BIG DATA ANALYTICS

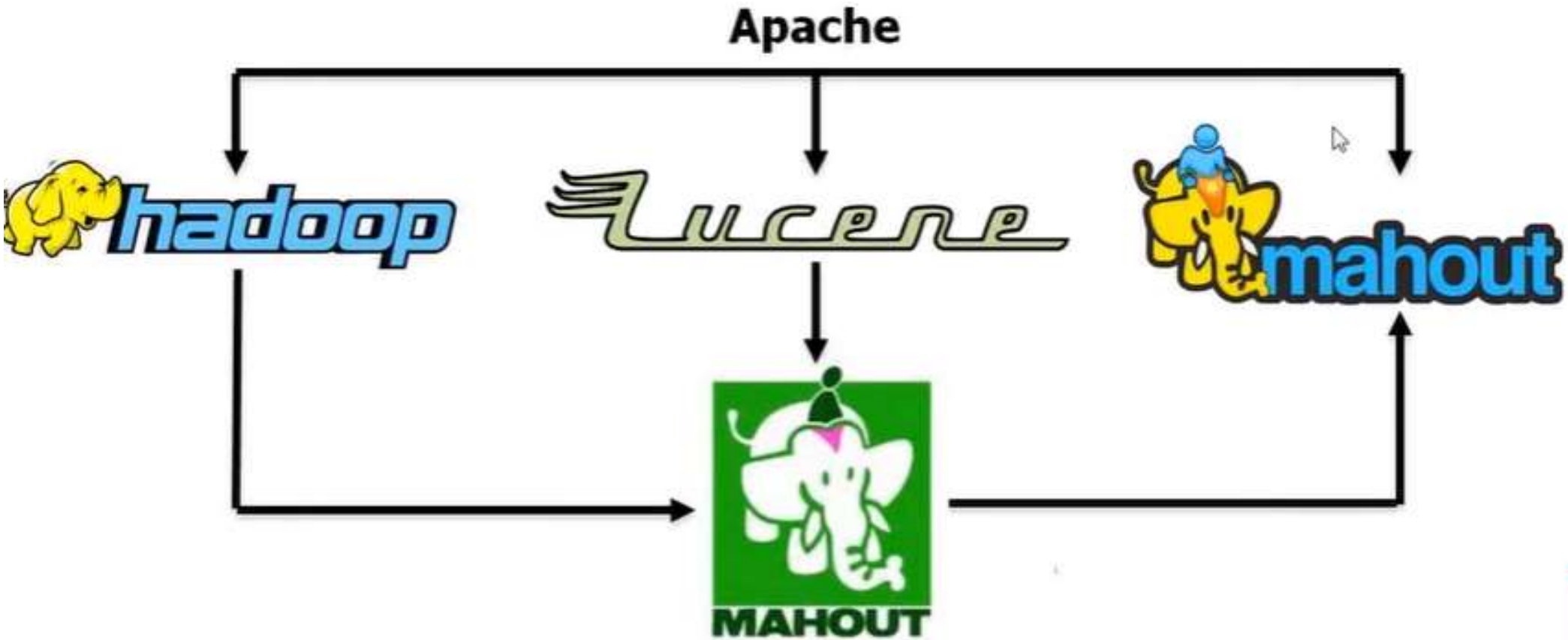
Part – 8

"MAHOUT"

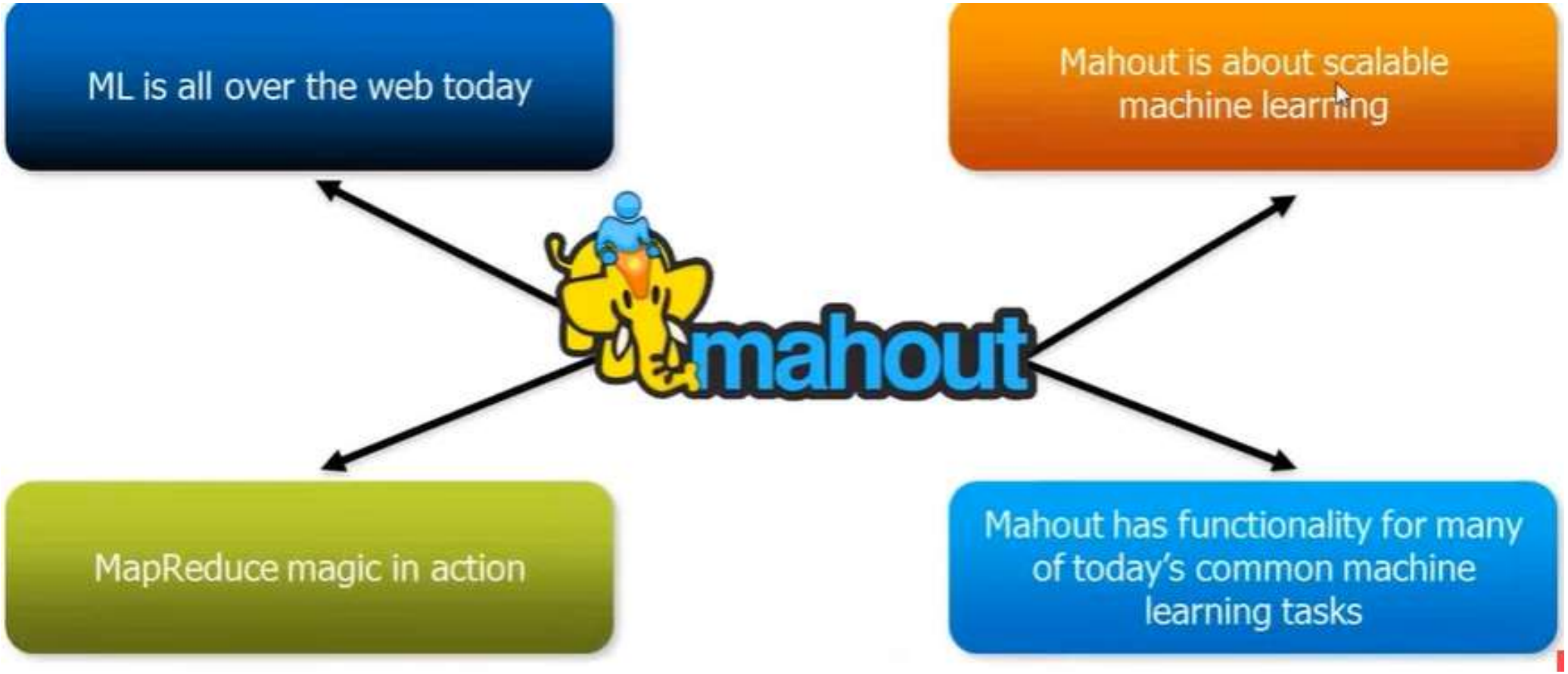
Mahout

- ✓ Mahout began life in 2008 as a subproject of Apache's Lucene project, which provides the well-known open source search engine of the same name.
- ✓ Lucene provides advanced implementations of search, text mining, and information-retrieval techniques.
- ✓ In the universe of computer science, these concepts are adjacent to machine learning techniques like clustering and, to an extent, classification.
- ✓ As a result, some of the work of the Lucene committers that fell more into these machine learning areas was spun off into its own subproject.
- ✓ Soon after, Mahout absorbed the Taste open source collaborative filtering project

Apache Mahout and its related projects within the Apache Software Foundation



Mahout

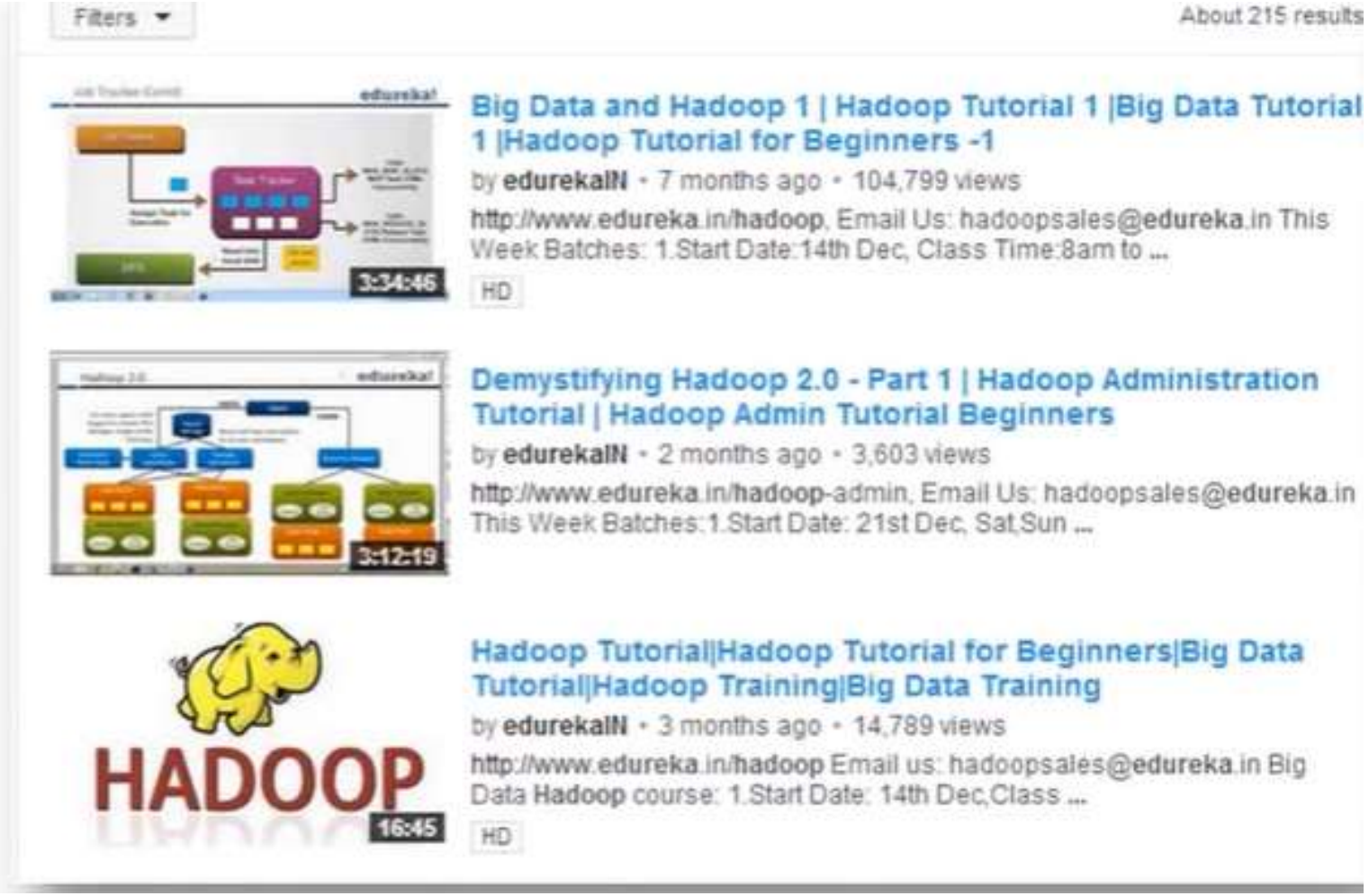

ML is all over the web today
Mahout is about scalable machine learning
mahout
MapReduce magic in action
Mahout has functionality for many of today's common machine learning tasks

Use Case:

YouTube utilizes recommendation systems to bring videos to a user that it believes the user will be interested in.

