M.Tech Program

Advanced Industry Integrated Programs

Jointly offered by University and LTIMindTree

Applied Machine Learning

Knowledge partner

LTIMindtree

Implementation partner

L&T EduTech

Modules to cover….

1. Supervised Learning
2. Case Study 1
3. Advanced Learning Algos
4. Case Study 2
5. **Unsupervised Learning and Recommender Systems**
6. Case Study 3

Unsupervised Learning and Recommender Systems

Unsupervised Learning and Recommender Systems

Learning outcomes

- **Understand and apply unsupervised learning techniques**, such as K-means clustering and anomaly detection, to identify patterns and outliers in data without labeled outcomes.
- **Develop recommender systems** using both collaborative filtering approaches and content-based deep learning methods and evaluate their effectiveness in personalized content delivery.
- **Analyze and implement deep reinforcement learning models**, including Deep Q Networks (DQN), to solve complex decision-making problems in dynamic environments.

Unsupervised Learning and Recommender Systems

Topics

- Unsupervised Learning Techniques
- K-Means Clustering
- Optimal number of Cluster
- Anomaly Detection
- Choosing Between Supervised Learning and Anomaly Detection
- Recommender Systems with Collaborative Filtering
- Recommender Systems with Content-Based Deep Learning
- Deep Reinforcement Learning Models

Unsupervised Learning Techniques

Unsupervised Learning and Recommender Systems

Introduction to Unsupervised Learning

What is Unsupervised Learning?

- ✓ Unsupervised learning is a type of machine learning algorithm that is used to draw inferences from datasets consisting of input data without labeled responses.
- ✓ Unlike supervised learning, where the model is trained on labeled data, unsupervised learning works on unlabeled data to discover hidden patterns or intrinsic structures within the data.

Key Characteristics:

- ✓ No pre-existing labels or outputs.
- ✓ The model identifies natural grouping or patterns based on input features.
- ✓ Commonly used for clustering and association problems.

Unsupervised Learning and Recommender Systems

Unsupervised Learning - Example

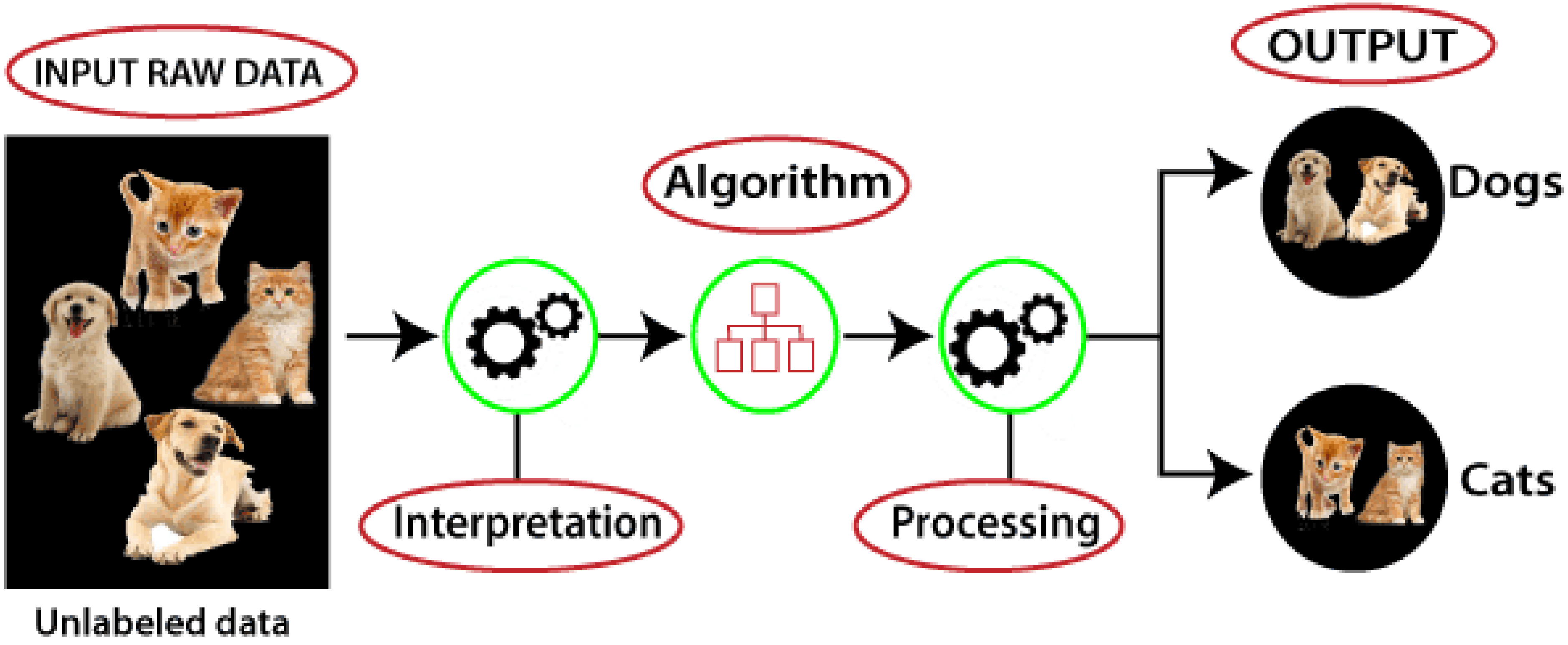


Image Source: https://quizmanthon.com/elearn-unsupervised-ml.php

Unsupervised Learning and Recommender Systems

Key Differences from Supervised Learning

Aspect	Supervised Learning	Unsupervised Learning
Data Labeling	Uses labeled data (input-output pairs).	Uses unlabeled data, with no explicit instructions on what to do with it.
Objective	Aims to learn a mapping from inputs to outputs, making predictions based on learned data.	Aims to infer the natural structure present in a set of data points.
Examples	Classification and Regression tasks (e.g., spam detection, house price prediction).	Clustering (e.g., customer segmentation, market basket analysis) and dimensionality reduction (e.g., Principal Component Analysis).
Learning Process	Relies on a development of mathematical/functional relationship between input and output.	Self-organizes the data to identify patterns without feedback.

Unsupervised Learning and Recommender Systems

Key Differences from Supervised Learning..

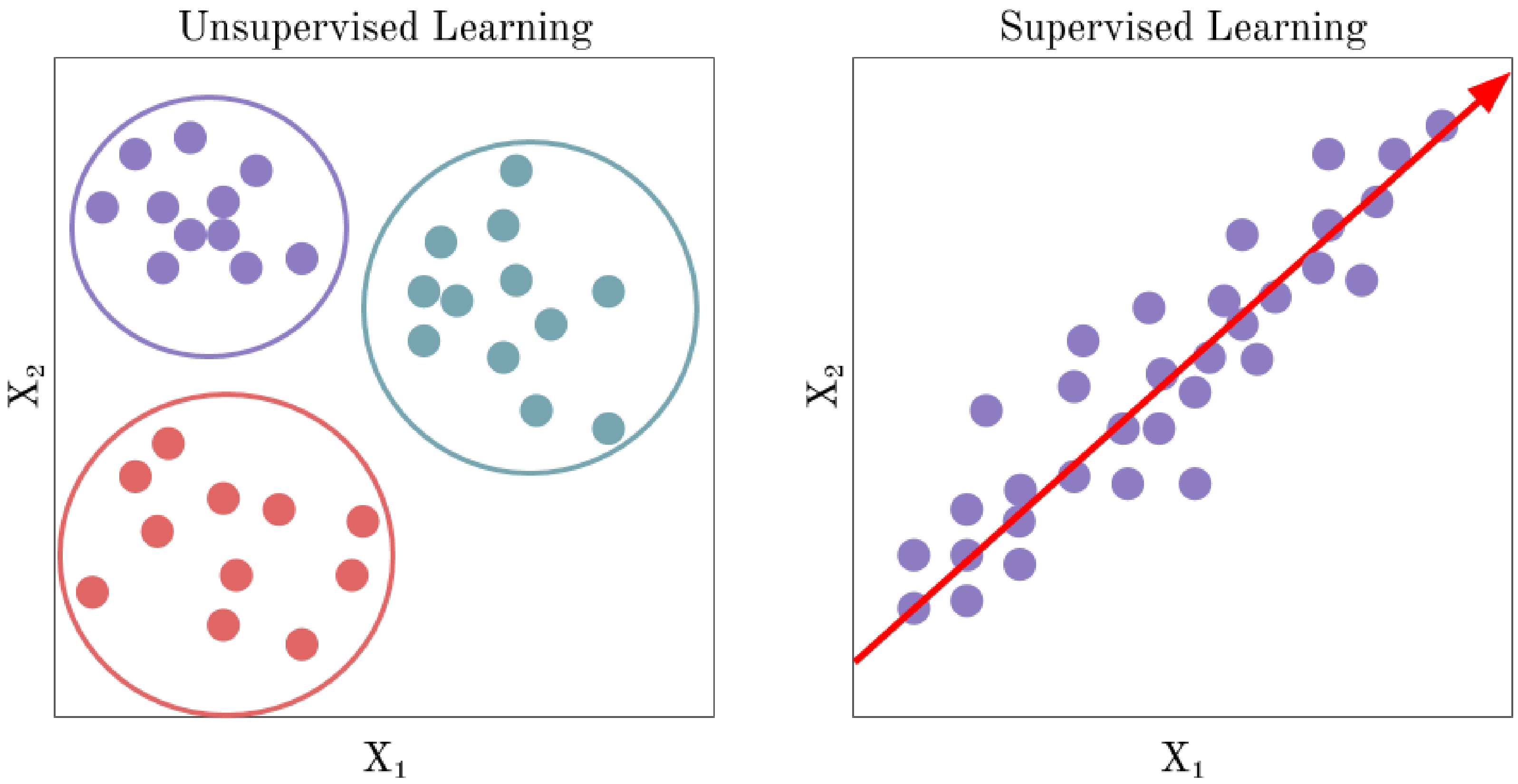


Image Source: https://www.researchgate.net/figure/Supervised-and-unsupervised-learning_fig2_370979689

K-Mean clustering

Unsupervised Learning and Recommender Systems

Unsupervised Learning – Clustering:

Clustering is a technique used to group similar data points into clusters, where data points in the same group are more similar to each other than to those in other groups.

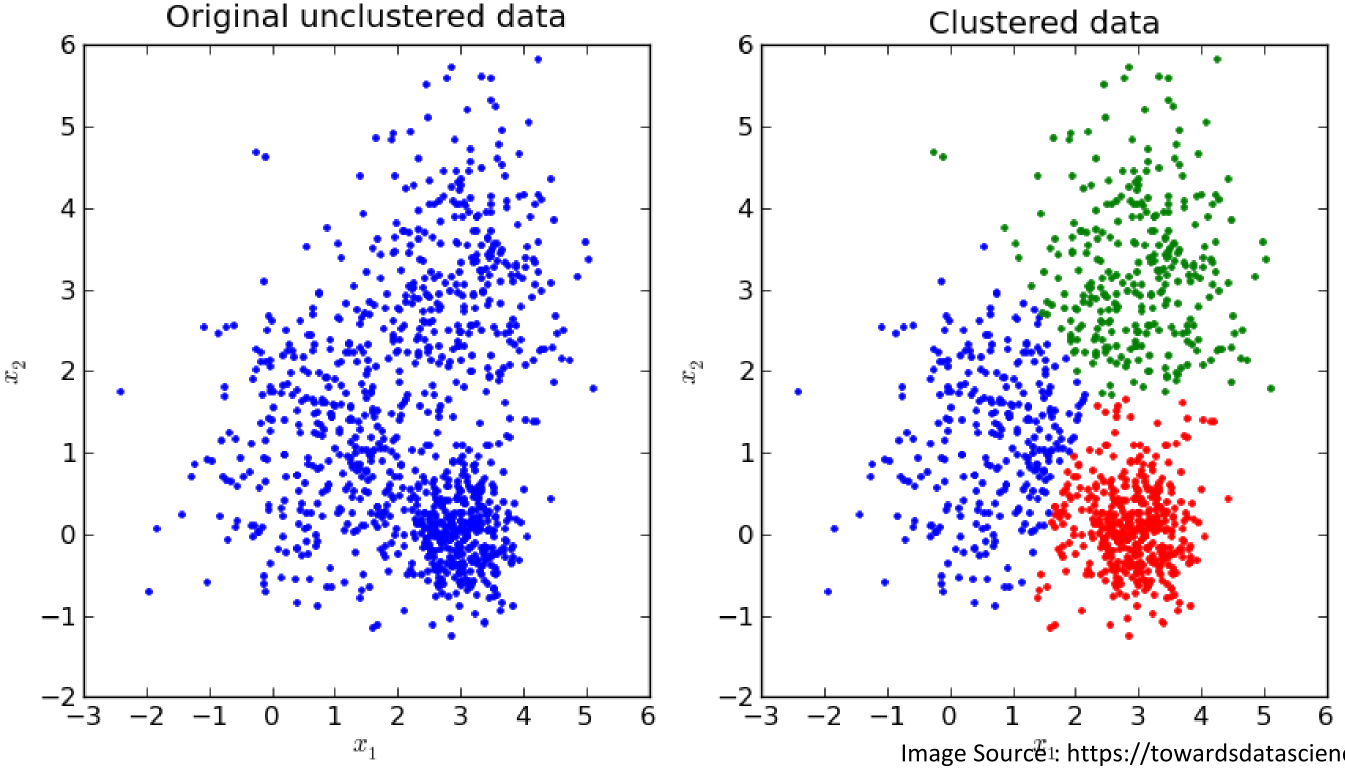


Image Source: https://towardsdatascience.com/k-means-data-clustering-bce3335d2203

Unsupervised Learning and Recommender Systems

Introduction to K-Means Clustering

- ✓ K-Means is an **unsupervised learning algorithm** used for **clustering** data into a predefined number of clusters (K).
- ✓ The algorithm partitions the dataset into K distinct, non-overlapping subsets (clusters).

