloud**

**It is based on IR algorithm and it consists of:**

- 1) The algorithmic rule takes the quantity of Search requests as input.
- 2) The algorithmic rule then breaks the Search requests into range of chunks needed for the knowledge retrieval from the general public cloud.

3) Based on the 2 assumptions, the algorithmic rules will the mapping per formalities and determines the quantity of buckets needed to perform the scale back function of the algorithm

### Various Techniques used for information retrieval and they are

#### 1) MAP REDUCE MECHANISM

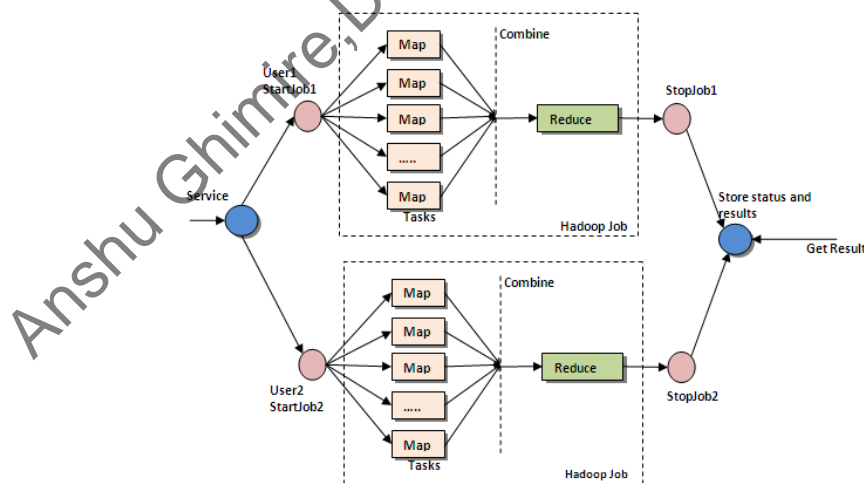
The thought of Map Reduce was introduced by Google in 2004 and is that the backbone of the many larger knowledge computations. Map Reduce is basically a divide and conquer algorithmic rule that breaks down the matter in to little parts and process it in parallel to accomplish economical computation on a bigger knowledge set.

The Map Reduce mechanism includes steps

1. Map
2. Reduce

In **Map step**, the most node acquires the input, partitions it up into smaller sub-problems, and distributes them to knowledge nodes, a knowledge node could try this over successively, resulting in a multi-level tree structure. The information node processes the smaller drawback, and passes the response back to its main node.

**Reduce:** In scale back step, the most node then collects the responses to all or any the sub-problems and merges them in several ways to stipulate the output – the reply to the matter it absolutely was at first attempting to resolve.



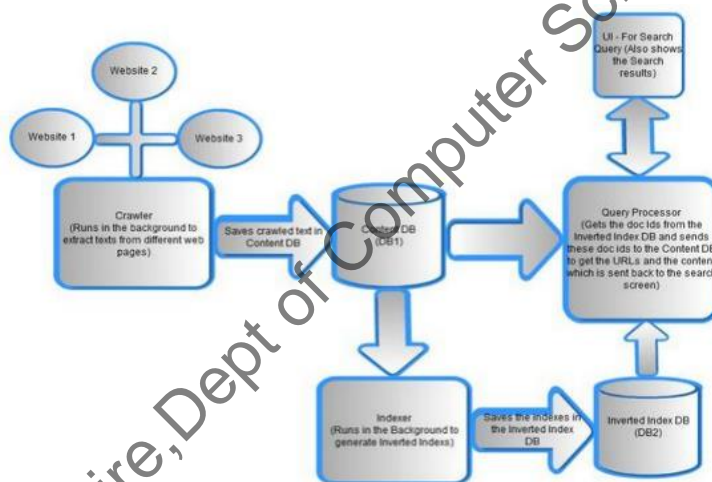
Map Reduce structure

## 2) CLOUD COMPUTING DESIGN

The cloud computing design used for the experiment includes 3 differing types of servers, namely:

1) Main Server 2) Secondary Server 3) Database Server

The cloud design has each master nodes and slave nodes. During this enactment, a main server is one that gets shopper requests and handles them. The master node is gift in main server and also the slave nodes in secondary server. Search requests area unit forwarded to the Map Reduce algorithmic rule gift in main server. Map Reduce takes care of the looking out and compartmentalization procedure by instigating an oversized range of Map and scale back processes. Once the Map Reduce method for a specific search key's completed, it returns the output worth to the most server and successively to the shopper. The entire design is portrayed in Figure two.