They are designed to:

- ✓ Increase the numbers of videos the user will watch
- ✓ Increase the length of time he spends on the site, and
- ✓ Maximize the enjoyment of his YouTube experience.



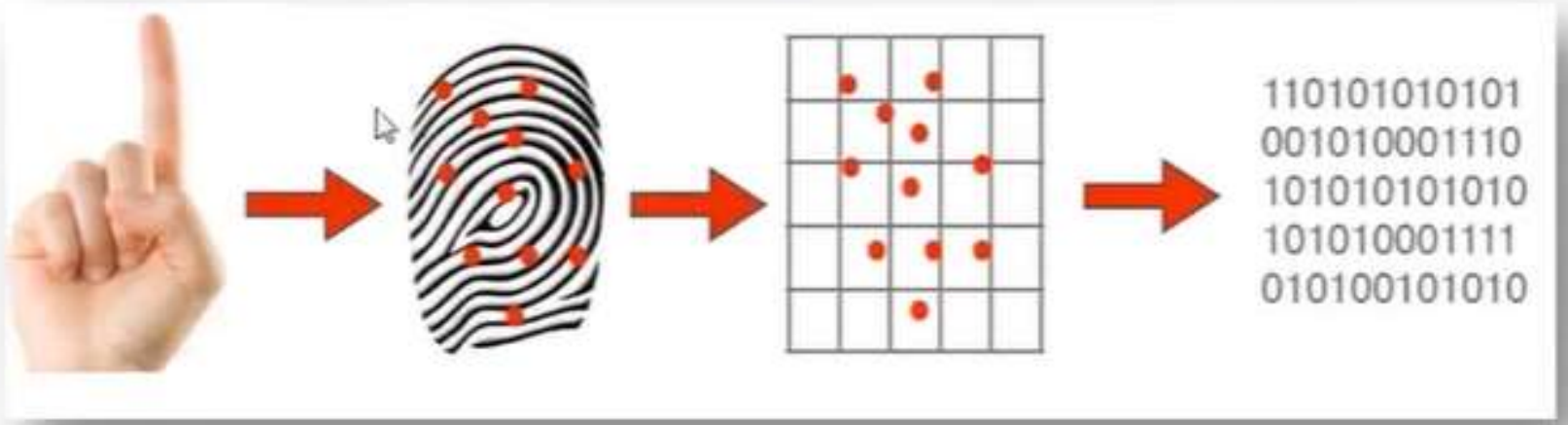
Use case: Biometric

Biometrics : The Science of establishing the identity of an individual based on the physical, chemical or behavioral attributes of the person.

Why is it Important?

- ✓ Identify Individual credentials
- ✓ Identify and prevent banking fraud
- ✓ Enforcement of law and security

Finger Print Scanner Work:



A fingerprint scanner system has two basic jobs

- ✓ Get an image of your finger
- ✓ Determine whether the pattern of ridges and valleys in this image matches the pattern of ridges and valleys in pre-scanned images

Process

- ✓ Only specific characteristics, which are unique to every fingerprint, are filtered are saved as an encrypted biometric key or mathematical representation.
- ✓ No image of a fingerprint is ever saved, only a series of numbers (a binary code), which is used for verification. The algorithm cannot be reconverted to an image, so no one can duplicate your fingerprints

Mahout:

- ✓ What is Learning?
- ✓ Can a Machine learn?
- ✓ How to do it ?

Mahout : Scalable Machine learning Library

Machine Learning is Programming Computers to optimize Performance Criterion using Example Data or Past experience.

- ✓ A branch of artificial intelligence
- ✓ Systems that learn from data
- ✓ Classify data after learning
- ✓ Learn on test data sets
- ✓ Generalisation – the ability to classify unseen data sets

Mahout : Perform

Collaborative Filtering

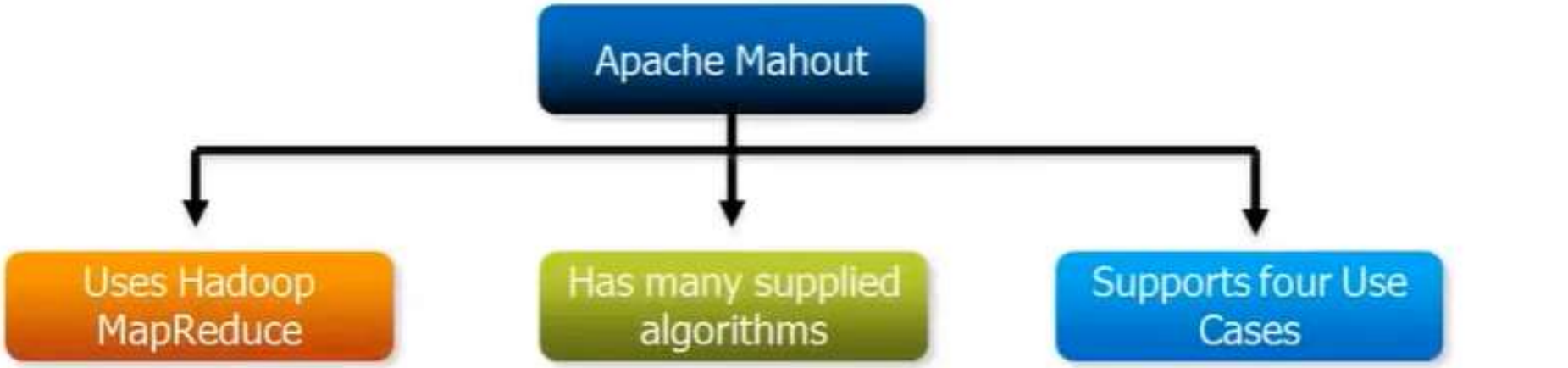
Clustering

- ✓ Apache Mahout is an Apache project to produce open source implementations of distributed or otherwise scalable machine learning algorithms focused primarily in the areas of collaborative filtering, clustering and classification, often leveraging, but not limited to, the Hadoop platform.
- ✓ The Apache Mahout project aims to make building intelligent applications easier and faster. Mahout co-founder Grant Ingersoll introduces the basic concepts of machine learning and then demonstrates how to use Mahout to cluster documents, make recommendations, and organize content.

Three specific machine-learning tasks that Mahout currently implements are:

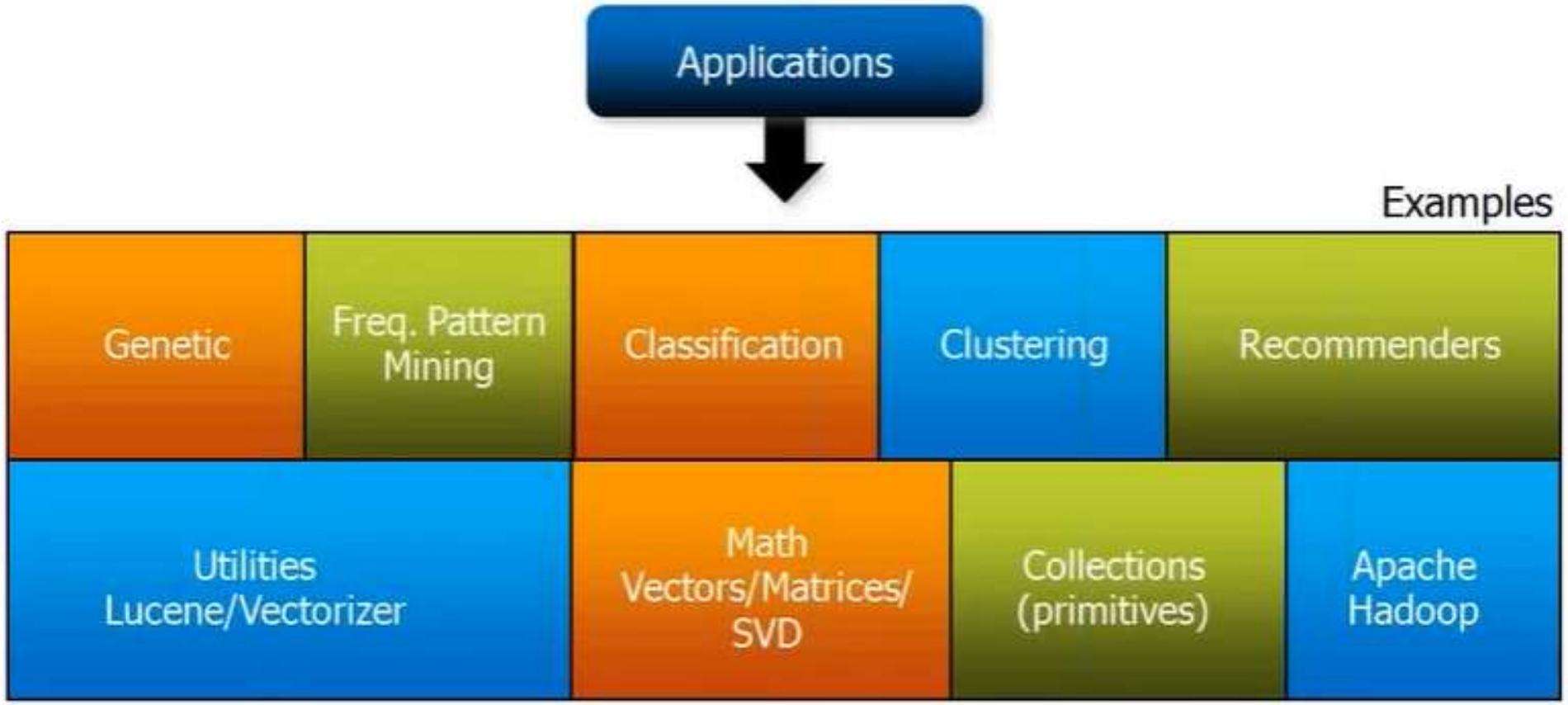
- ✓ Collaborative Filtering
- ✓ Clustering
- ✓ Classification