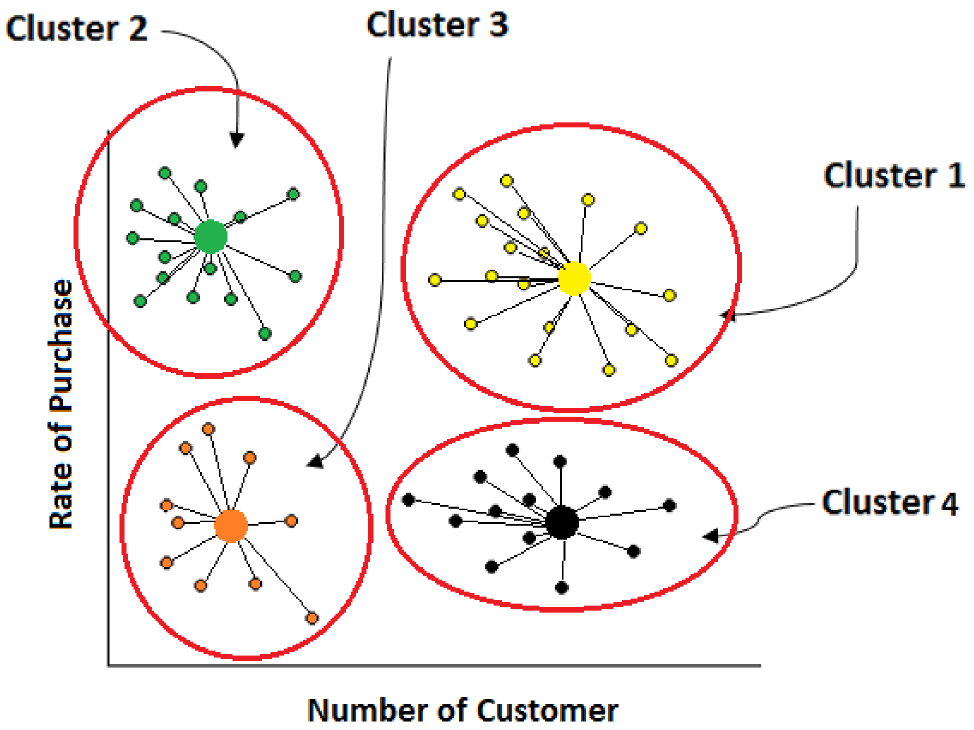


Image Source : https://towardsai.net/p/machine-learning/fully-explained-k-means-clustering-with-python

Unsupervised Learning and Recommender Systems

How K-Means Clustering Works

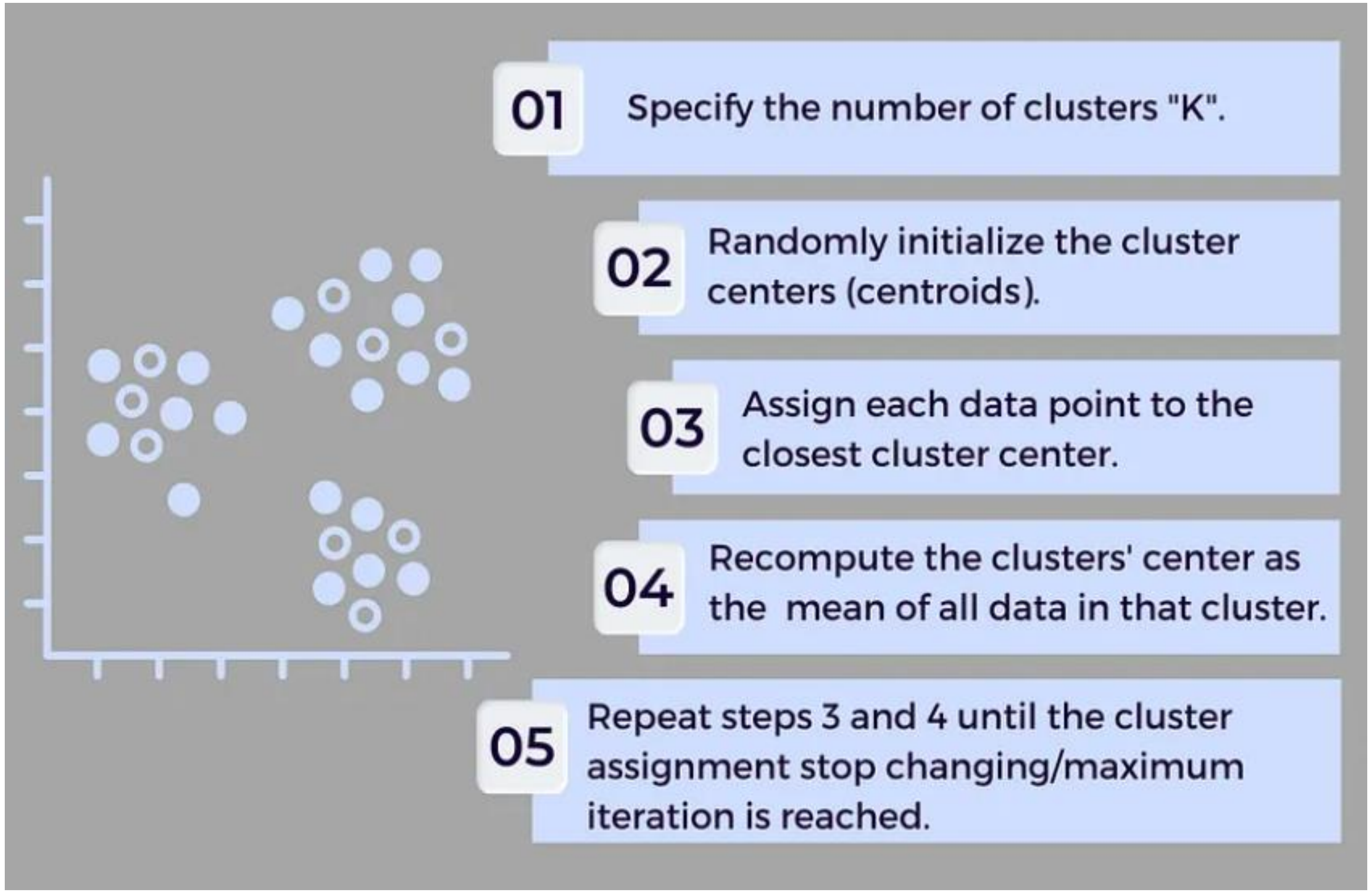


Image Source : https://towardsdatascience.com/how-to-perform-kmeans-clustering-using-python-7cc296cec092

Unsupervised Learning and Recommender Systems

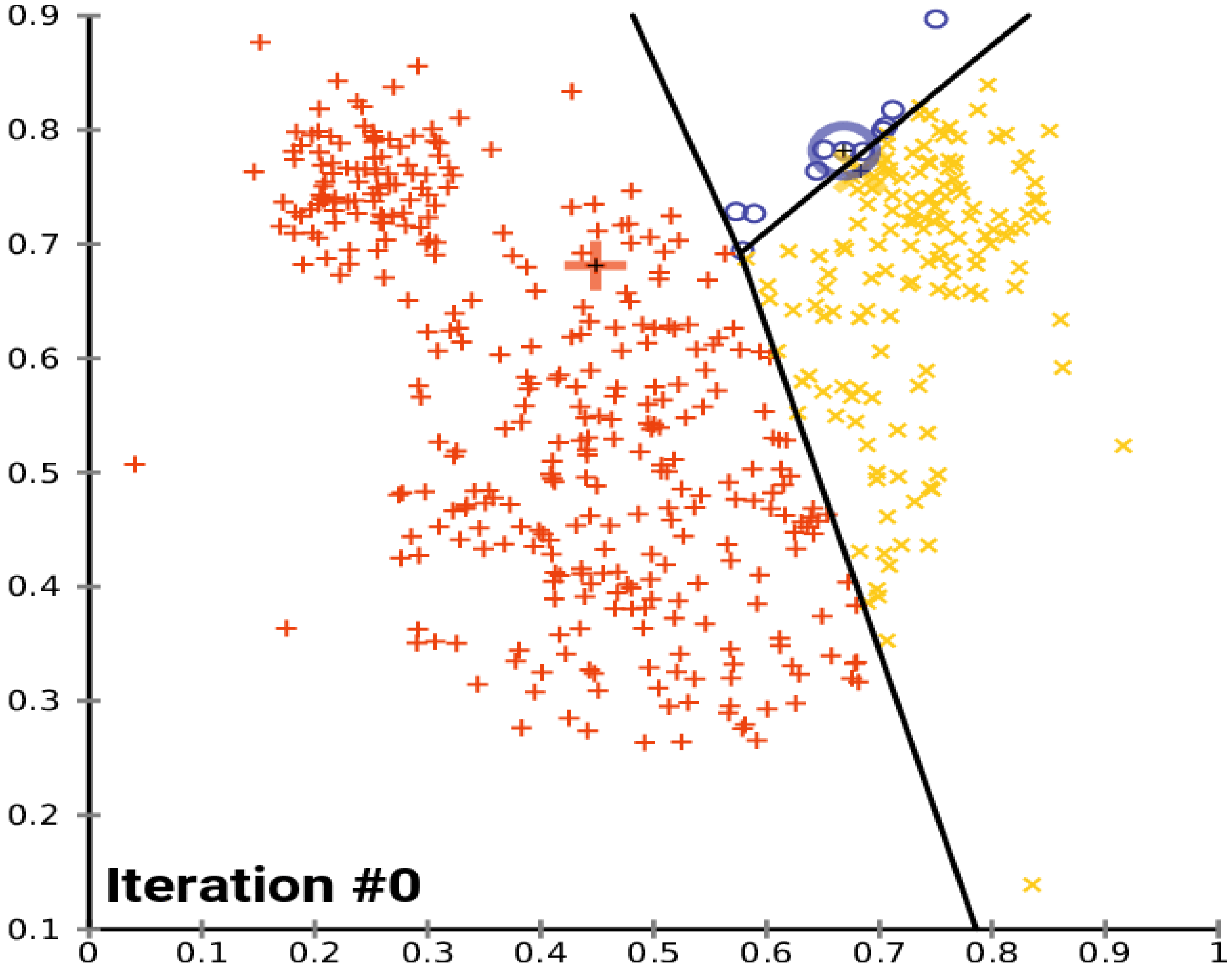


Image Source : Wikipedia

Unsupervised Learning and Recommender Systems

K-Mean clustering

Convergence Criteria:

✓ The algorithm stops when the centroids do not change or the maximum number of iterations is reached.

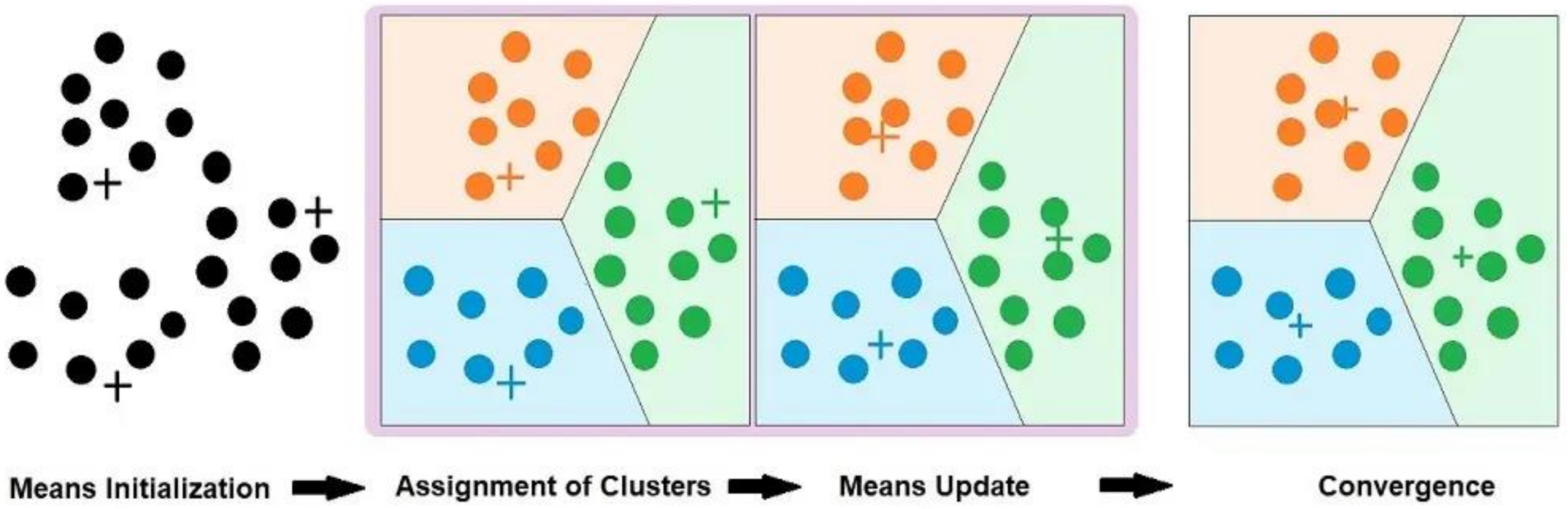


Image Source: https://www.ejable.com/tech-corner/ai-machine-learning-and-deep-learning/k-means-clustering/

Unsupervised Learning and Recommender Systems

- **Objective Function**

To minimize the variance within each cluster and maximize the variance between different clusters.

number of clusters

number of cases

centroid for cluster *j*

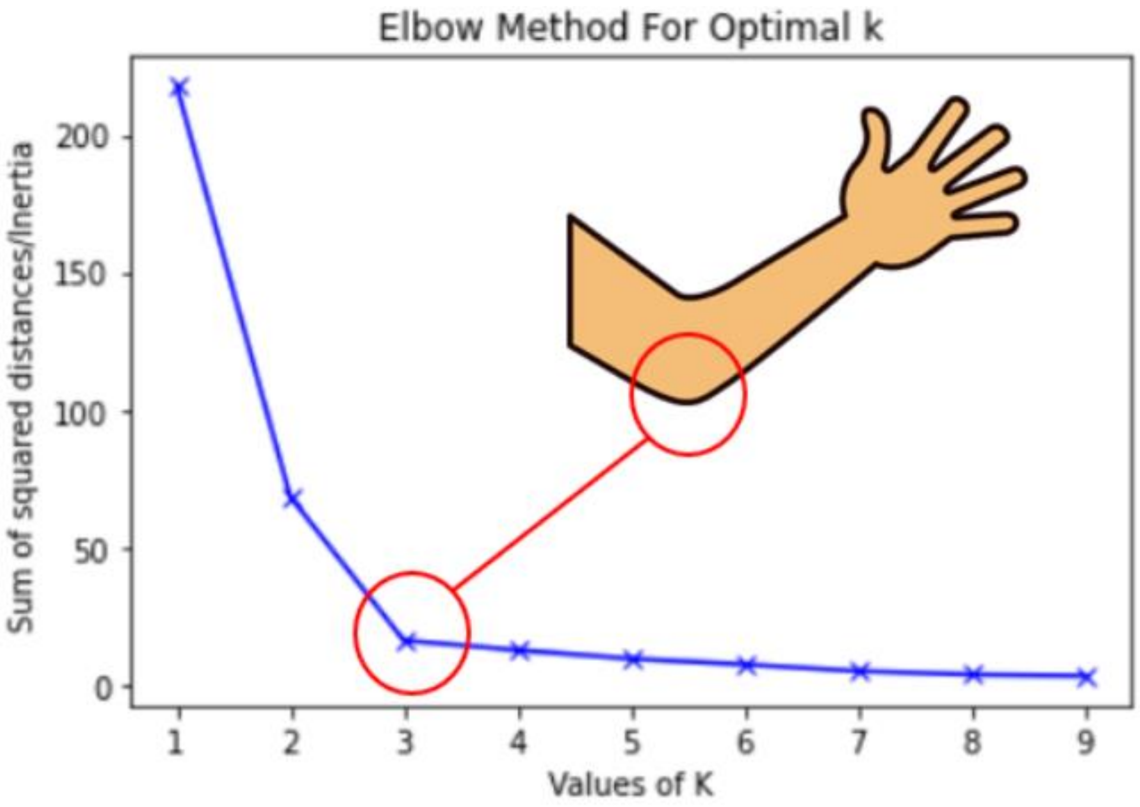
case *i*

$$\text{objective function} \leftarrow J = \sum_{j=1}^{k} \sum_{i=1}^{n} \underbrace{\left\| x_i^{(j)} - c_j \right\|^2}_{\text{Distance function}}$$

Unsupervised Learning and Recommender Systems

K-Mean clustering : Choosing the Number of Clusters (K)

- **Elbow Method:**
  - ✓ Plot the **Within-Cluster Sum of Squares (WCSS)** against the number of clusters K.
  - ✓ Look for an "elbow" point where the WCSS starts to diminish at a slower rate. This point suggests the optimal K.