Implementation of data Retrieval (IR) algorithmic rule during a Cloud computing atmosphere

## 8.6. Link Analysis in Cloud Setup

Computer-based link analysis is a set of techniques for exploring associations among large numbers of objects of different types. These methods have proven crucial in assisting human investigators in comprehending complex webs of evidence and drawing conclusions that are not apparent from any single piece of information.

These methods are equally useful for creating variables that can be combined with structured data sources to improve automated decision-making processes. Typically, linkage data is modeled as a

graph, with nodes representing entities of interest and links representing relationships or transactions. Links and nodes may have attributes specific to the domain.

For example, link attributes might indicate the certainty or strength of a relationship, the dollar value of a transaction, or the probability of an infection.

Some linkage data, such as telephone call detail records, may be simple but voluminous, with uniform node and link types and a great deal of regularity.

Other data, such as law enforcement data, may be extremely rich and varied, though sparse, with elements possessing many attributes and confidence values that may change over time.

Various techniques are appropriate for distinct problems. For example, heuristic, localized methods might be appropriate for matching known patterns to a network of financial transactions in a criminal investigation. Efficient global search strategies, on the other hand, might be best for finding centrality or severability in a telephone network.

Link analysis can be broken down into two components—link generation, and utilization of the resulting linkage graph.

### **Link Generation**

Link generation is the process of computing the links, link attributes and node attributes. There are several different ways to define links. The different approaches yield very different linkage graphs. A key aspect in defining a link analysis is deciding which representation to use.

### **Explicit Links**

A link may be created between the nodes corresponding to each pair of entities in a transaction. For example, with a call detail record, a link is created between the originating telephone number and the destination telephone number. This is referred to as an explicit link.

### **Aggregate Links**

A single link may be created from multiple transactions. For example, a single link could represent all telephone calls between two parties, and a link attribute might be the number of calls represented. Thus, several explicit links may be collapsed into a single aggregate link.

### **Inferred Relationships**

Links may also be created between pairs of nodes based on inferred strengths of relationships between them

### **Utilization**

Once a linkage graph, including the link and node attributes, has been defined, it can be browsed, searched or used to create variables as inputs to a decision system.

### **Visualization**

In visualizing linking graphs, each node is represented as an icon, and each link is represented as a line or an arrow between two nodes. The node and link attributes may be displayed next to the items or accessed via mouse actions. Different icon types represent different entity types. Similarly, link attributes determine the link representation (line strength, line color, arrowhead, etc.).

### **Variable Creation**

Link analysis can append new fields to existing records or create entirely new data sets for subsequent modeling stages in a decision system. For example, a new variable for a customer might be the total number of email addresses and credit card numbers linked to that customer.

### **Search**

Link analysis query mechanisms include retrieving nodes and links matching specified criteria, such as node and link attributes, as well as search by example to find more nodes that are similar to the specified example node.

A more complex task is similarity search, also called clustering. Here, the objective is to find groups of similar nodes. These may actually be multiple instances of the same physical entity, such as a single individual using multiple accounts in a similar fashion.

### **Network Analysis**

Network analysis is the search for parts of the linkage graph that play particular roles. It is used to build more robust communication networks and to combat organized crime.

### **Applications**

Link analysis is increasingly used in law enforcement investigations, detecting terrorist threats, fraud detection, detecting money laundering, telecommunications network analysis, classifying web pages, analyzing transportation routes, pharmaceuticals research, epidemiology, detecting nuclear proliferation and a host of other specialized applications.

For example, in the case of money laundering, the entities might include people, bank accounts and businesses, and the transactions might include wire transfers, checks and cash deposits. Exploring relationships among these different objects helps expose networks of activity, both legal and illegal.

### **Strengths**

Link analysis often makes information accessible that is not apparent from any single data record.

### **Weaknesses**

Link analysis is as much an art as a science, and just configuring a link analysis can be a major