Mahout- How Does It work



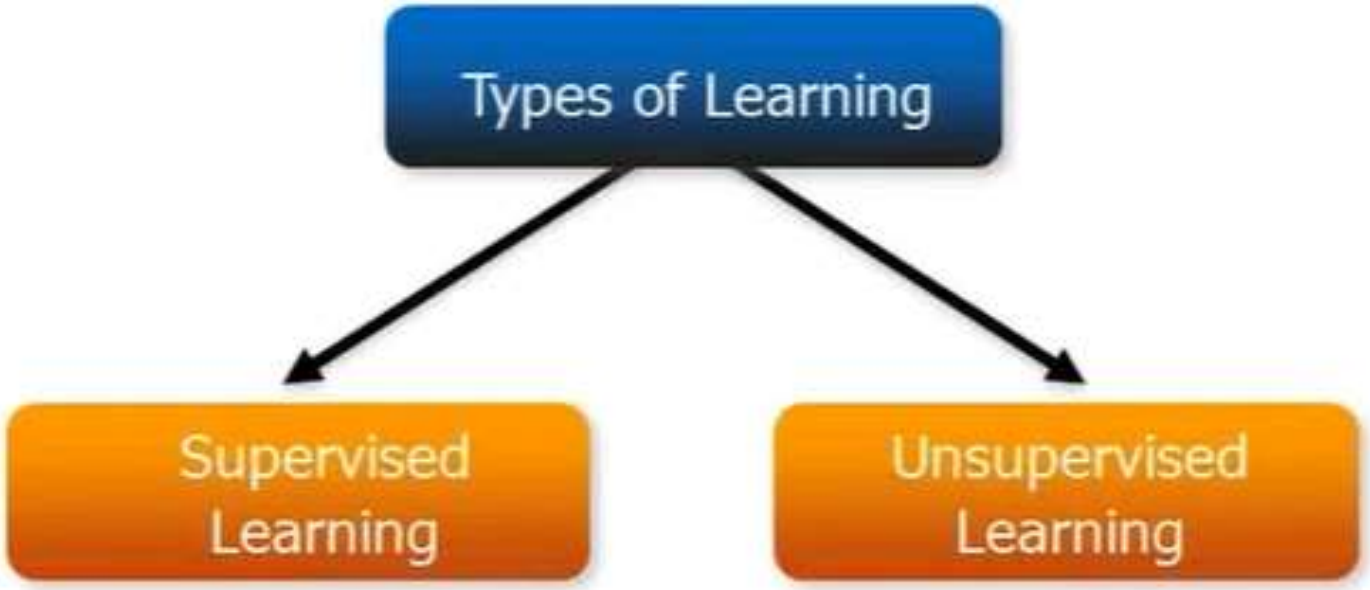
- ✓ Recommendation mining
- ✓ Clustering
- ✓ Classification
- ✓ Frequent Item set Mining

Mahout: Applications



Learning Technique: Types of Learning

Attain knowledge by study, experience, or by being taught.



Supervised Learning:

Supervised learning : Training data includes both the input and the desired results.

- ✓ For some examples, the correct results (targets) are known and are given in input to the model during the learning process.
- ✓ The construction of a proper training, validation and test set is crucial.
- ✓ These methods are usually fast and accurate.
- ✓ Have to be able to generalize: give the correct results when new data are given in input without knowing a priori the target.

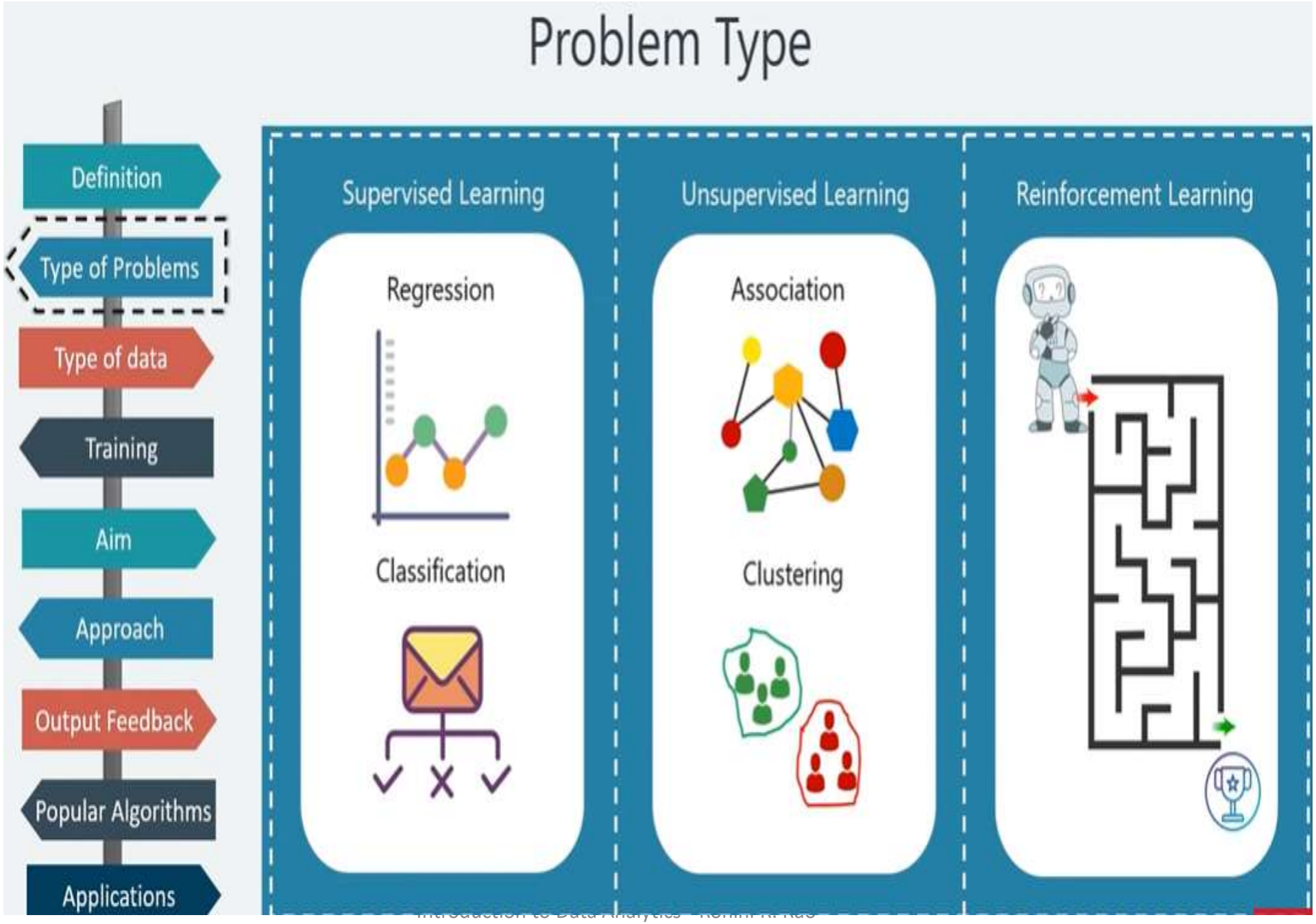
Unsupervised Learning:

Unsupervised Learning:

- ✓ The model is not provided with the correct results during the training.
- ✓ Can be used to cluster the input data in classes on the basis of their statistical properties only Cluster significance and labeling.
- ✓ The labeling can be carried out even if the labels are only available for a small number of objects representative of the desired classes.