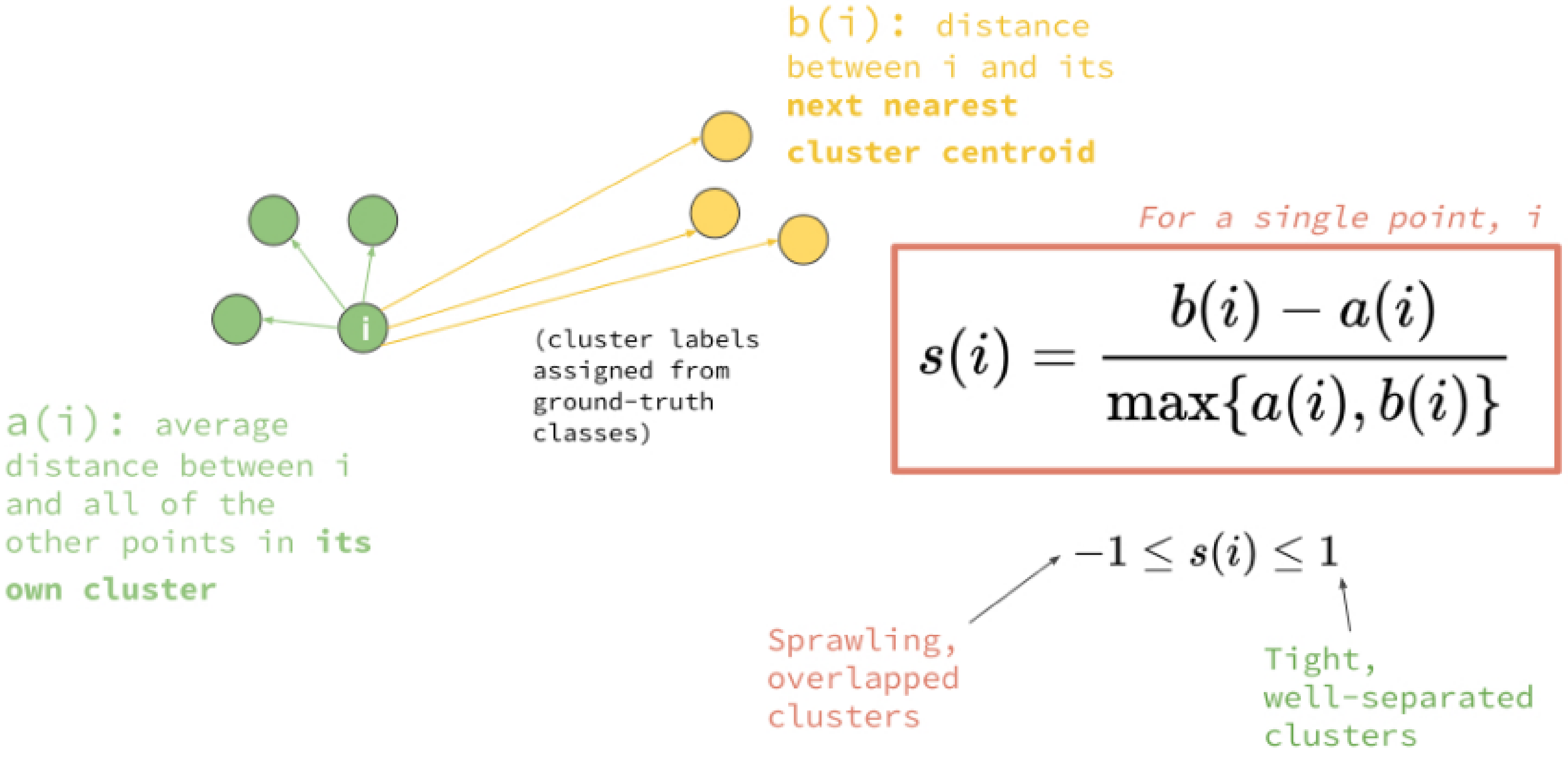
Line plot between K and inertia

Image Source: https://www.linkedin.com/pulse/k-means-elbow-method-clustering-bogus%C5%82aw-konefa%C5%82

Unsupervised Learning and Recommender Systems

K-Mean clustering - Silhouette Analysis:

- Silhouette score consider both with-in and between cluster distances

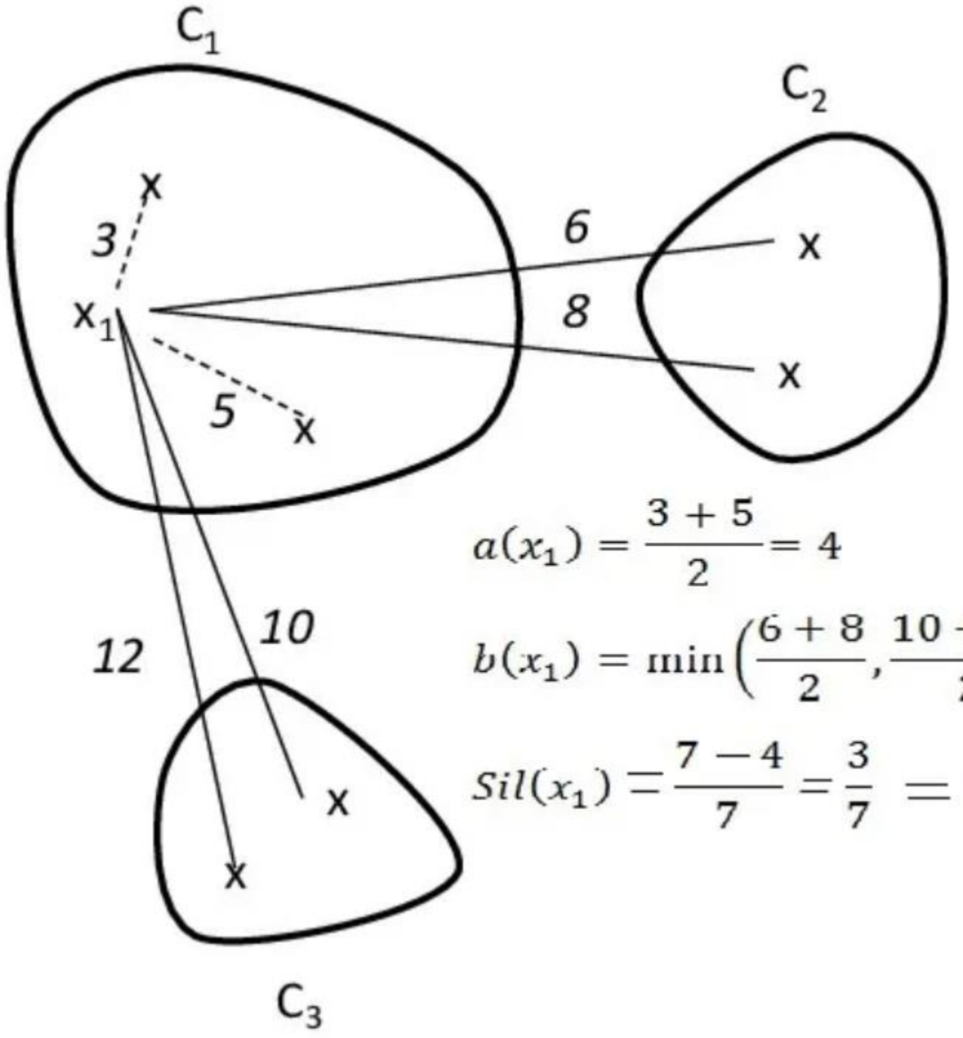


$$s(i) = \frac{b(i) - a(i)}{\max\{a(i), b(i)\}}$$

$$-1 \leq s(i) \leq 1$$

Image Source: https://www.platform.ai/post/the-silhouette-loss-function-metric-learning-with-a-cluster-validity-index

Unsupervised Learning and Recommender Systems



$$a(x_1) = \frac{3+5}{2} = 4$$

$$b(x_1) = \min\left(\frac{6+8}{2}, \frac{10+12}{2}\right) = 7$$

$$Sil(x_1) = \frac{7-4}{7} = \frac{3}{7} = 0.42$$

Image Source: https://medium.com/@MrBam44/how-to-evaluate-the-performance-of-clustering-algorithms

Unsupervised Learning and Recommender Systems

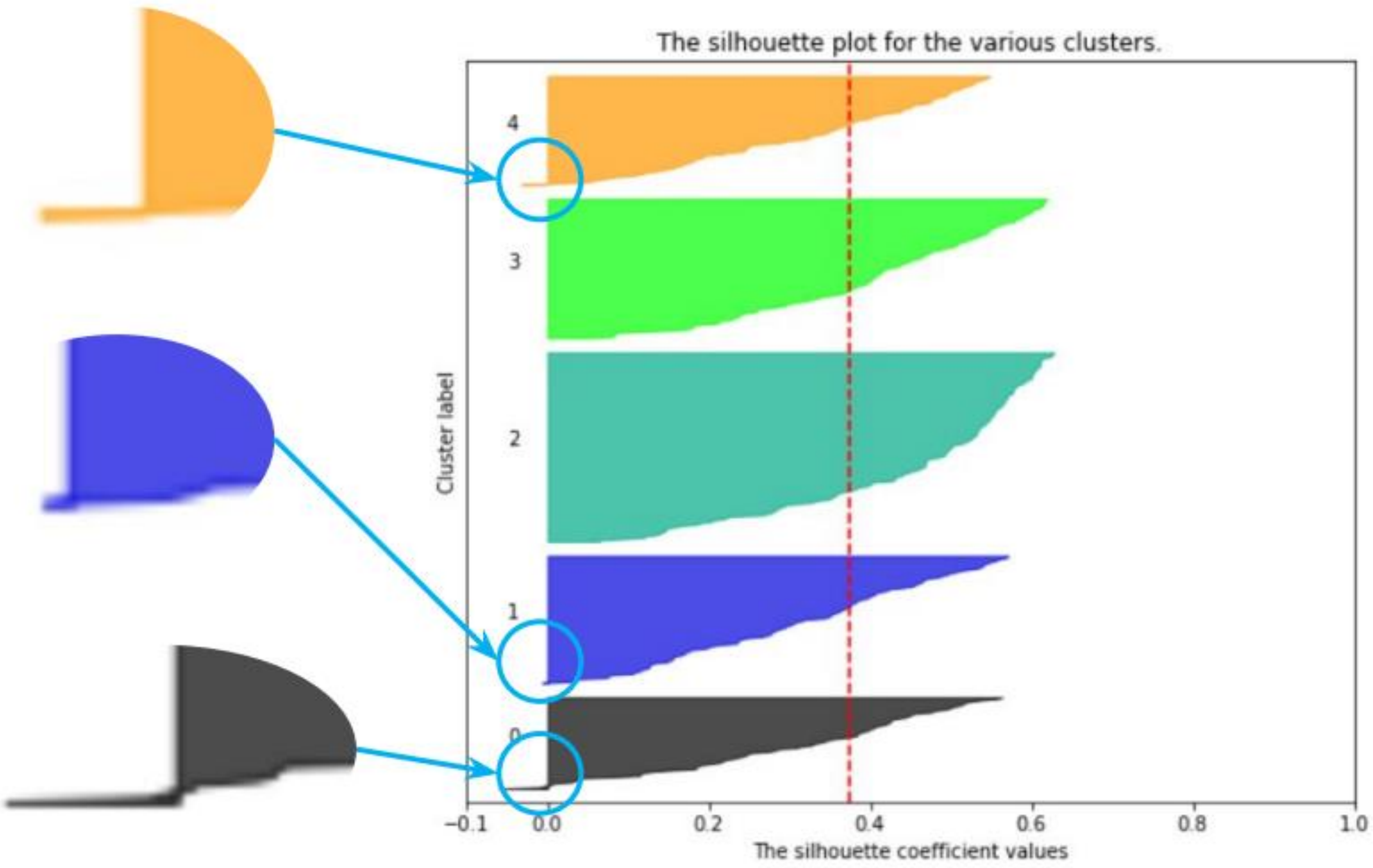
K-Mean clustering - Silhouette Analysis..

Silhouetee score ranges from -1 to +1:

- ✓ Positive values indicate that data points belong to the correct clusters, indicating good clustering results.
- ✓ A score of zero suggests overlapping clusters or data points equally close to multiple clusters.
- ✓ Negative values indicate that data points are assigned to incorrect clusters, indicating poor clustering results.
- ✓ A higher Silhouette score indicates better clustering results.

Unsupervised Learning and Recommender Systems

- ✓ Any silhouetee score less than 0 is considered as an outlier
- ✓ In the plot shown, there are outliers in the 4th, 1st and 0th cluster



Unsupervised Learning and Recommender Systems

K-Mean clustering - Applications of K-Means Clustering:

✓ **Customer Segmentation :** Grouping customers based on purchasing behavior for targeted marketing.

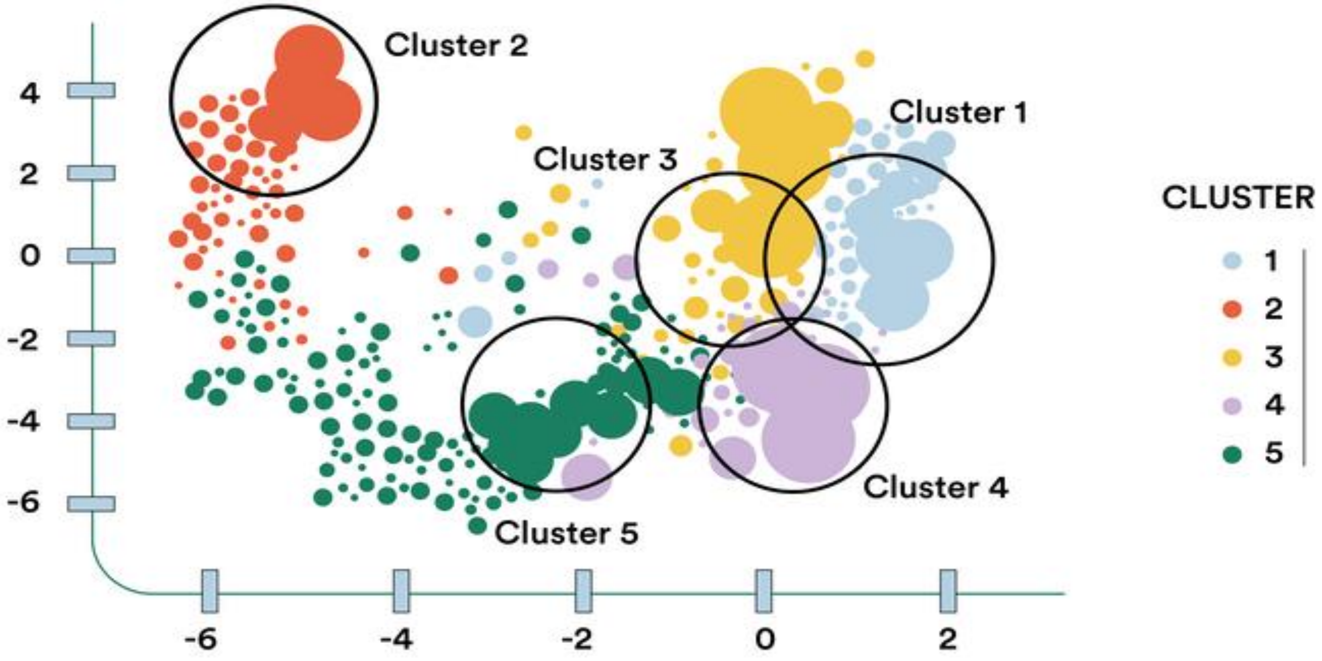


Image Source: https://segmentify.com/blog/top-customer-segmentation-examples-every-marketer-needs-to-know/