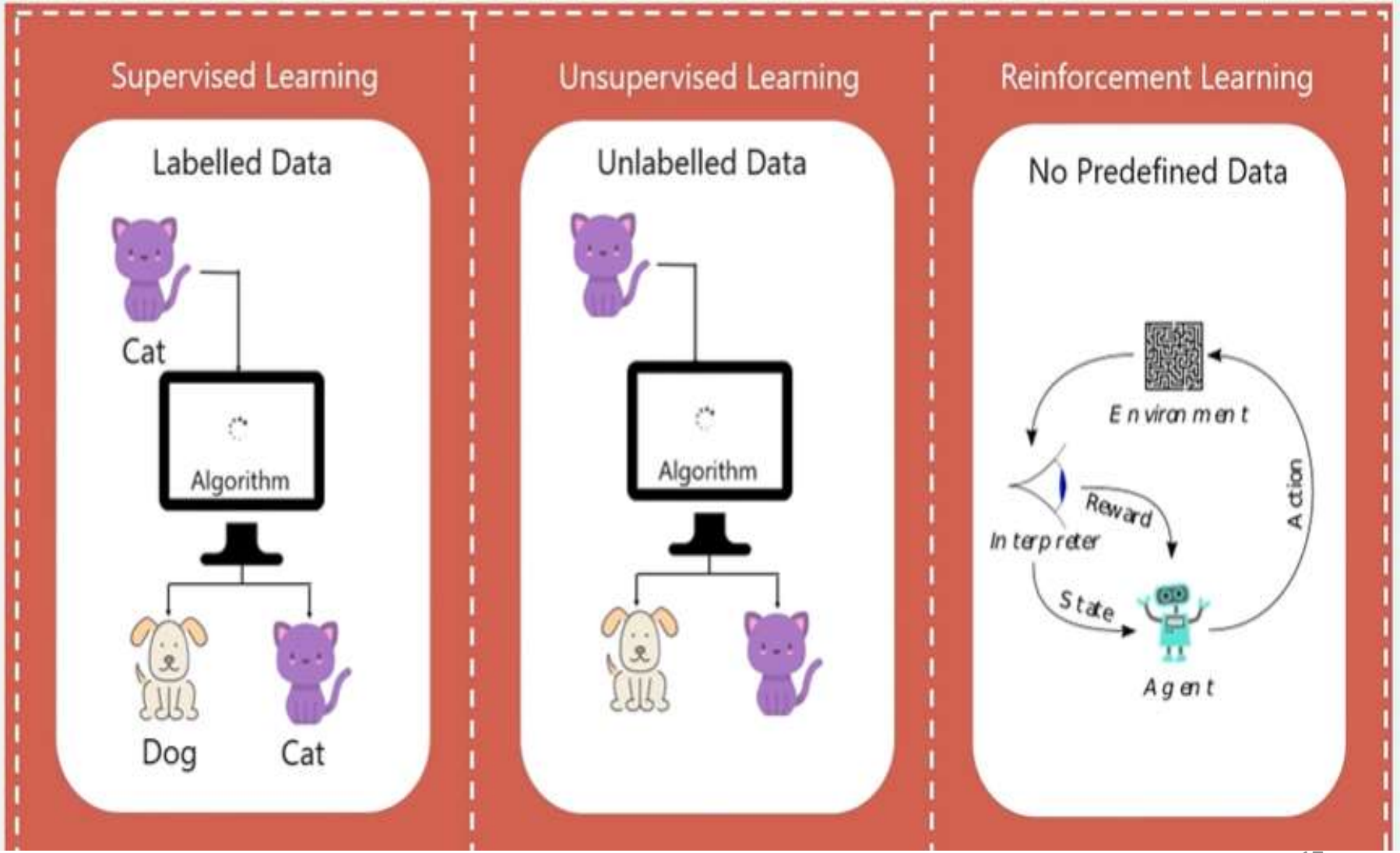
Supervised & Unsupervised Learning

Parameters	Supervised machine learning	Unsupervised machine learning
Input Data	Algorithms are trained using labeled data.	Algorithms are used against data that is not labeled
Computational Complexity	Simpler method	Computationally complex
Accuracy	Highly accurate	Less accurate
No. of classes	No. of classes is known	No. of classes is not known
Data Analysis	Uses offline analysis	Uses real-time analysis of data
Algorithms used	Linear and Logistics regression, Random forest, Support Vector Machine, Neural Network, etc.	K-Means clustering, Hierarchical clustering, Apriori algorithm, etc.


Problem Type
Definition
Type of Problems
Type of data
Training
Aim
Approach
Output Feedback
Popular Algorithms
Applications
Supervised Learning
Regression
Classification
Unsupervised Learning
Association
Clustering
Reinforcement Learning

Type of data

- Definition
- Type of Problems
- Type of data
- Training
- Aim
- Approach
- Output Feedback
- Popular Algorithms
- Applications



Training

- Definition
- Type of Problems
- Type of data
- Training
- Aim
- Approach
- Output Feedback
- Popular Algorithms
- Applications