Unsupervised Learning and Recommender Systems

K-Mean clustering - Applications of K-Means Clustering:

✓ **Image Segmentation: Extracting the Region of Interest from the Image**

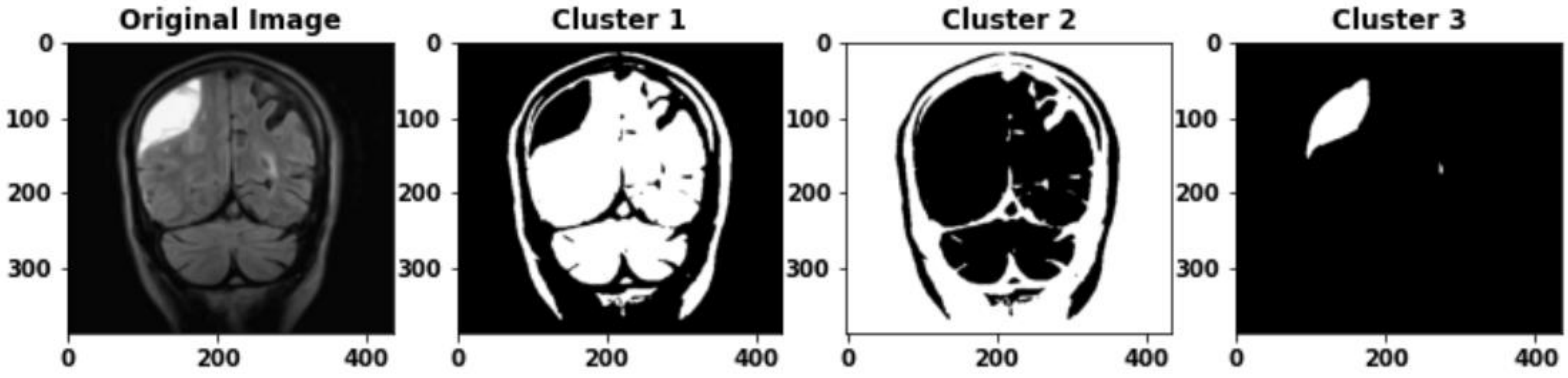


Image Source : https://www.tandfonline.com/doi/full/10.1080/03772063.2021.1892536

Unsupervised Learning and Recommender Systems

K-Mean clustering - Applications of K-Means Clustering:

- ✓ **Image Compression:** Reducing the number of colors in an image to compress it while maintaining quality.

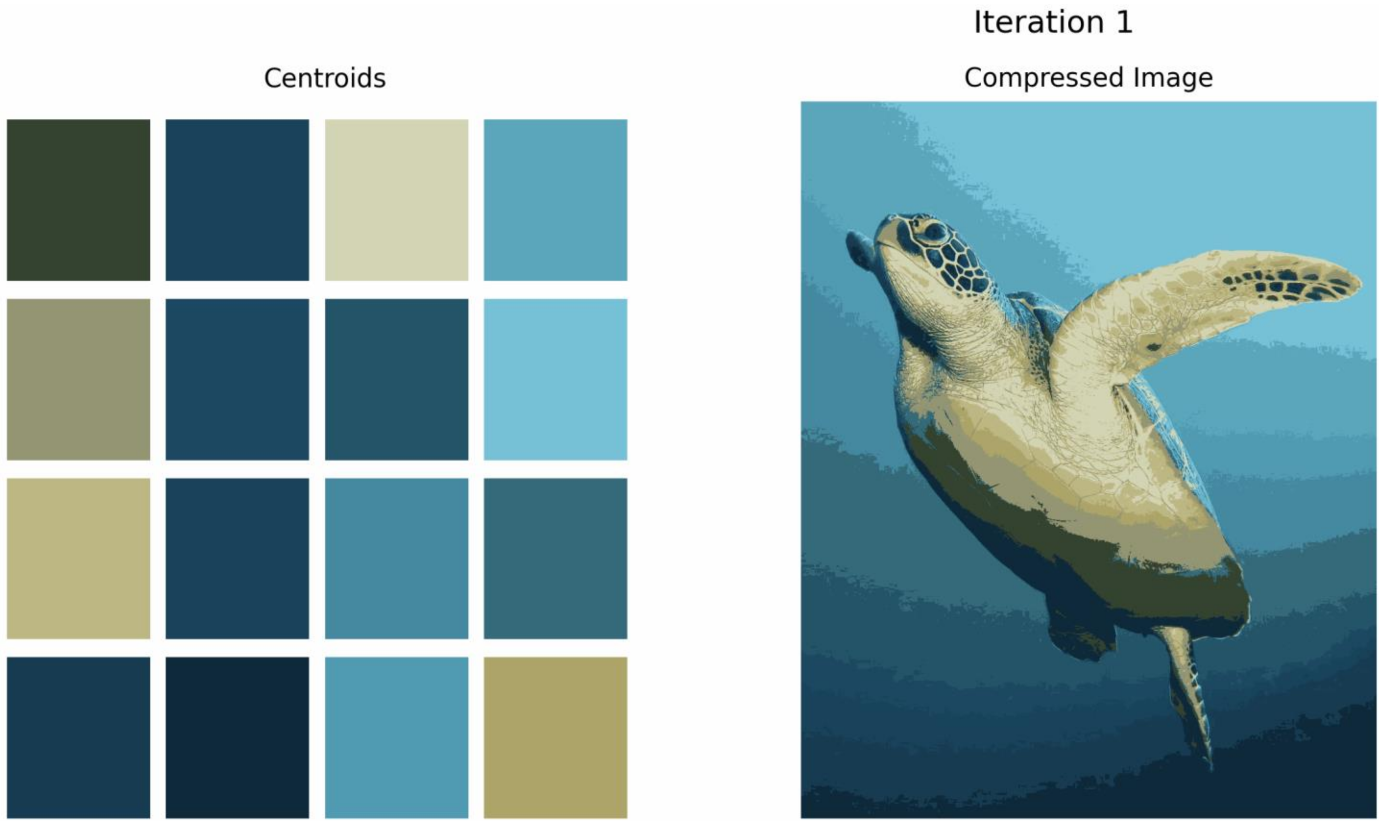


Image Source: https://towardsdatascience.com/clear-and-visual-explanation-of-the-k-means-algorithm-applied-to-image-compression-b7fdc547e410

Unsupervised Learning and Recommender Systems

K-Mean clustering - Document Clustering:

- ✓ Organizing a large collection of documents into a small number of meaningful clusters for easier management and retrieval.

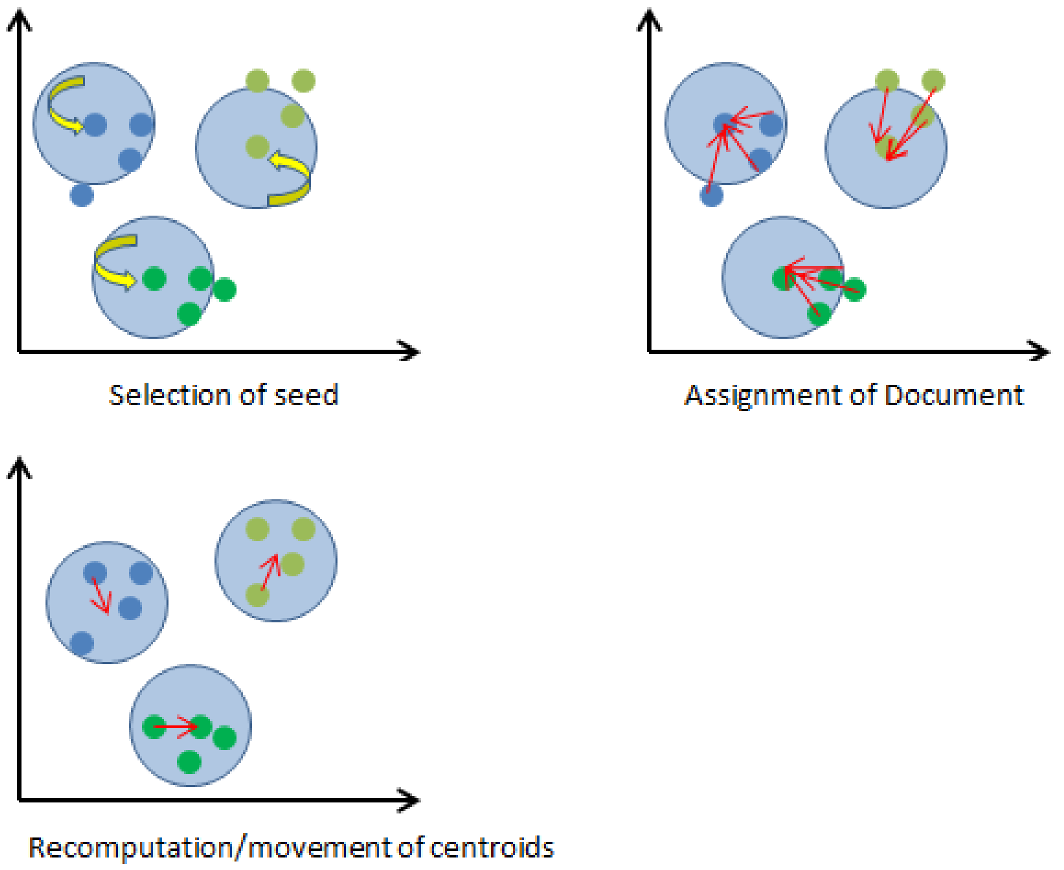


Image Source: https://www.codeproject.com/Articles/439890/Text-Documents-Clustering-using-K-Means-Algorithm

Unsupervised Learning and Recommender Systems

K-Mean clustering - Anomaly Detection:

✓ Identifying unusual data points by clustering normal data points together and considering outliers as anomalies.

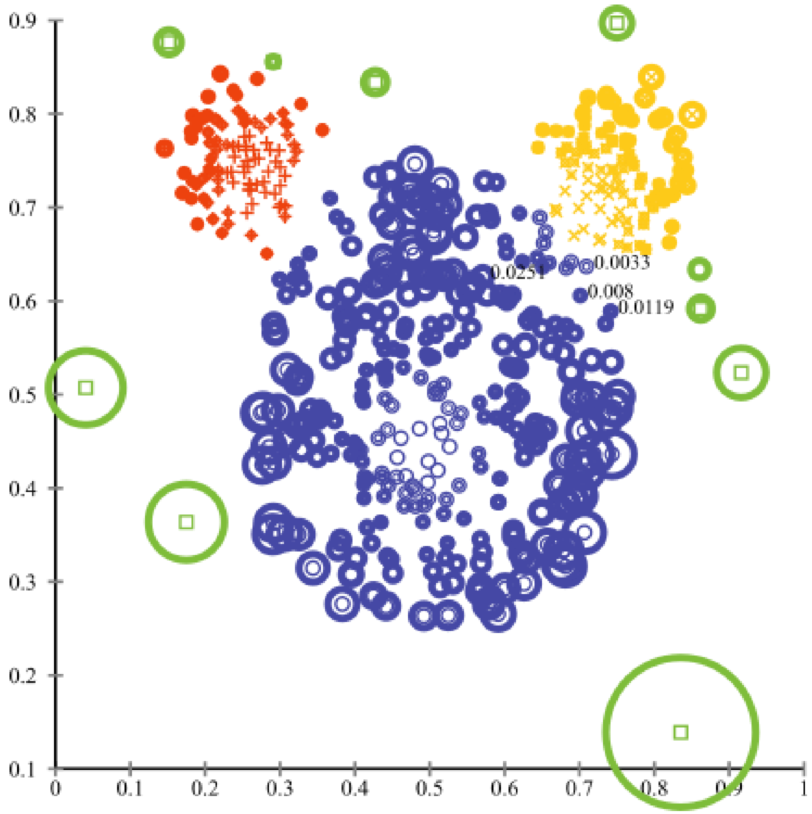
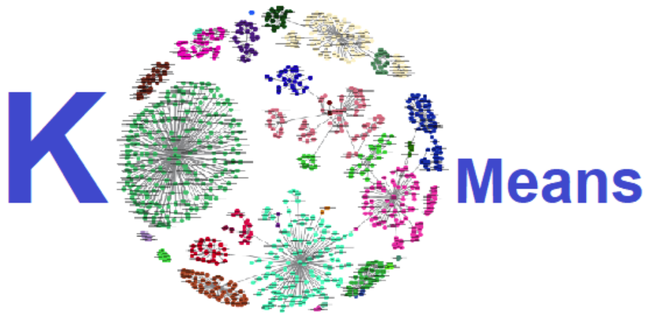


Image Source: https://www.mdpi.com/2673-2688/1/1/5

Unsupervised Learning and Recommender Systems

K-Mean clustering : Advantages and Limitations of K-Means Clustering

- **Advantages:**
  - ✓ Simple and easy to implement.
  - ✓ Scales well to large datasets.
  - ✓ Works well with compact, well-separated clusters.



- **Limitations:**
  - ✓ Requires the number of clusters (K) to be specified in advance.
  - ✓ Sensitive to initial centroid placement.
  - ✓ Not suitable for non-spherical clusters or clusters with different sizes and densities.
  - ✓ Can be affected by outliers and noise in the data.

Image Source: https://www.linkedin.com/pulse/k-means-clustering-gokulprasanth-t-tsgnc

Anomaly Detection

Unsupervised Learning and Recommender Systems

Anomaly Detection

- ✓ Anomaly detection is the process of identifying **rare items, events, or observations** that raise suspicions by differing significantly from the majority of the data.
- ✓ Anomalies are also referred to as outliers, novelties, noise, deviations, or exceptions.

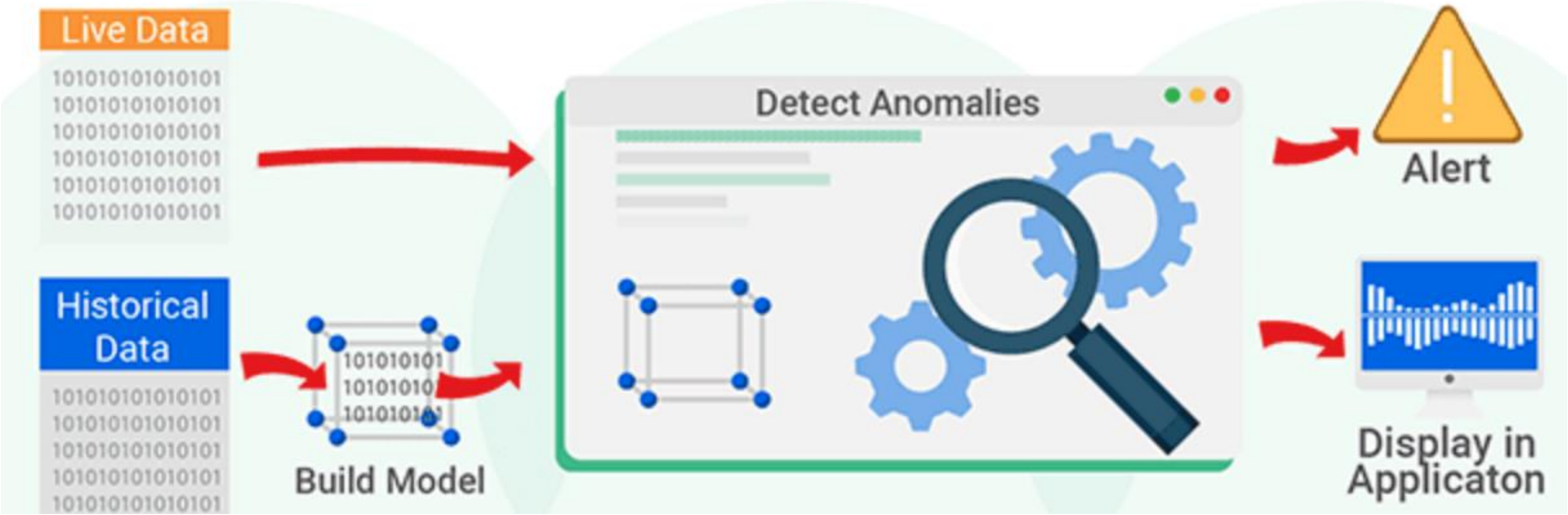
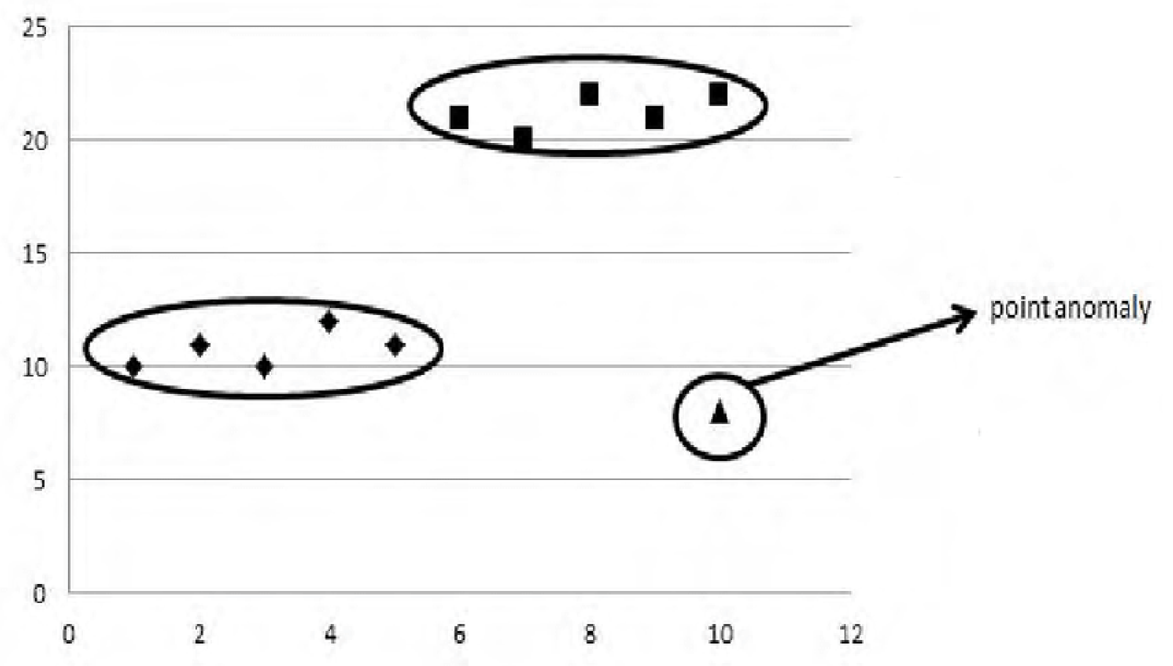


Image Source: https://www.linkedin.com/pulse/importance-anomaly-detection-digital-payments-shantanu-thakre

Unsupervised Learning and Recommender Systems

Anomaly Detection - Types of Anomalies:

- **Point Anomalies:**
  - ✓ A single data point is considered an anomaly if it deviates significantly from the rest of the dataset.
  - ✓ **Example:** Credit card transactions with unusually high amounts.



Unsupervised Learning and Recommender Systems

Anomaly Detection - Types of Anomalies:

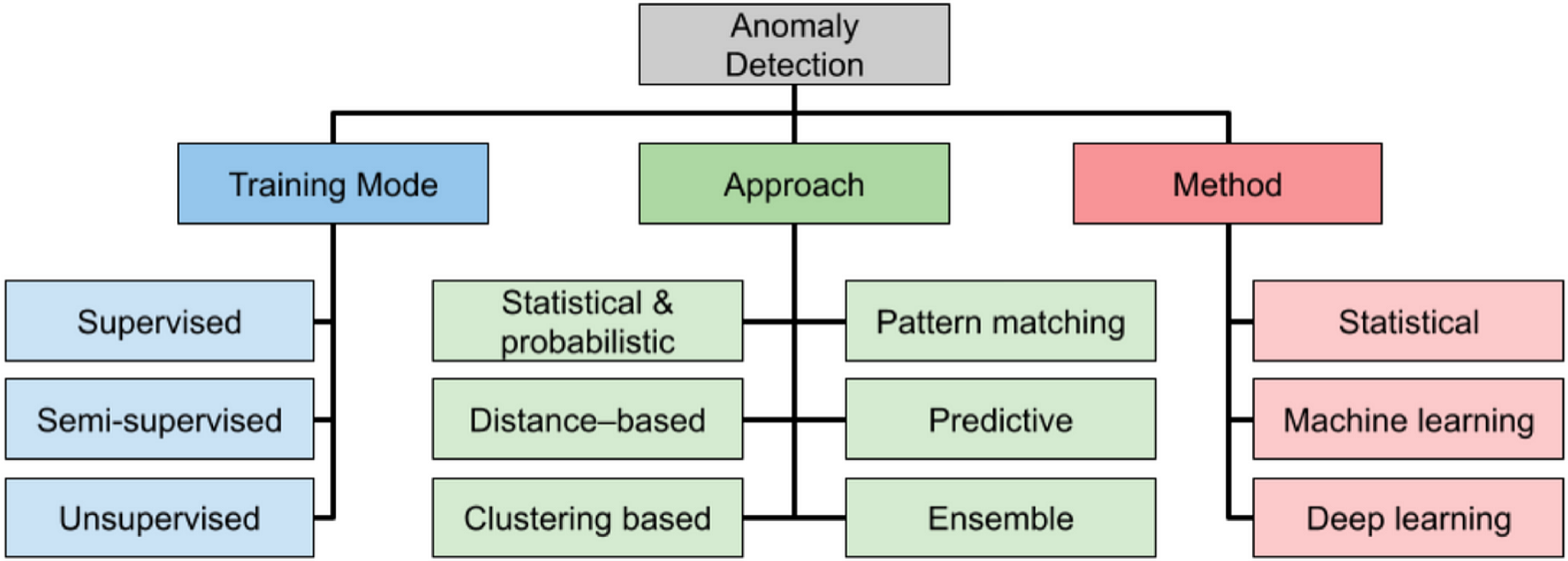


Image Source : https://medium.com/@saipraneethk181200/detection-of-anomalies-in-multivariate-time-series-4acf4fef81e4

Unsupervised Learning and Recommender Systems

Anomaly Detection - Types of Anomalies:

- **Contextual Anomalies:**
  - ✓ Data points that are considered anomalous within a specific context but not otherwise
  - ✓ **Example:** Temperature readings in a geographical area that are normal for winter but anomalous for summer.

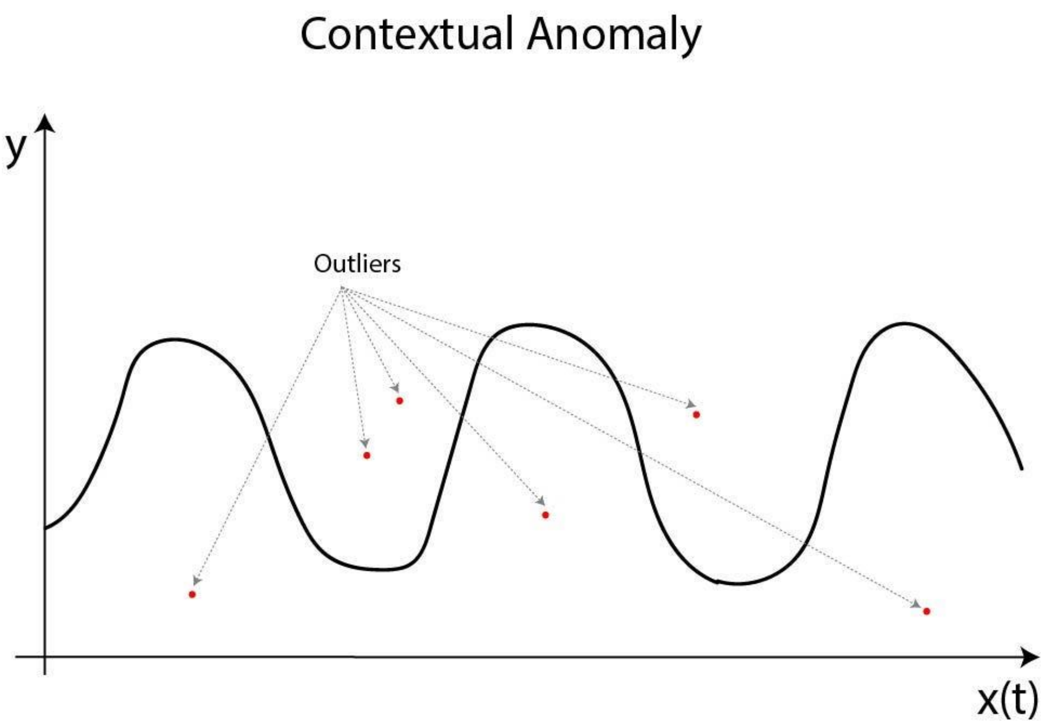


Image Source: https://medium.com/datadailyread/types-of-data-anomalies-2f6fb1747eb1

Unsupervised Learning and Recommender Systems

Anomaly Detection - Types of Anomalies:

- **Collective Anomalies:**
  - ✓ A collection of related data points is anomalous, but individual data points within the collection may not be anomalies.
  - ✓ **Example:** A sudden surge in network traffic could indicate a Distributed Denial of Service (DDoS) attack.

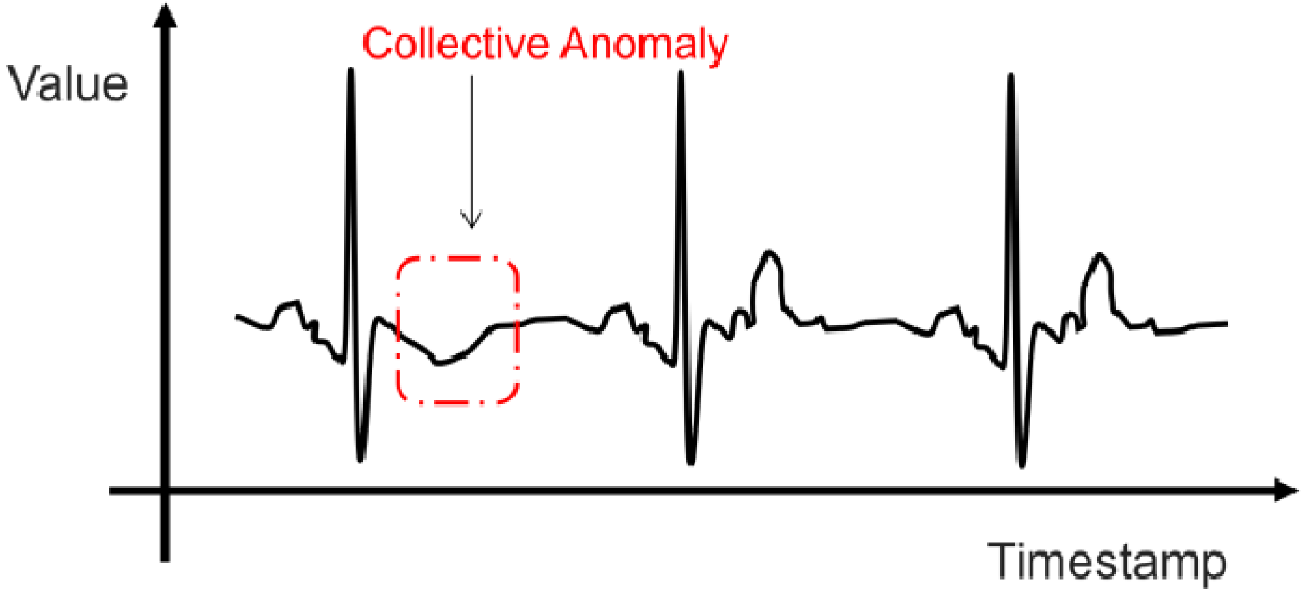


Image Source: https://www.researchgate.net/figure/An-example-of-ECG-collective-anomaly_fig1_374861640

Unsupervised Learning and Recommender Systems

Anomaly Detection - Anomaly Detection Techniques:

- **Statistical Methods:**
  - ✓ Based on the assumption that normal data points occur in high probability regions of a stochastic model, while anomalies occur in the low probability regions.
  - ✓ Techniques include **Z-Score, Grubbs' Test,** and **Isolation Forest.**

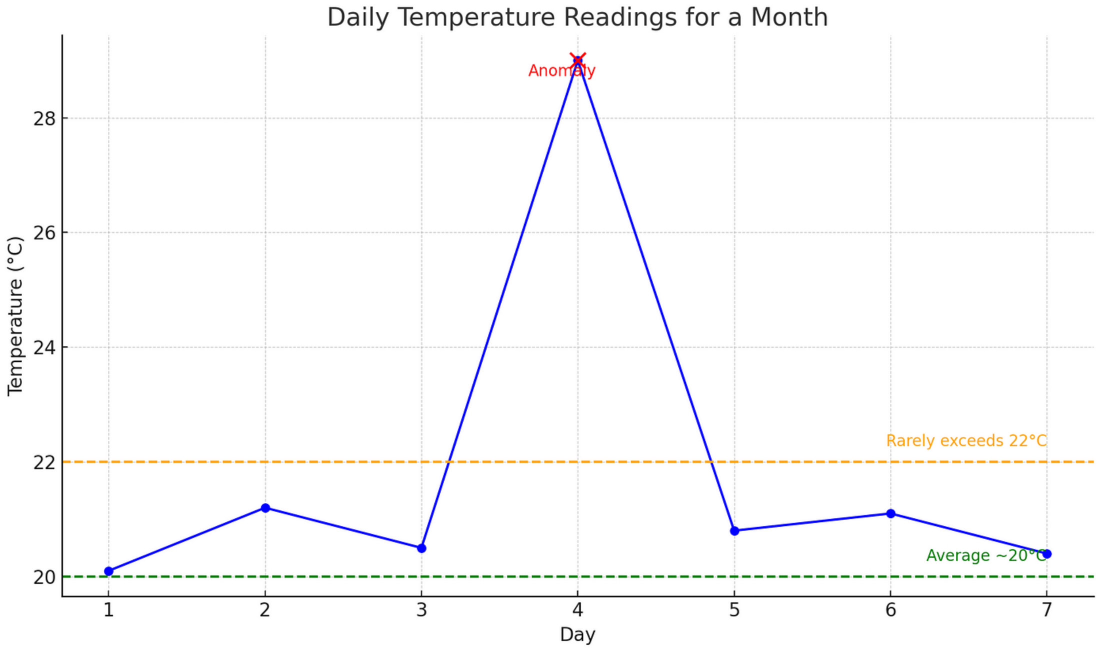
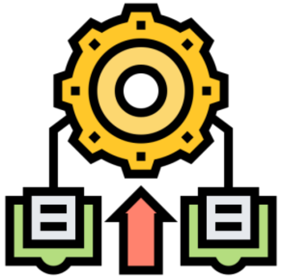


Image Source: https://medium.com/@gabrielpierobon/statistical-techniques-for-anomaly-detection-part-2-grubbs-test-univariate-and-multivariate-dcc45c70584c

Unsupervised Learning and Recommender Systems

Anomaly Detection - Machine Learning Methods:



- **Machine Learning Methods:**
  - **Supervised Learning:**
    - ✓ Requires a labeled dataset with normal and anomaly data points for training.
    - ✓ Algorithms include **Support Vector Machines (SVM)**, **Decision Trees**, and **Random Forests**.
  - **Unsupervised Learning:**
    - ✓ No labeled data required. The model learns the structure of normal data and identifies deviations as anomalies.
    - ✓ Techniques include **K-Means Clustering**, **DBSCAN**, and **Autoencoders.**

Unsupervised Learning and Recommender Systems

Anomaly Detection - Machine Learning Methods:

- **Deep Learning Methods:**
  - ✓ Effective for complex and high-dimensional datasets.
  - ✓ Techniques include **Recurrent Neural Networks (RNNs)**, **Long Short-Term Memory (LSTM)** networks, and **Variational Autoencoders (VAE)**.