Supervised Learning

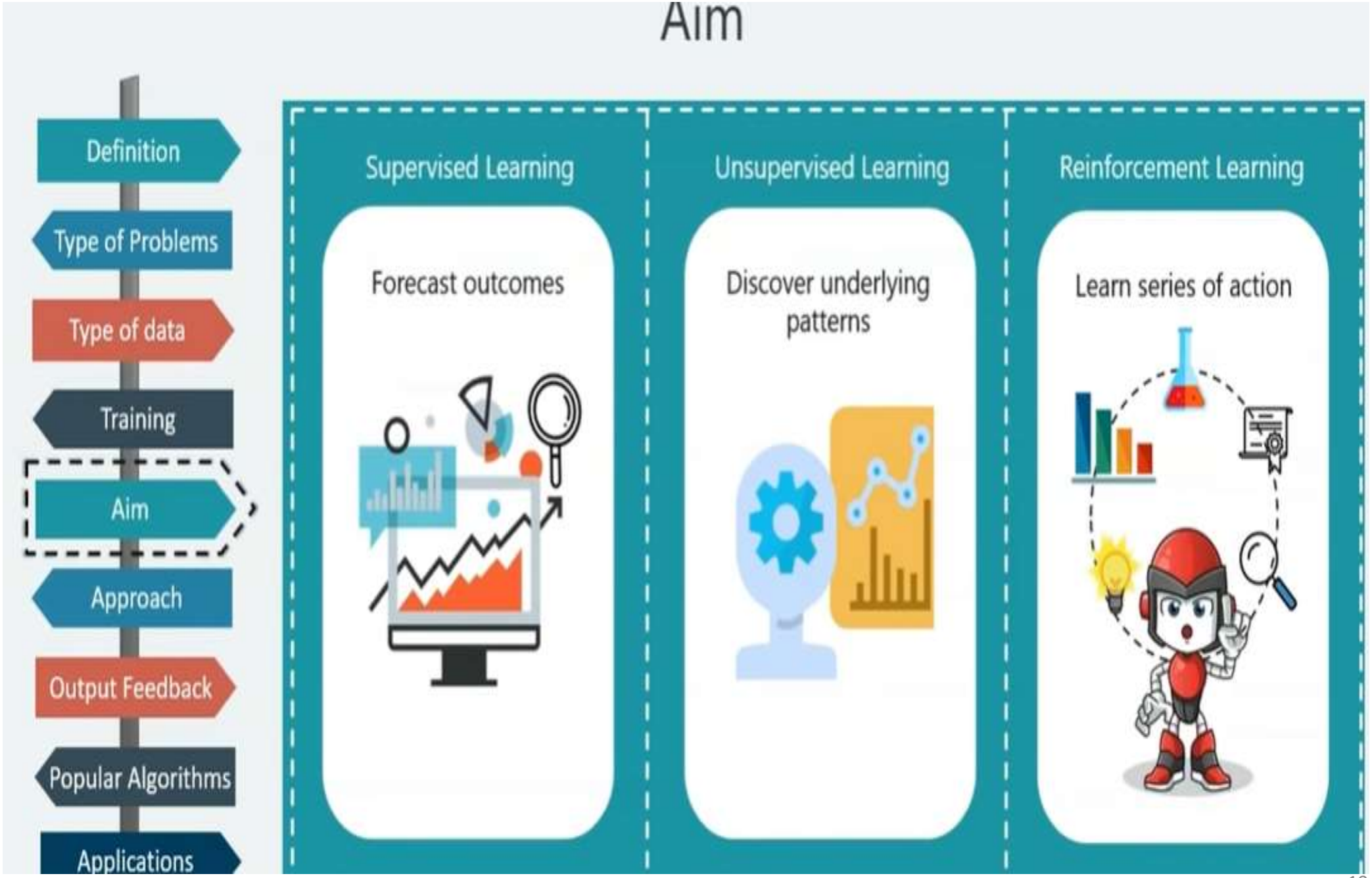
External supervision

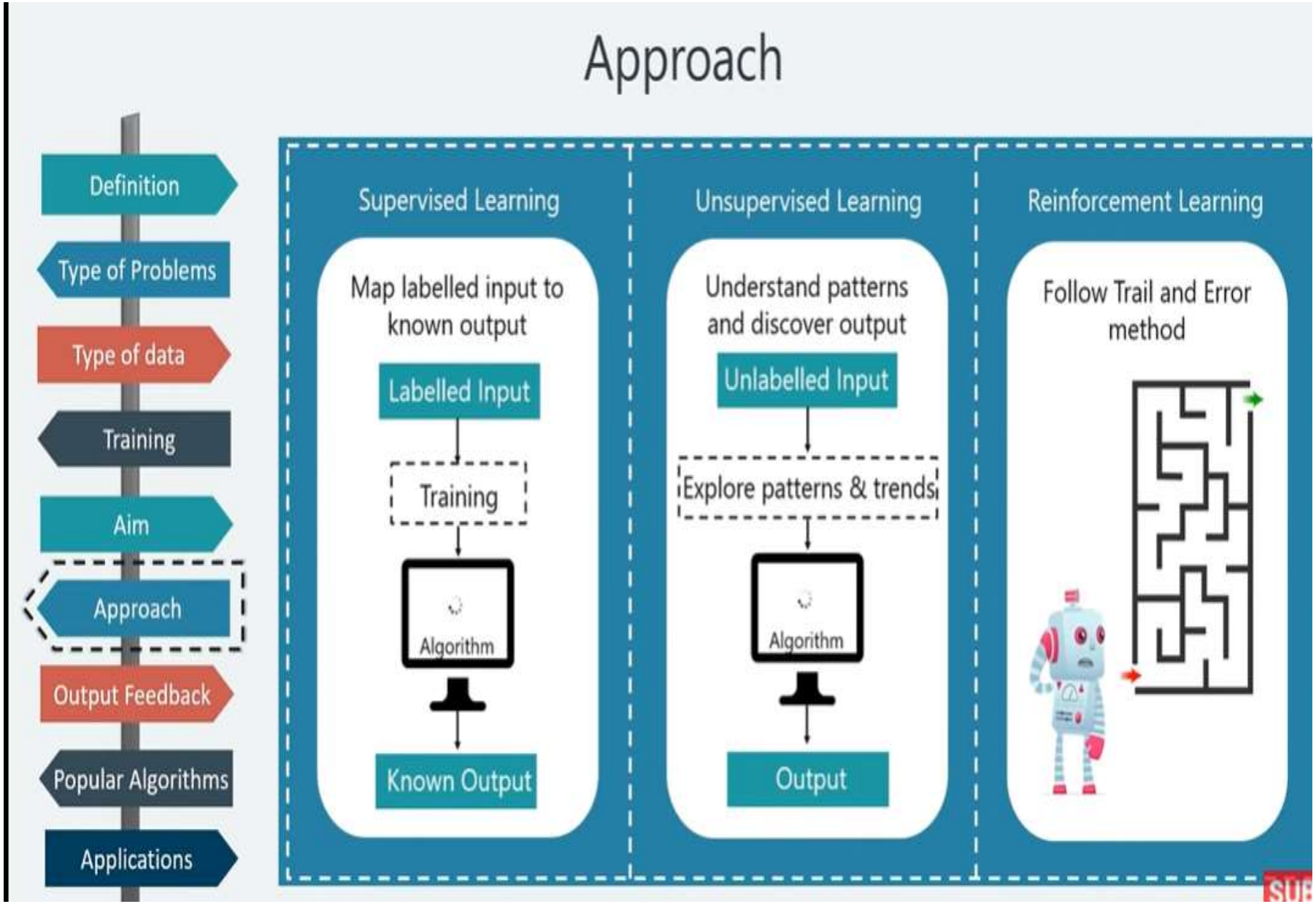
Unsupervised Learning

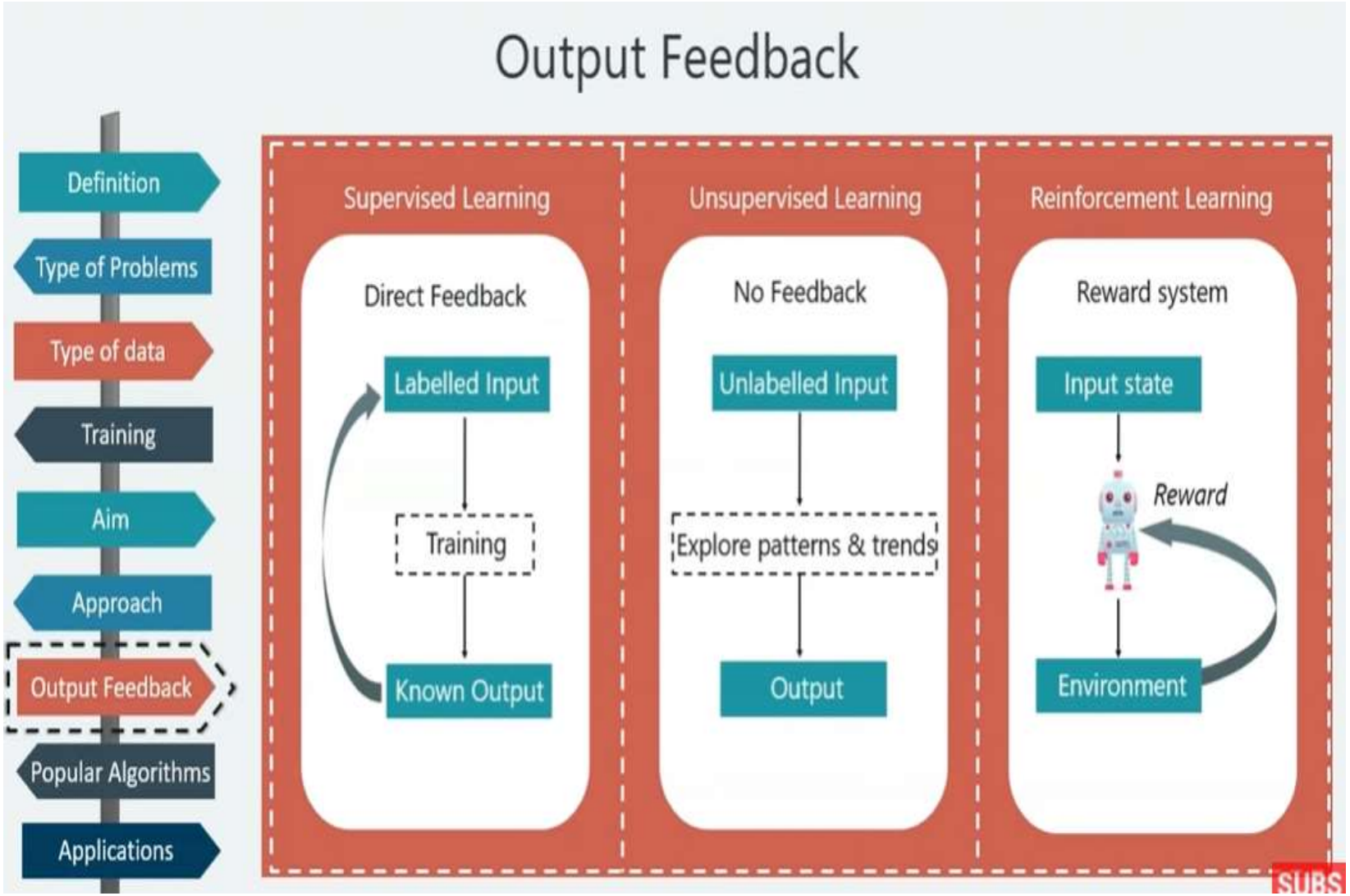
No supervision

Reinforcement Learning

No supervision

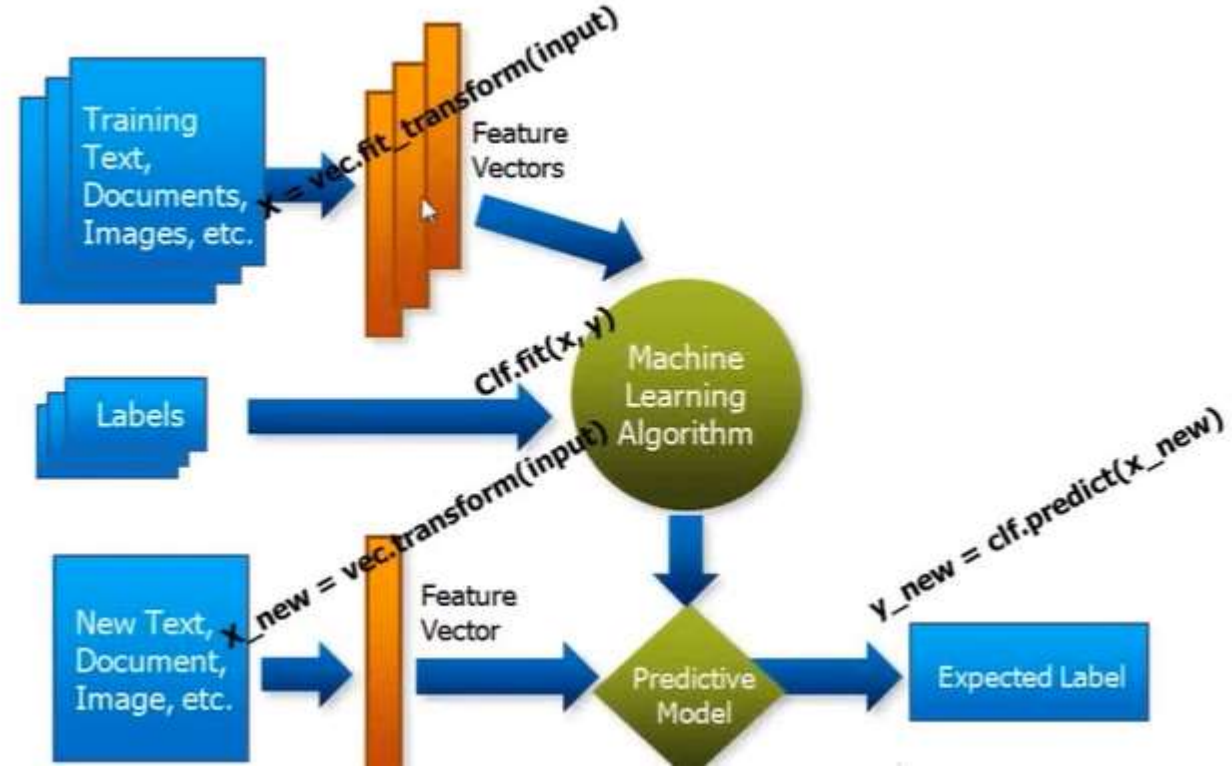

Aim
Definition
Type of Problems
Type of data
Training
Aim
Approach
Output Feedback
Popular Algorithms
Applications
Supervised Learning
Forecast outcomes
Unsupervised Learning
Discover underlying patterns
Reinforcement Learning
Learn series of action


Approach
Definition
Type of Problems
Type of data
Training
Aim
Approach
Output Feedback
Popular Algorithms
Applications
Supervised Learning
Map labelled input to known output
Labelled Input
Training
Algorithm
Known Output
Unsupervised Learning
Understand patterns and discover output
Unlabelled Input
Explore patterns & trends
Algorithm
Output
Reinforcement Learning
Follow Trail and Error method

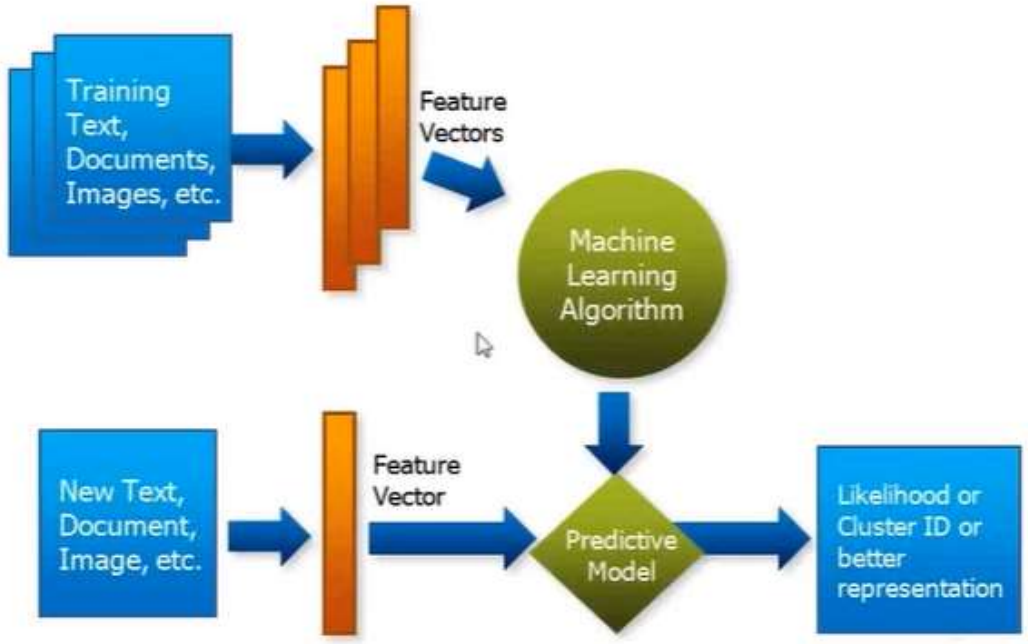

Output Feedback
Definition
Type of Problems
Type of data
Training
Aim
Approach
Output Feedback
Popular Algorithms
Applications
Supervised Learning
Direct Feedback
Labelled Input
Training
Known Output
Unsupervised Learning
No Feedback
Unlabelled Input
Explore patterns & trends
Output
Reinforcement Learning
Reward system
Input state
Reward
Environment
SUBS