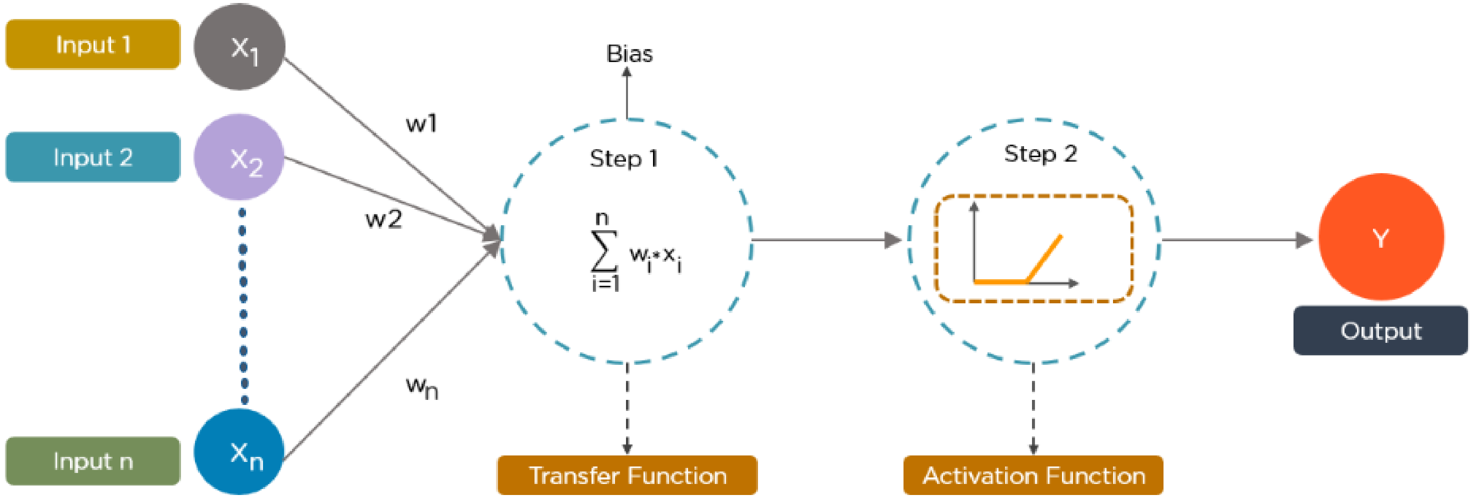


Image Source: https://www.researchgate.net/figure/Deep-Learning-Algorithm-41-Convolutional-Neural-Network-Convolutional-Neural-Network_fig3_367511650

Unsupervised Learning and Recommender Systems

Anomaly Detection - Applications of Anomaly Detection:

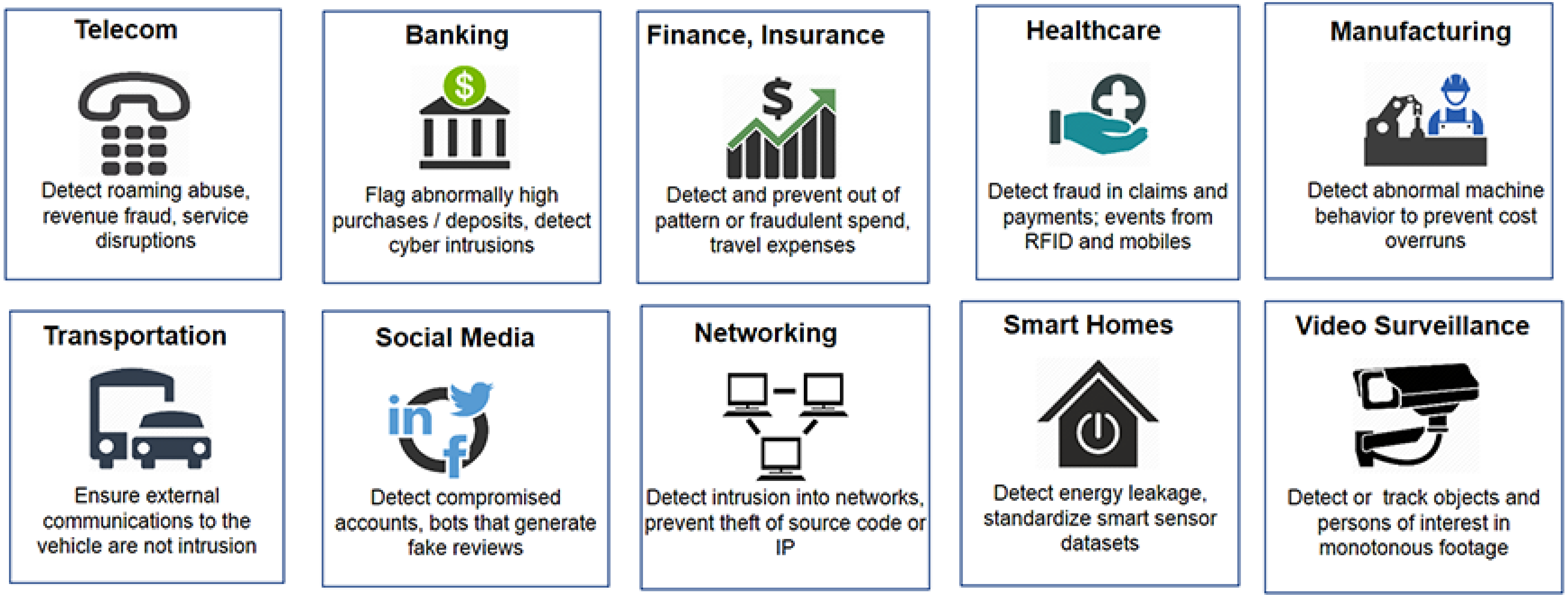


Image Source: https://www.gathr.one/blog/new-approaches-to-real-time-anomaly-detection-for-streaming-data/

Unsupervised Learning and Recommender Systems

Anomaly Detection - Challenges in Anomaly Detection:

- **High Dimensionality:**
  - ✓ Analyzing data with many features can be computationally expensive and may lead to overfitting.
- **Imbalanced Data:**
  - ✓ Anomalies are rare, leading to imbalanced datasets where the majority class (normal data) dominates.
- **Dynamic Data:**
  - ✓ In real-time applications, the distribution of data can change over time, requiring adaptive models.
- **False Positives and Negatives:**
  - ✓ Balancing sensitivity (detecting true anomalies) and specificity (avoiding false alarms) is challenging.

Choosing Between Supervised Learning and Anomaly Detection

Unsupervised Learning and Recommender Systems

Choosing Between Supervised Learning and Anomaly Detection

Introduction to Supervised Learning vs. Anomaly Detection

- **Supervised Learning:**

  ✓ Involves learning a function that maps an input to an output based on example input-output pairs.

  ✓ Requires a **labeled dataset** where the desired output is known.

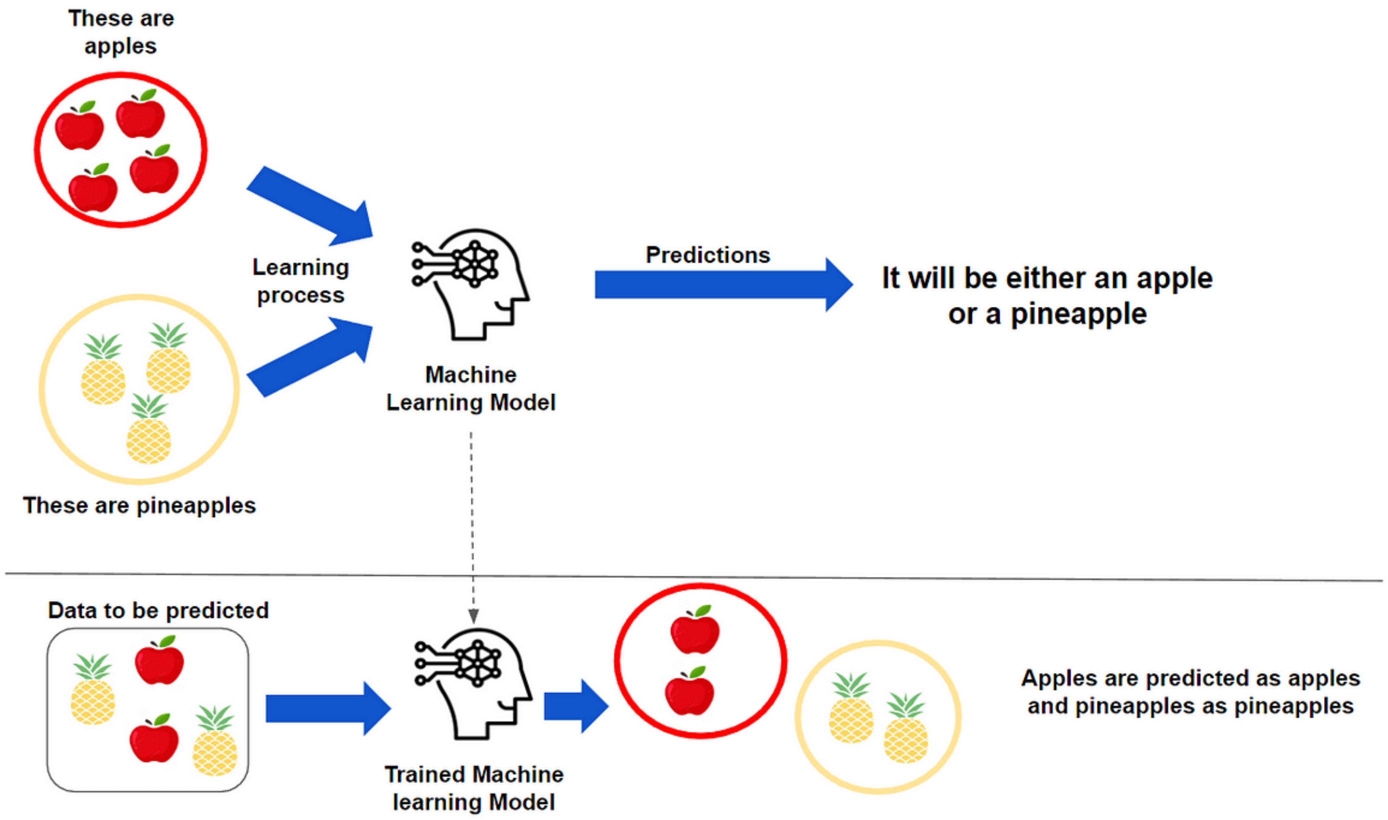
  ✓ Common tasks include **classification** and **regression**.

Unsupervised Learning and Recommender Systems

Choosing Between Supervised Learning and Anomaly Detection

Introduction to Supervised Learning vs. Anomaly Detection

- **Supervised Learning:**



Unsupervised Learning and Recommender Systems

Choosing Between Supervised Learning and Anomaly Detection

Introduction to Supervised Learning vs. Anomaly Detection

- **Anomaly Detection:**

✓ Focuses on identifying **outliers** or **anomalies** in data that deviate significantly from the norm.

✓ Does **not require labeled data**; can be used when normal behavior is well-defined, and anomalies are rare and different.

✓ Common tasks include **fraud detection, network intrusion detection,** and **fault detection**.

Unsupervised Learning and Recommender Systems

Choosing Between Supervised Learning and Anomaly Detection

Introduction to Supervised Learning vs. Anomaly Detection

- **Anomaly Detection:**

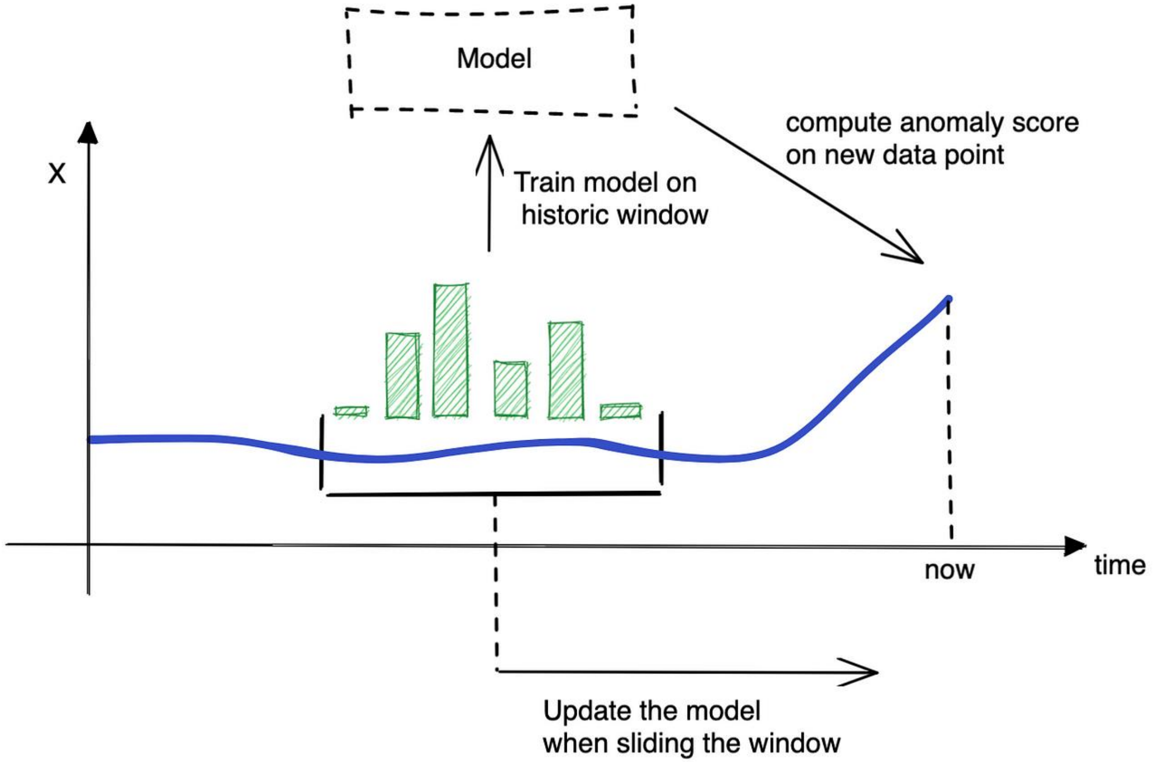


Image Source : https://towardsdatascience.com/real-time-anomaly-detection-with-python-36e3455e84e2

Unsupervised Learning and Recommender Systems

Choosing Between Supervised Learning and Anomaly Detection

Differences Between Supervised Learning and Anomaly Detection

Criteria	Supervised Learning	Anomaly Detection
Data Requirements	Requires a large amount of labeled data with normal and abnormal classes.	Requires mostly normal data; anomalies are rare and do not need explicit labeling.
Goal	Learn a mapping function from input to output for future predictions.	Identify rare and significant deviations from normal patterns.
Model Output	Predicted class labels or continuous values.	Binary output (anomaly or normal) or anomaly score.
Application Areas	Spam detection, image recognition, customer segmentation.	Fraud detection, network security, fault detection in machinery.

Unsupervised Learning and Recommender Systems

Example Scenarios for Decision Making

- **Scenario 1: Network Security Monitoring**
  - ✓ **Context:** Large volumes of network traffic data with rare intrusion attempts.
  - ✓ **Approach:** Anomaly detection to identify suspicious activities without requiring labeled data for every type of intrusion.
- **Scenario 2: Email Spam Filtering**
  - ✓ **Context:** A large dataset of emails with labeled categories (spam or not spam).
  - ✓ **Approach:** Supervised learning to classify incoming emails based on patterns learned from labeled data.

Recommender Systems

Unsupervised Learning and Recommender Systems

Recommender System

- **Introduction to Recommender Systems**
  - ✓ Recommender systems are algorithms designed to suggest relevant items to users, such as movies, products, or news articles, based on their preferences and behaviors.
- **Importance:**
  - ✓ Enhance user experience by providing personalized recommendations.
  - ✓ Drive user engagement, retention, and sales in various industries, such as e-commerce, streaming services, and social media.

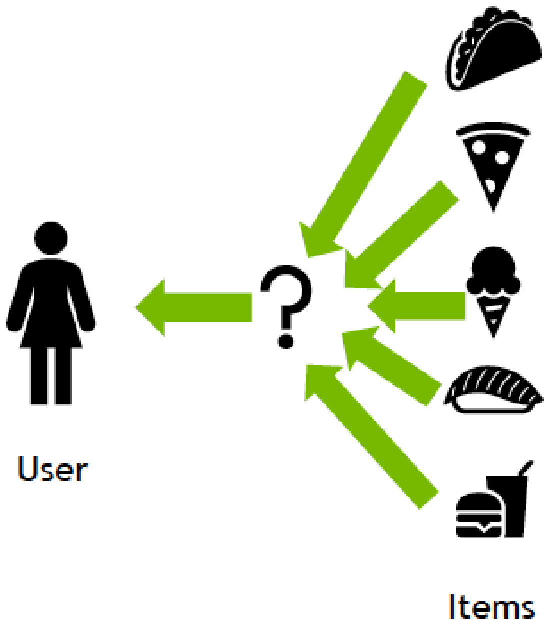


Image Source : https://developer.nvidia.com/blog/how-to-build-a-winning-recommendation-system-part-1/

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

Introduction to Recommender Systems

✓ Recommender systems are algorithms aimed at suggesting relevant items to users (e.g., movies, books, products).

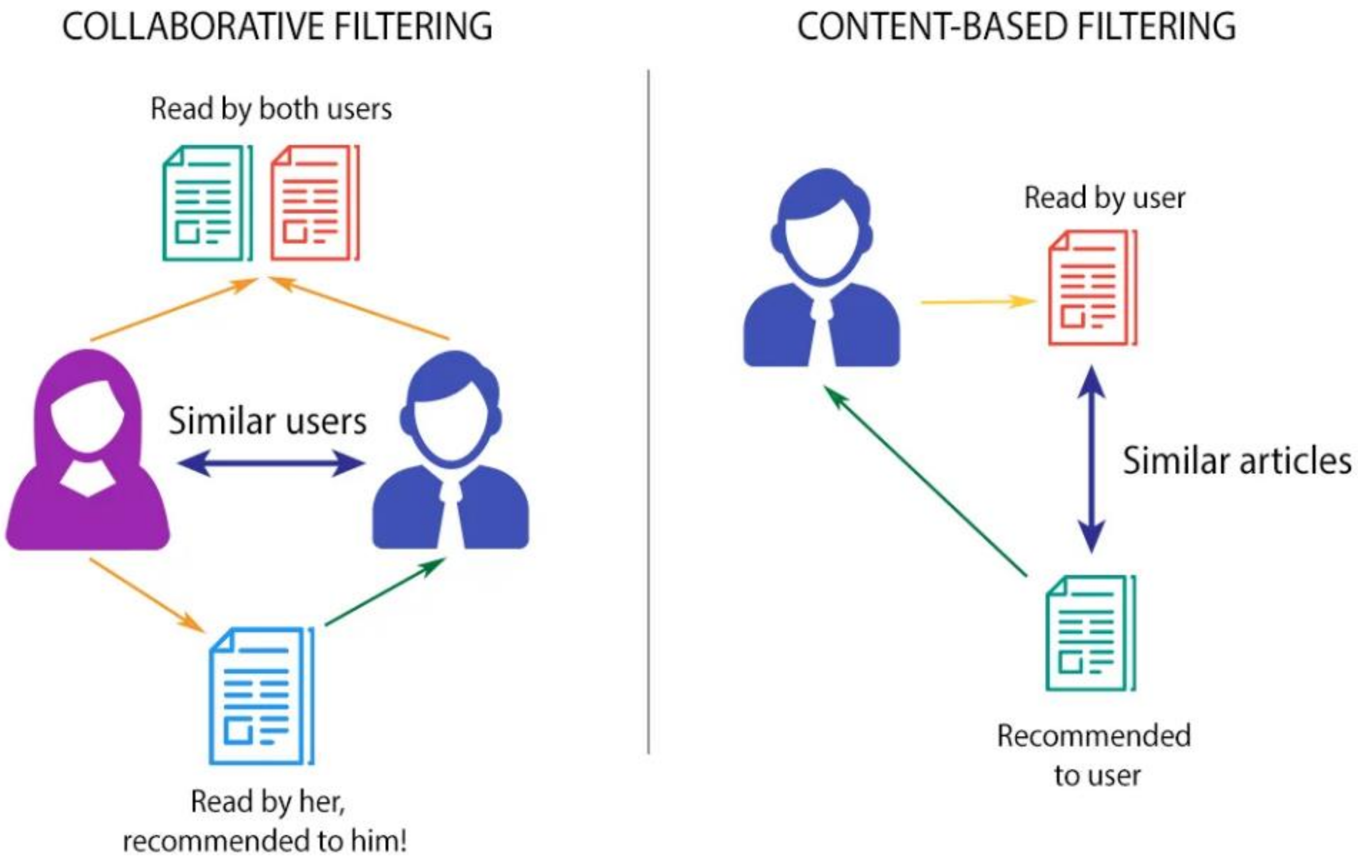
Two primary approaches:

a. **Collaborative based**

b. **Content based**

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning



Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content

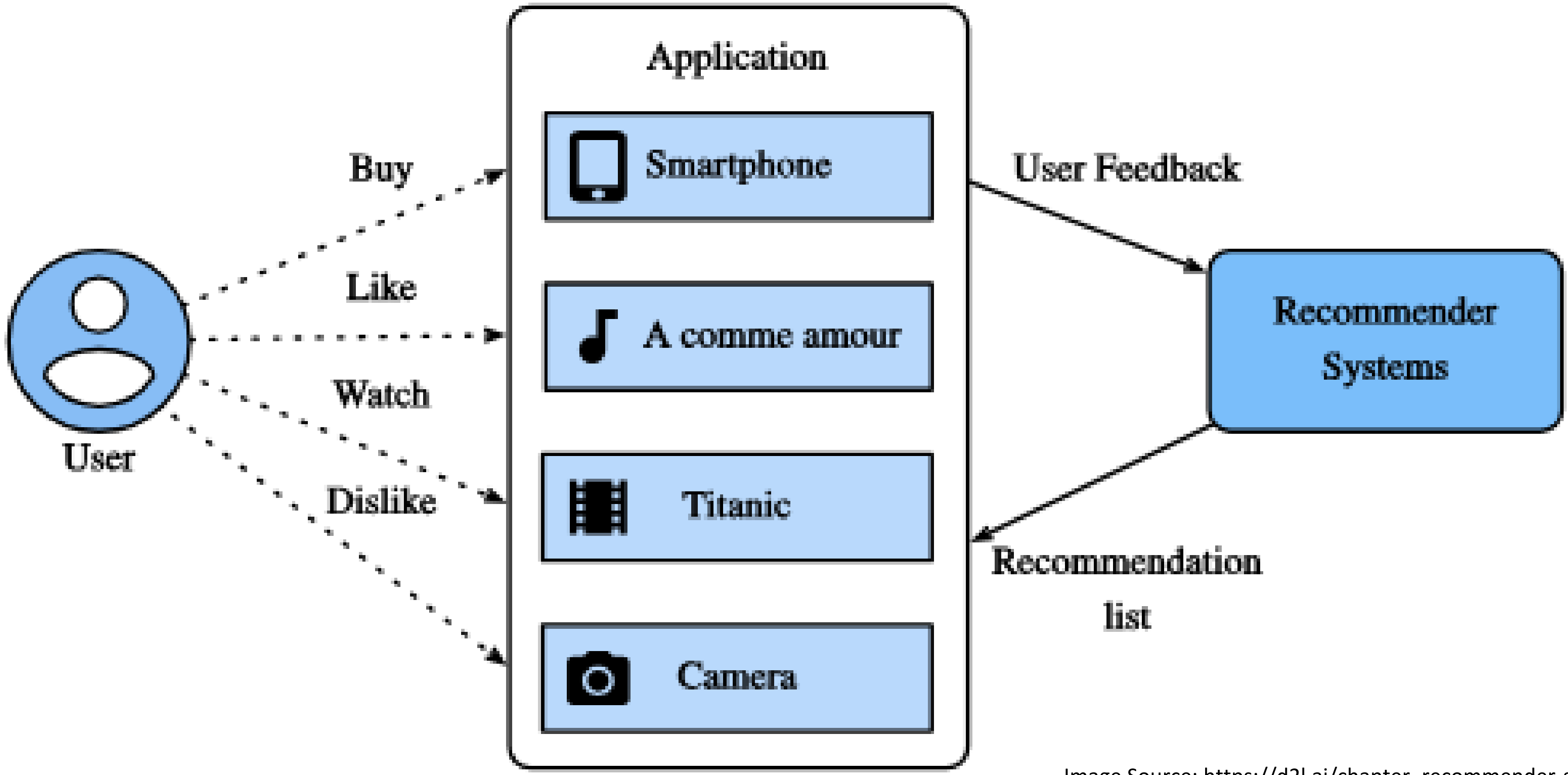
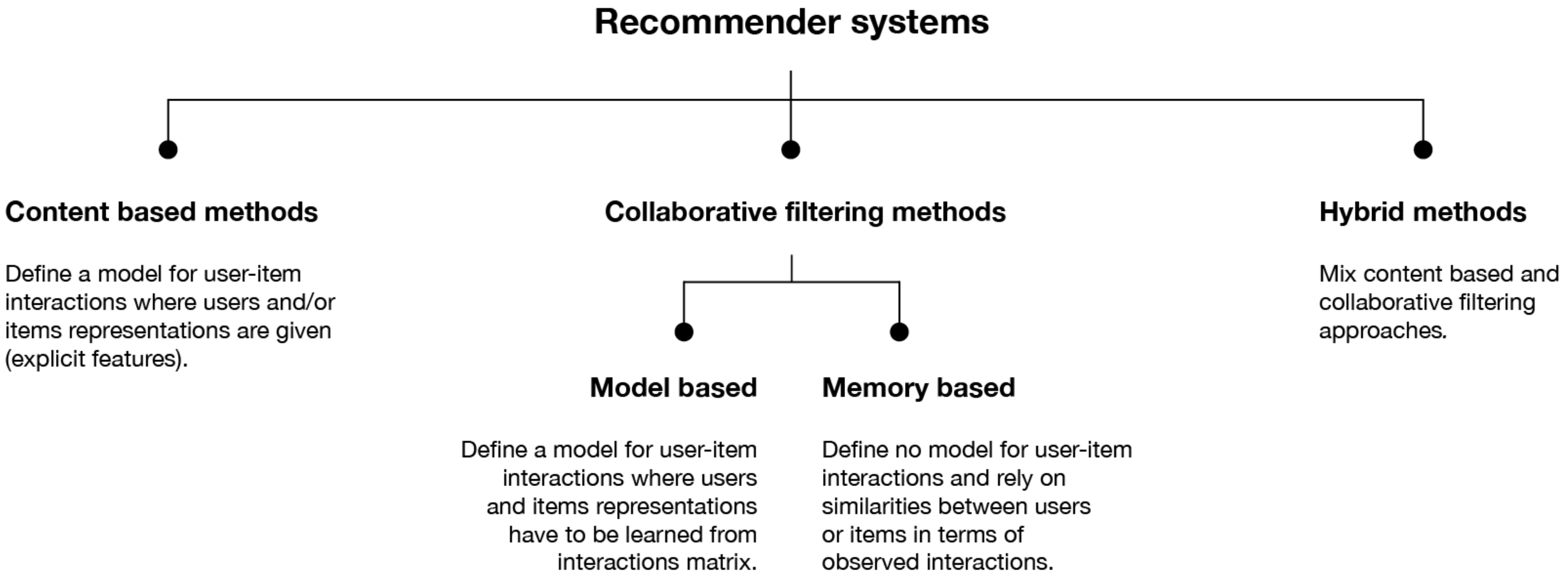


Image Source: https://d2l.ai/chapter_recommender-systems/recsys-intro.html

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning



Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering

- **Collaborative Filtering Approach**

  Collaborative filtering predicts user preferences based on the preferences of similar users or items.

  ❖ **Types of Collaborative Filtering:**

  - ✓ **User-Based Filtering:** Recommends items based on the similarity between users.
  - ✓ **Item-Based Filtering:** Recommends items based on the similarity between items.

  ❖ **Advantages:**

  - ✓ Simple and effective for large datasets.
  - ✓ Requires no domain knowledge.

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

- **Collaborative Filtering:**

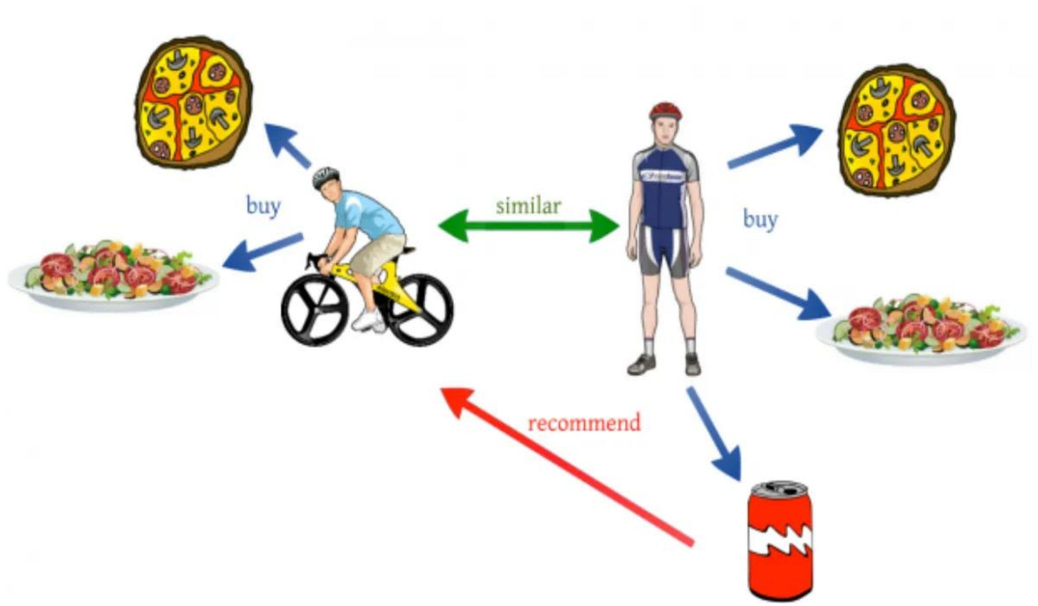
  ✓ **Challenges:**

  - **Cold start problem:** Difficulties in recommending for new users or new items.
  - **Sparsity:** Large user-item matrices are often sparse, making similarity calculation challenging.
  - **Scalability:** May not scale well with massive datasets.