Applications
Definition
Type of Problems
Type of data
Training
Aim
Approach
Output Feedback
Popular Algorithms
Applications
Supervised Learning
Risk Evaluation
Forecast Sales
Unsupervised Learning
Recommendation Systems
Anomaly Detection
Reinforcement Learning
Self driving cars
Gaming

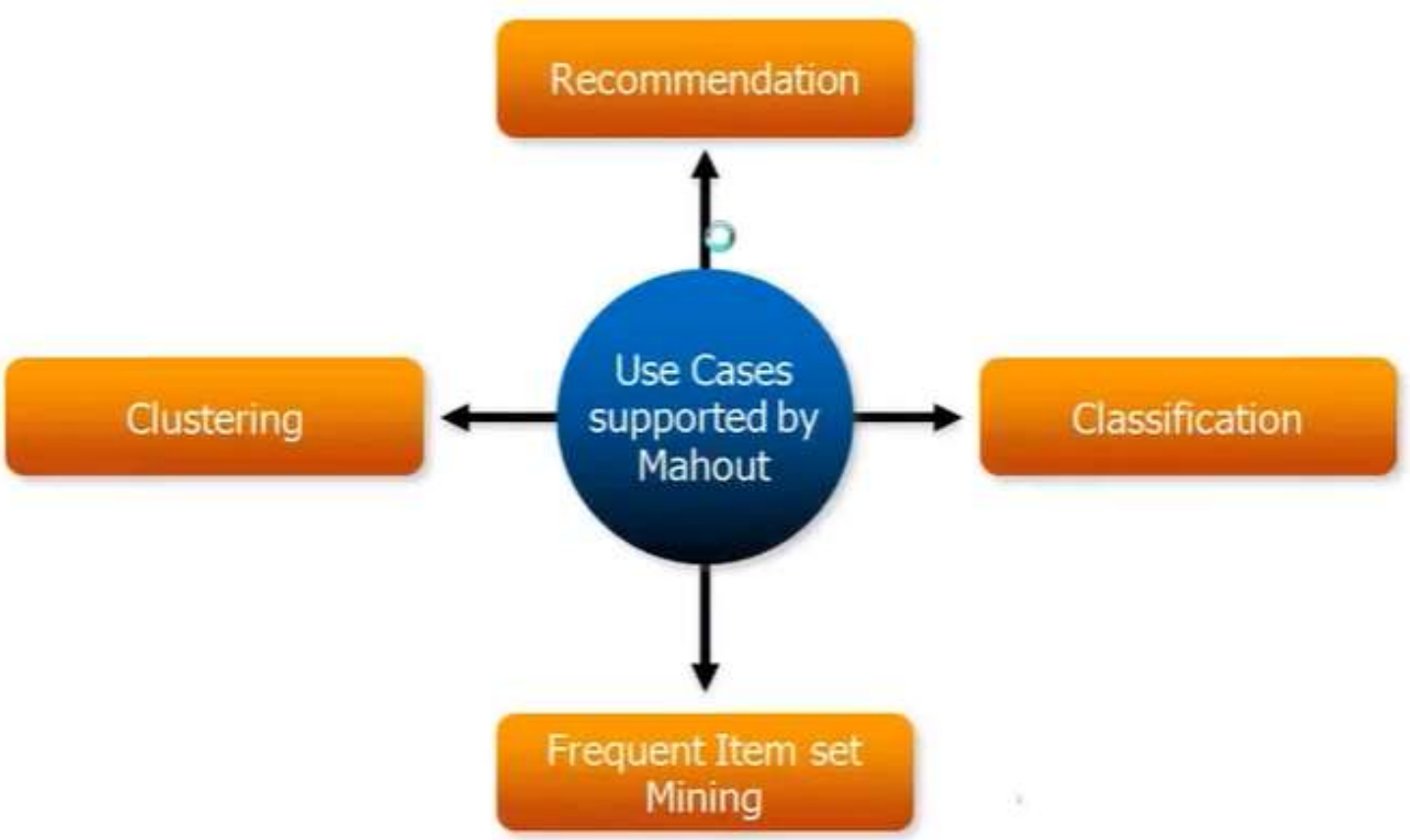
Supervised Learning:



Unsupervised Learning:



Applications Of Mahout:



Clustering:

Organizing data into *clusters* such that there is:

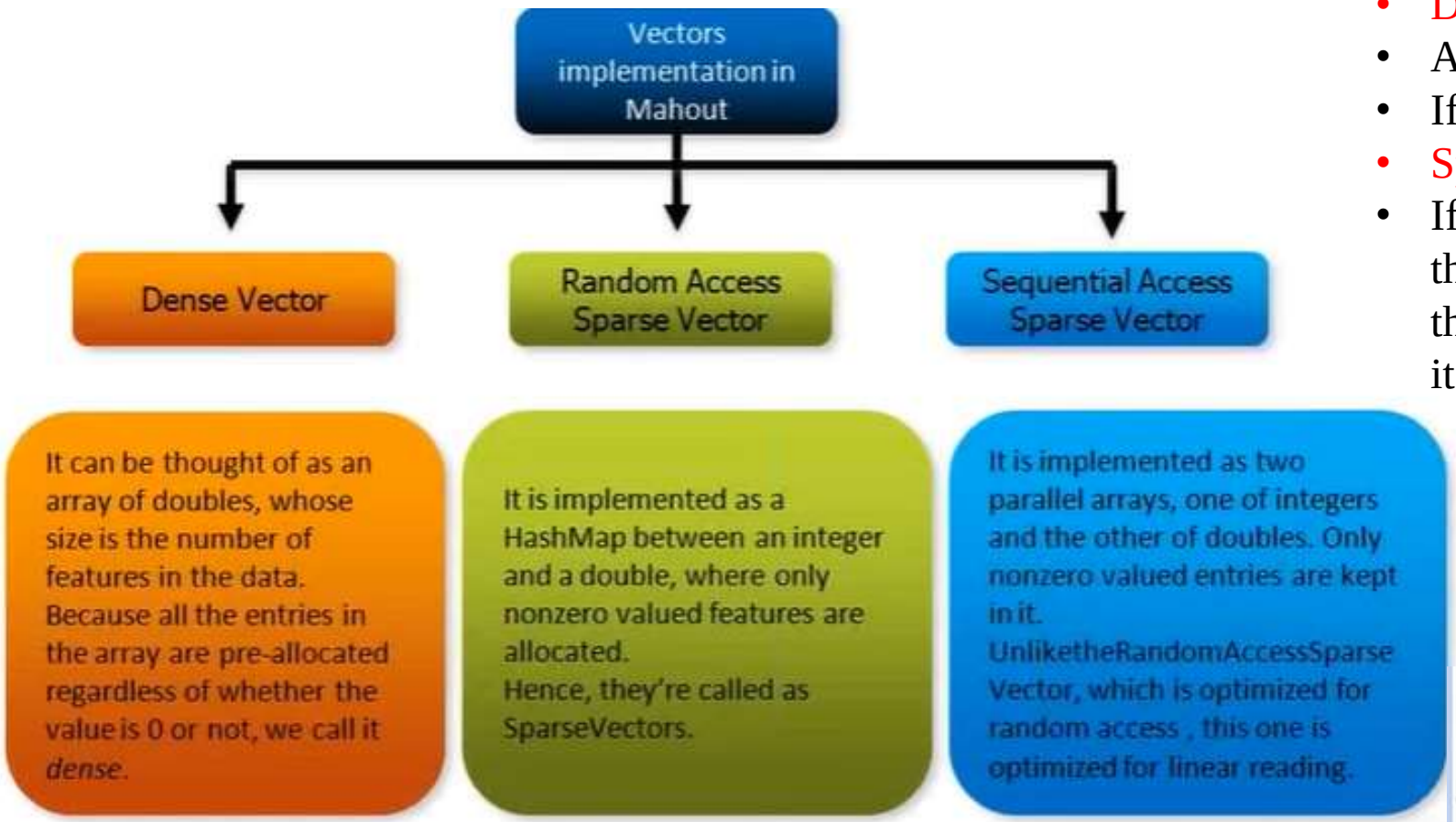
- ✓ High intra-cluster similarity
- ✓ Low inter-cluster similarity
- ✓ Informally, finding natural groupings among objects.

Why do we want to do it??

Why Clustering?

- ✓ Organizing data into clusters shows internal structure of the data
  Ex. Clusty and clustering genes
- ✓ Sometimes the partitioning is the goal
  Ex. Market segmentation
- ✓ Prepare for other AI techniques
  Ex. Summarize news (cluster and then find centroid)
- ✓ Techniques for clustering is useful in knowledge discovery in data
  Ex. Underlying rules, reoccurring patterns, topics, etc.

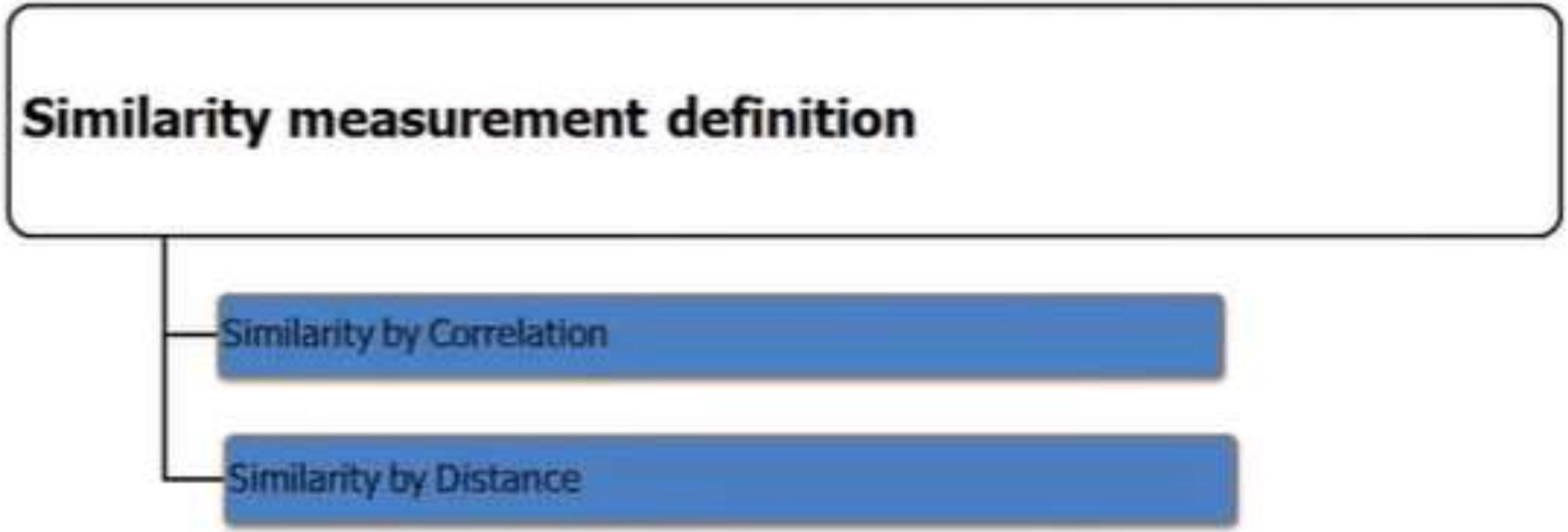
Vector Implementations:



- Dense: Assume 50 Dimension Vector.
- All values should exist.
- If some value is missing replace with 0.
- Sparse (Random):
- If any value is not allocated to specific feature then don't consider or ignore them or exclude that variable which has no value assigned to it..

```
{x, y, z} - {12.0, 0, 87.9} Dense vector

Sparse Vector{x:12.0, z:87.9}
Sequential Access sparseVector
[0,1,2]
[12.0, 34.5, 67.8]
```


Similarity measurement definition
Similarity by Correlation
Similarity by Distance

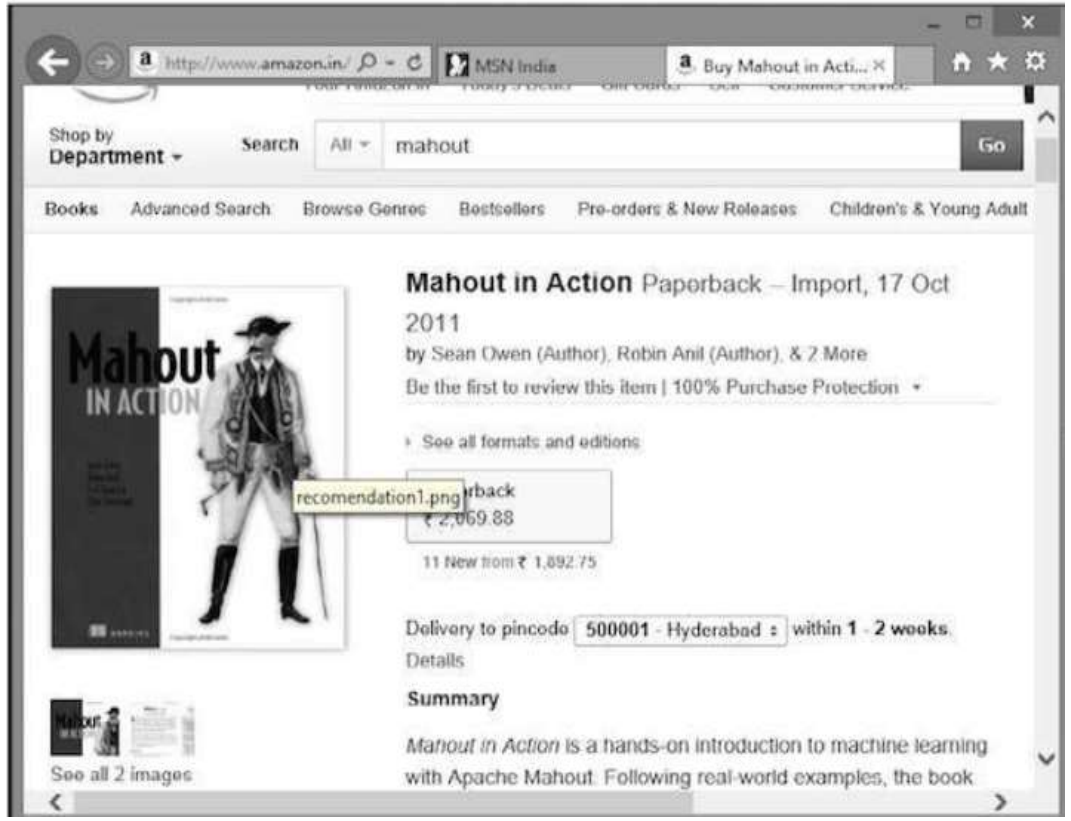
Similarity by distance

- Euclidean distance measure
- Manhatten distance measure
- Cosine distance measure
- Tanimoto distance measure
- Squared Euclidean distance measure

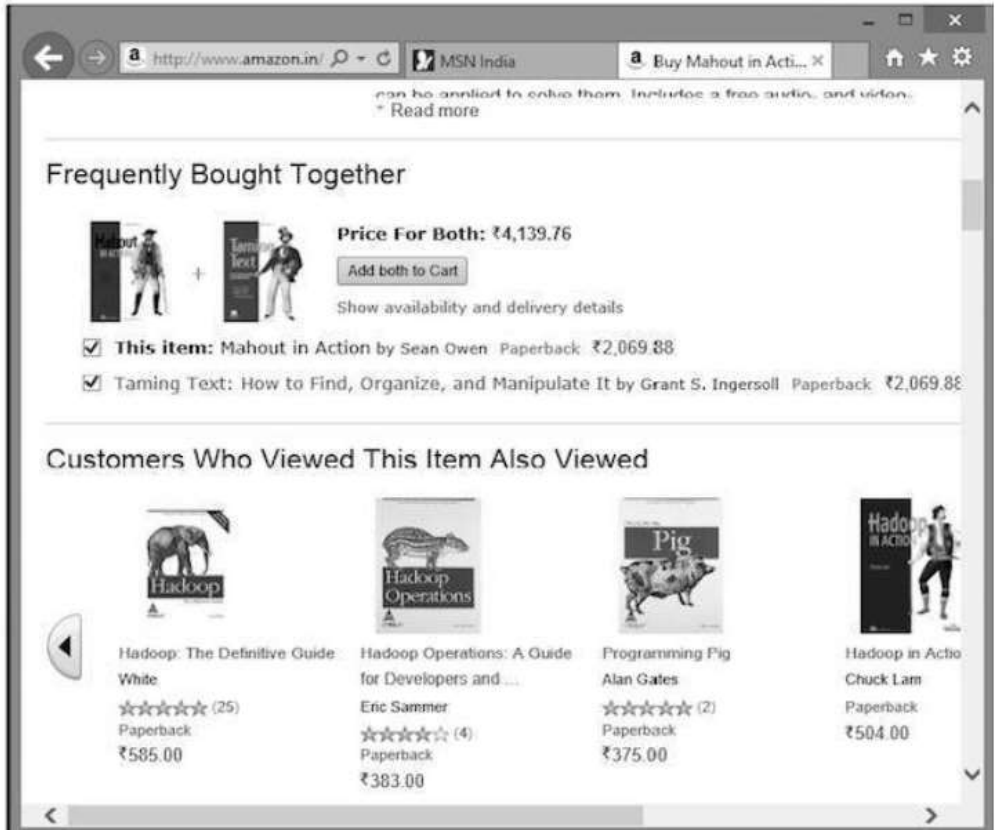
Recommendation or Collaborative Filtering

A Mahout-based collaborative filtering engine **takes users' preferences for items** ("tastes") and **returns estimated preferences for other items**.

- Suppose you want to purchase the book "**Mahout in Action**" from Amazon:



- Along with the **selected product**, Amazon also **displays a list of related recommended items**, as shown below.

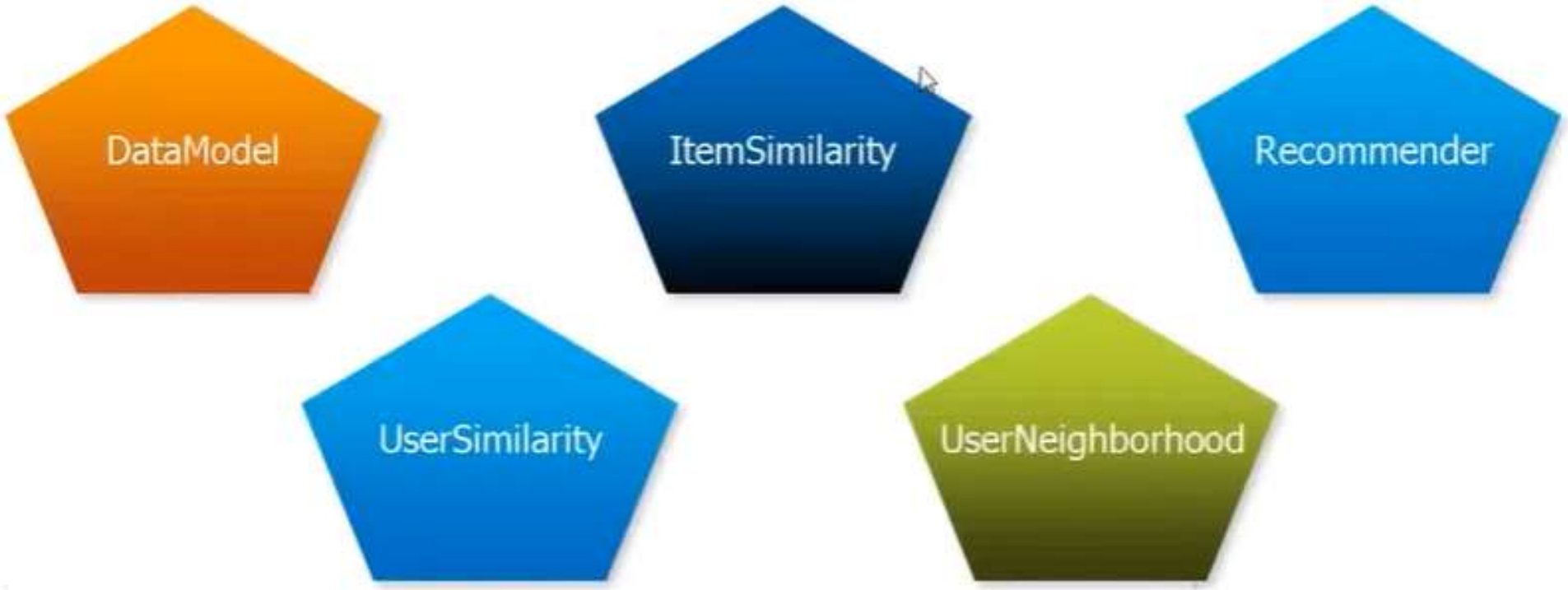


- Such recommendation lists are produced with the help of **recommender engines**. Mahout provides recommender engines of several types such as: user-based , item-based

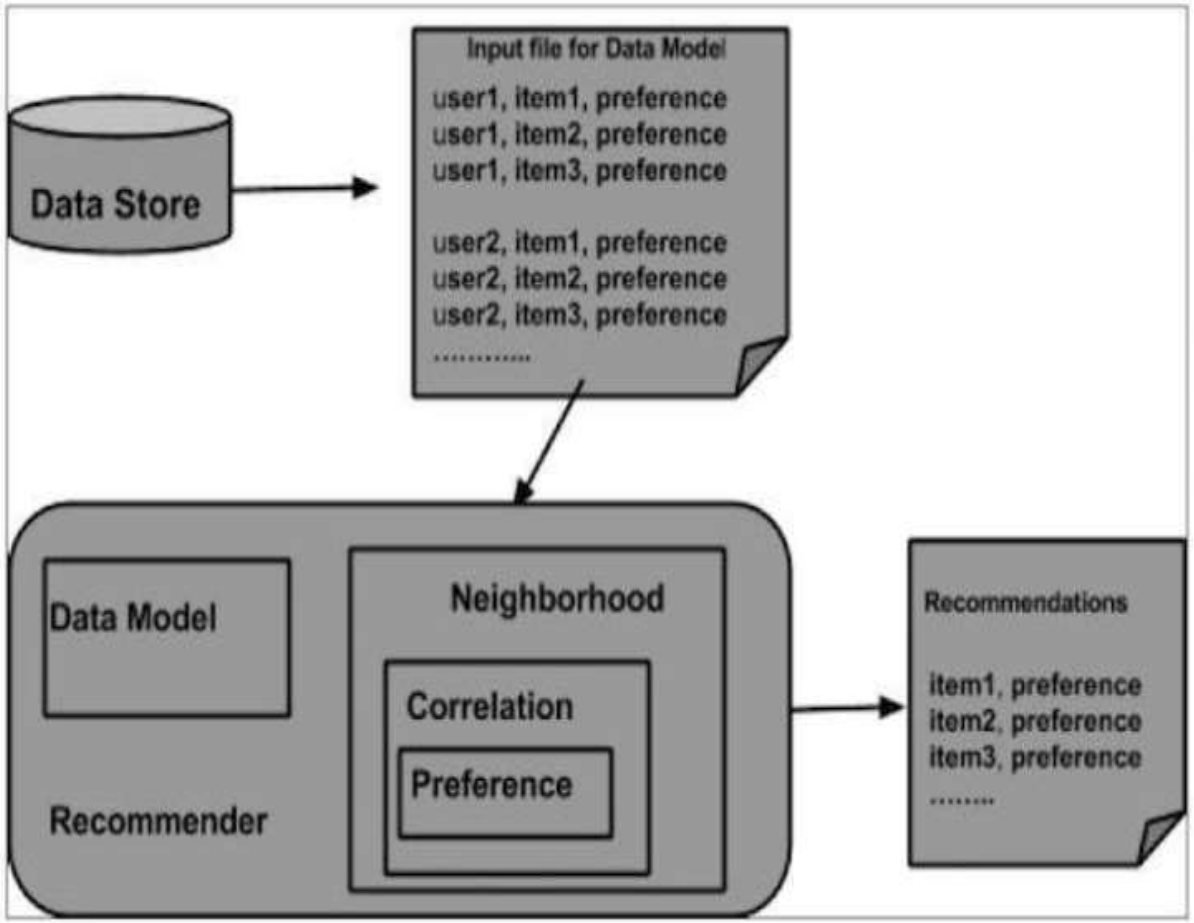
Mahout Packages:

- The components provided by Mahout to build a recommender engine are as follows:

Top-level packages define the Mahout interfaces to these key abstractions:



Architecture of Recommender Engine



Step1: Create DataModel Object

- The DataModel object requires the **file object**, which contains the **path of the input file**. Create the DataModel object as shown below.

DataModel datamodel = new FileDataModel(new File("input file"));

Step2: Create UserSimilarity Object

Create **UserSimilarity** object using **PearsonCorrelationSimilarity** cla as shown below:

UserSimilarity similarity = new PearsonCorrelationSimilarity(datamodel);

Step3: Create UserNeighborhood object

This object computes a "neighborhood" of users like a given user. There are two types of neighborhoods:

- **ThresholdUserNeighborhood**
- **NearestNUserNeighborhood**

UserNeighborhood neighborhood = new ThresholdUserNeighborhood(3.0, similarity, model);

Step4: Create Recommender Object

- Create **UserbasedRecomender** object. Pass all the above created objects to its constructor as shown below.

UserBasedRecommender recommender = new GenericUserBasedRecommender(model, neighborhood, similarity);

Step5: Recommend Items to a User

- Recommend products to a user using the **recommend()** method of Recommender interface.
- This method requires **two parameters**. The first represents the **user id** of the user to whom we need to send the recommendations, and the second represents the **number of recommendations to be sent.**

```
List<RecommendedItem> recommendations = recommender.recommend(2, 3);

for (RecommendedItem recommendation : recommendations) {
  System.out.println(recommendation);
 }
```

Machine Learning Tools:

DATA SIZE	CLASSFICATION	TOOLS
Lines Sample Data	Analysis and Visualization	Whiteboard,…
KBs - low MBs Prototype Data	Analysis and Visualization	Matlab, Octave, R, Processing,
MBs - low GBs Online Data	Analysis	NumPy, SciPy, Weka, BLAS/LAPACK
	Visualization	Flare, AmCharts, Raphael, Protovis
GBs - TBs - PBs Big Data	Analysis	**Mahout,** Giraph MLib

STEPS TO BE FOLLOWED IN MAHOUT

1. Getting the data

2. Coping text files to The Hadoop Distributed File System (HDFS)

3. Convert our dataset into a SequenceFiles

4. Convert sequenceFiles to sparse vector file format

5. Running k-means text clustering algorithm

6. Interpreting the clustering final result

STEP 1

The first step is to get our dataset that will eventually represent our raw material on which we will test our clustering algorithm.

STEP 2

- After downloading our text collections locally, and in order to be able to handle it with mahout, it's time to copy it to our HDFS.

STEP 3

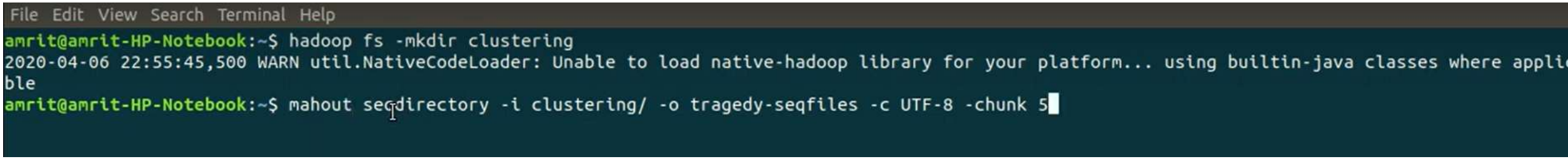
mahout seqdirectory -i <I> -o <O> -c UTF-8 -chunk 5

-i : specifying the input directory

-o : specifying the output directory

UTF-8 : specifying the encoding of our input files

-chunk : specifying the size of each block of data



STEP 4

- In order to be able to run properly, most algorithms in text mining require a numerical representation of texts.

STEP 4

- That’s why, we should turn the collections of texts we had in the previous steps into numerical feature vectors.

- Therefore, every document is represented as a vector where each element of the vector is a word and its weight respectively.

STEP 4

mahout seq2sparse -nv -i tragedy-seqfiles -o tragedy-vectors

-i : specifying the input directory

-o : specifying the output directory

-nv: very important option that keeps the files names for later use when displaying the result of text clustering

STEP 5

- Before passing to action by applying k-means clustering algorithm on our textual data, there is a simple step left.

- In order to have initial centroids values,we should, in the first place, run canopy clustering on our data.

```
mahout canopy -i <input vectors directory>
-o <output directory>
-t1 <threshold value 1>
-t2 <threshold value 2>
-dm
```

```
File  Edit  View  Search  Terminal  Help
amrit@amrit-HP-Notebook:~$ mahout canopy -i tragedy-vectors/tf-vectors -o tragedy-vectors/tragedy-canopy-centroids -dm org.apache.mahout.common.distan
ce.CosineDistanceMeasure -t1 1500 -t2 2000
```

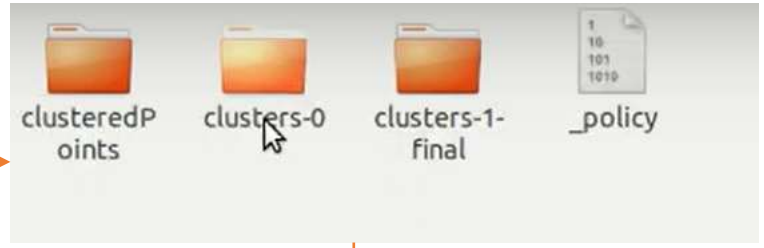
STEP 5

- Once we have generated initial centroids values we can finally run k-means algorithm on our documents.

STEP 5


```
mahout kmeans -i <INPUT> -c <CENTROID
DIRECTORY> -o <OUTPUT> -dm <DISTANCE
MEASURE> –clustering -cl -cd <convergence
delta parameter> -ow -x <MAX NO OF
ITERATIONS> -k <NO OF CLUSTERS>
```

```
amrit@amrit-HP-Notebook:~$ mahout kmeans -i tragedy-vectors/tfidf-vectors -c tragedy-canopy-centroids -o tragedy-kmeans-clusters -dm org.apache.mahou
.common.distance.CosineDistanceMeasure --clustering -cl -cd 0.1 -ow -x 20 -k 10
```




THANK YOU