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

- **Collaborative Filtering Approach – Example:**



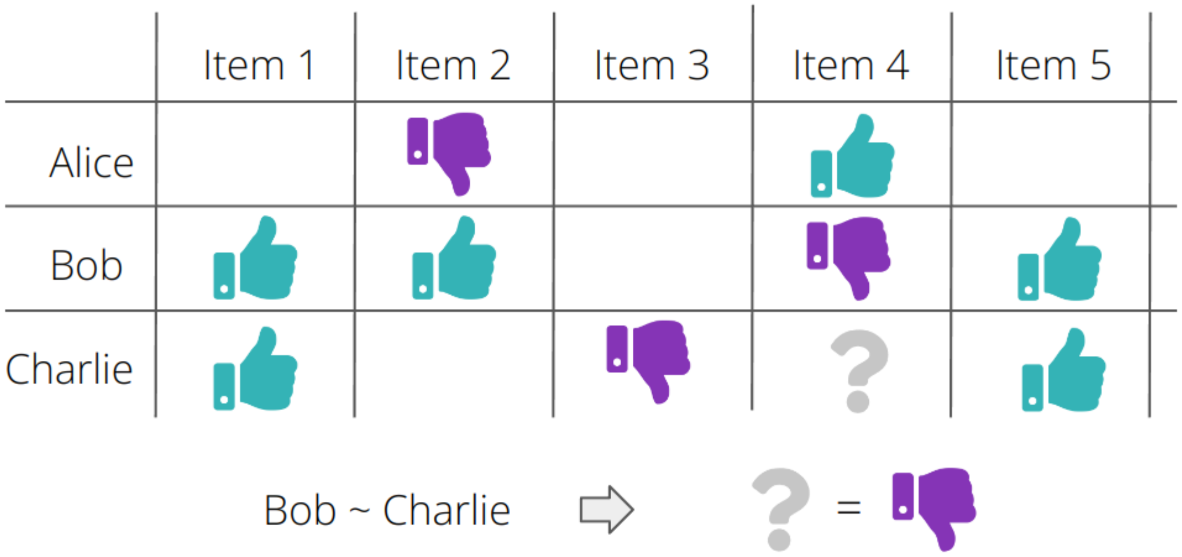


Image 1 Source: https://towardsdatascience.com/getting-started-with-recommender-systems-and-tensorrec-8f50a9943eef
Image 2 Source: https://dida.do/blog/collaborative-filtering

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

- **Types of Collaborative Filtering**

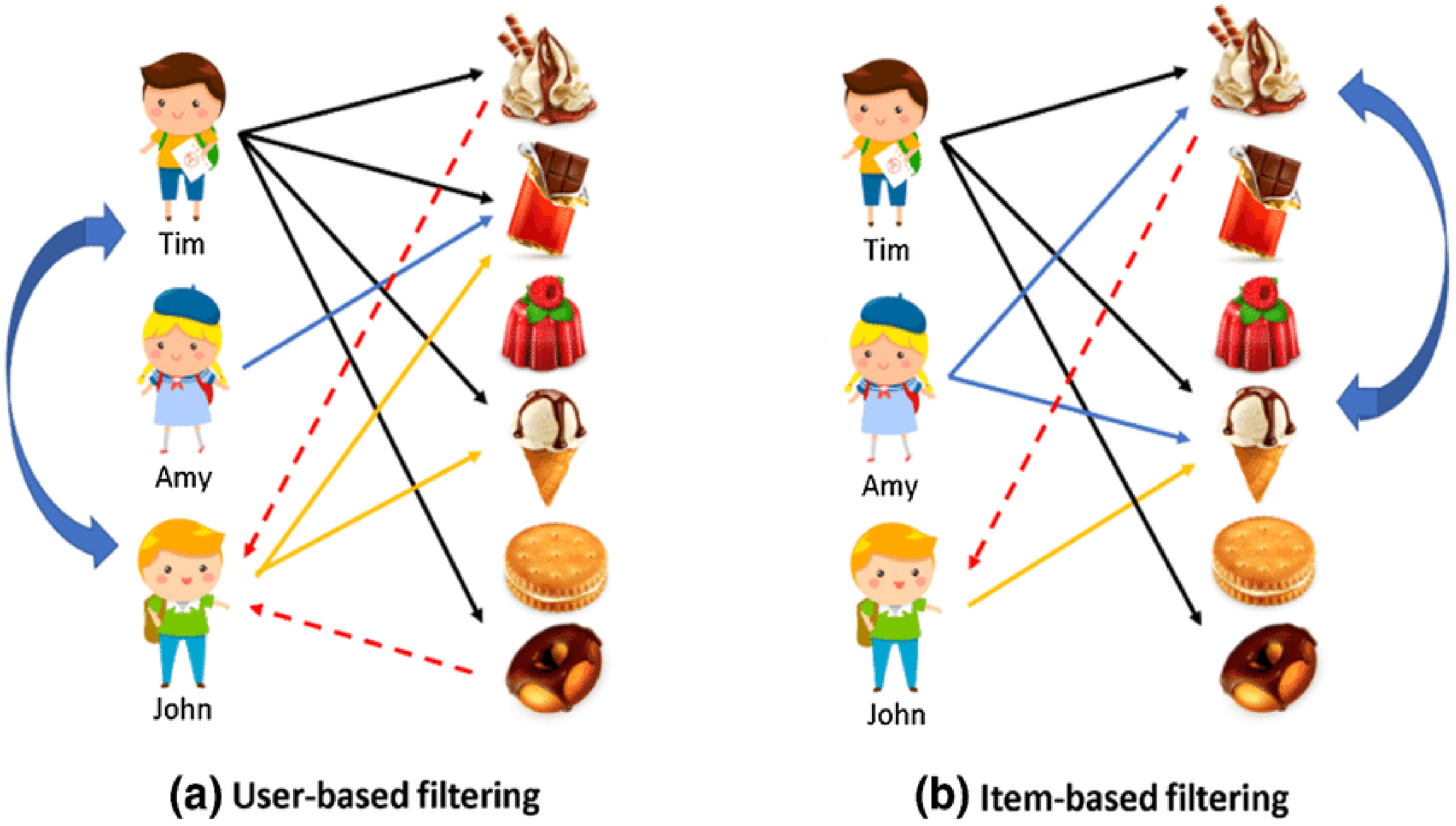


Image Source : https://www.researchgate.net/figure/Concepts-of-user-based-and-item-based-filtering_fig1_340361119

Unsupervised Learning and Recommender Systems

Collaborative Filtering – Using Singular Value Decomposition

- **User – Item Rating Matrix**

Name	Matrix	Alien	Star Wars	Casablanca	Titanic
Joe	1	1	1	0	0
Jim	3	3	3	0	0
John	4	4	4	0	0
Jack	5	5	5	0	0
Jill	0	2	0	4	4
Jenny	0	0	0	5	5
Jane	0	1	0	2	2

Apply SVD

$$\underbrace{\begin{bmatrix} .13 & .02 & -.01 \\ .41 & .07 & -.03 \\ .55 & .09 & -.04 \\ .68 & .11 & -.05 \\ .15 & -.59 & .65 \\ .07 & -.73 & -.67 \\ .07 & -.29 & .32 \end{bmatrix}}_{U} \underbrace{\begin{bmatrix} 12.4 & 0 & 0 \\ 0 & 9.5 & 0 \\ 0 & 0 & 1.3 \end{bmatrix}}_{\Sigma} \underbrace{\begin{bmatrix} .56 & .59 & .56 & .09 & .09 \\ .12 & -.02 & .12 & -.69 & -.69 \\ .40 & -.80 & .40 & .09 & .09 \end{bmatrix}}_{V^T}$$

U → User to Concept matrix
V → Concept to Item matrix
∑ → Singular value (weight of concepts)

Unsupervised Learning and Recommender Systems

Predicting Rating with SVD components

Will Jenny like the Movie Alien ????

Rating (by Jenny to Alien) = U row Jenny * VT column of Alien

Rating (by Jenny to Alien) = [0.07 -0.73 -0.67] * Transpose([0.59 -0.02 -0.8])
= 0.59 ~=1

Model rating prediction value indicating that She hate Alien.

We should NOT recommend Alien to her !!!!

Unsupervised Learning and Recommender Systems

Recommender System Using Collaborative Filtering

- **How Collaborative Filtering Works..**

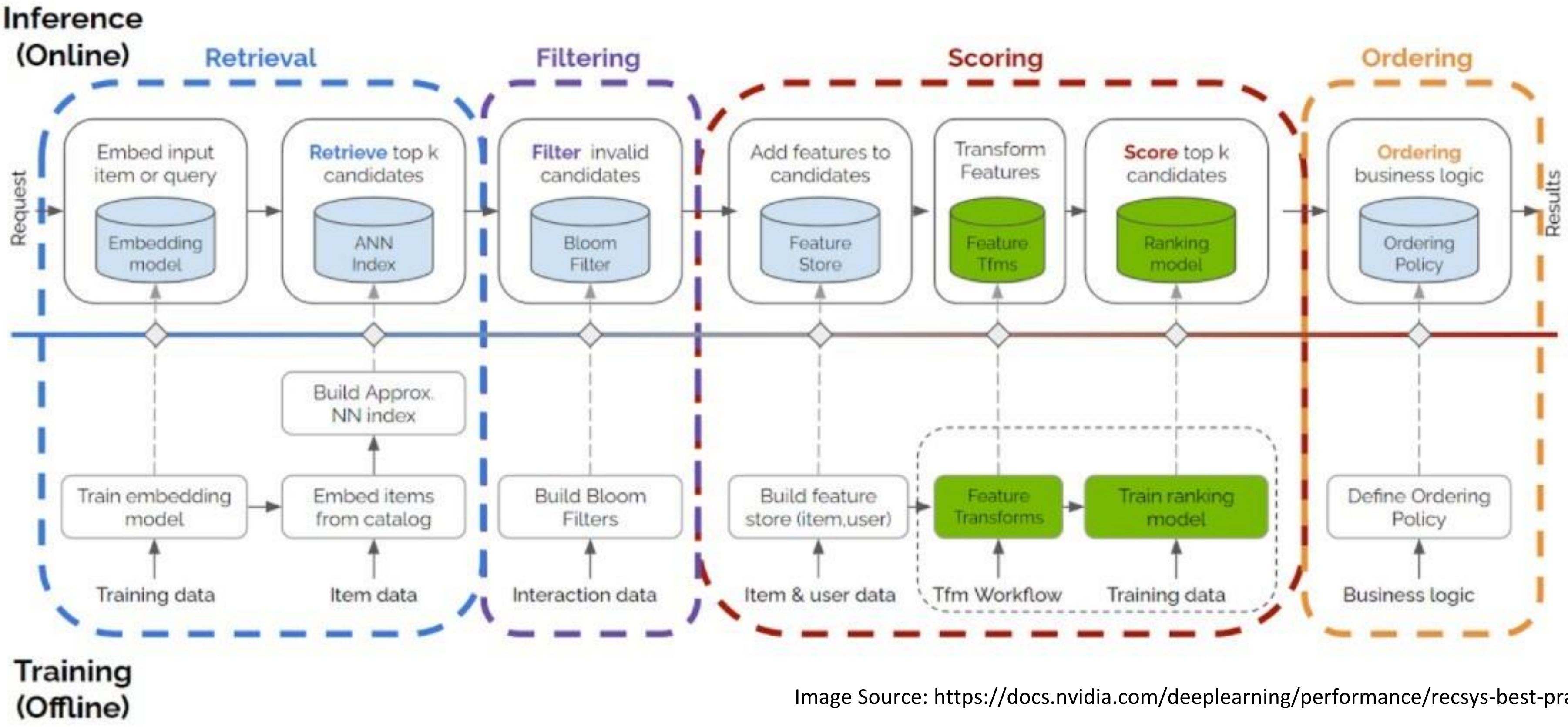


Image Source: https://docs.nvidia.com/deeplearning/performance/recsys-best-practices/index.html

Unsupervised Learning and Recommender Systems

Recommender System Using Collaborative Filtering

- **Similarity Calculation Example:**

  ✓ **Cosine Similarity** formula between two users (A and B):

$$\text{Similarity} = \frac{\sum(A_i \times B_i)}{\sqrt{\sum(A_i^2)} \times \sqrt{\sum(B_i^2)}}$$

  ✓ **Pearson Correlation** coefficient between two users (A and B):

$$\text{Correlation} = \frac{\sum(A_i - \bar{A}) \times (B_i - \bar{B})}{\sqrt{\sum(A_i - \bar{A})^2} \times \sqrt{\sum(B_i - \bar{B})^2}}$$

Unsupervised Learning and Recommender Systems

Recommender System Using Collaborative Filtering

- **Advantages of Collaborative Filtering**

  - **No Need for Domain Knowledge:**
    - ✓ Does not require any information about the items; works solely based on user behavior.
  - **Adaptability:**
    - ✓ Can adapt to user preferences as they change over time, providing more accurate and personalized recommendations.
  - **Scalability:**
    - ✓ Effective for large datasets with many users and items.

Unsupervised Learning and Recommender Systems

Recommender System Using Collaborative Filtering

- **Challenges of Collaborative Filtering**

  - **Cold Start Problem:**
    - ✓ Difficulty in recommending items to new users (new user problem) or recommending new items to users (new item problem) due to lack of sufficient data.
  - **Sparsity:**
    - ✓ In large datasets, users rate only a small subset of items, leading to a sparse user-item matrix and difficulty in finding similar users/items.
  - **Scalability:**
    - ✓ With millions of users and items, computing similarity scores and predictions can become computationally expensive.

Recommender System
Using a Content-Based Deep Learning Method

Unsupervised Learning and Recommender Systems

Recommender System Using a Content-Based Deep Learning Method

- **Introduction to Content-Based Recommender Systems**
  - ✓ Content-based recommender systems suggest items to users based on the attributes of items and the user’s past preferences.
- **What is a Content-Based Deep Learning Recommender System?**
  - ✓ Combines traditional content-based filtering techniques with deep learning models to improve recommendation accuracy and capture complex patterns in user preferences.
  - ✓ Utilizes deep neural networks (DNNs) to learn high-level features and representations from raw content data (e.g., text, images, audio).

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

Content-Based Filtering with Deep Learning

Content-based filtering recommends items by analyzing the content of items and user profiles using deep learning techniques.

- ✓ Uses item features (e.g., text, images, metadata) and user profiles to make recommendations.
- ✓ Employs deep learning models (e.g., Convolutional Neural Networks (CNNs) for images, Recurrent Neural Networks (RNNs) for sequences).

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

Content-Based Filtering with Deep Learning...

- ✓ **Approach:** Uses item features (e.g., text, images, metadata) and user profiles to make recommendations.
- ✓ Employs deep learning models (e.g., Convolutional Neural Networks (CNNs) for images, Recurrent Neural Networks (RNNs) for sequences).

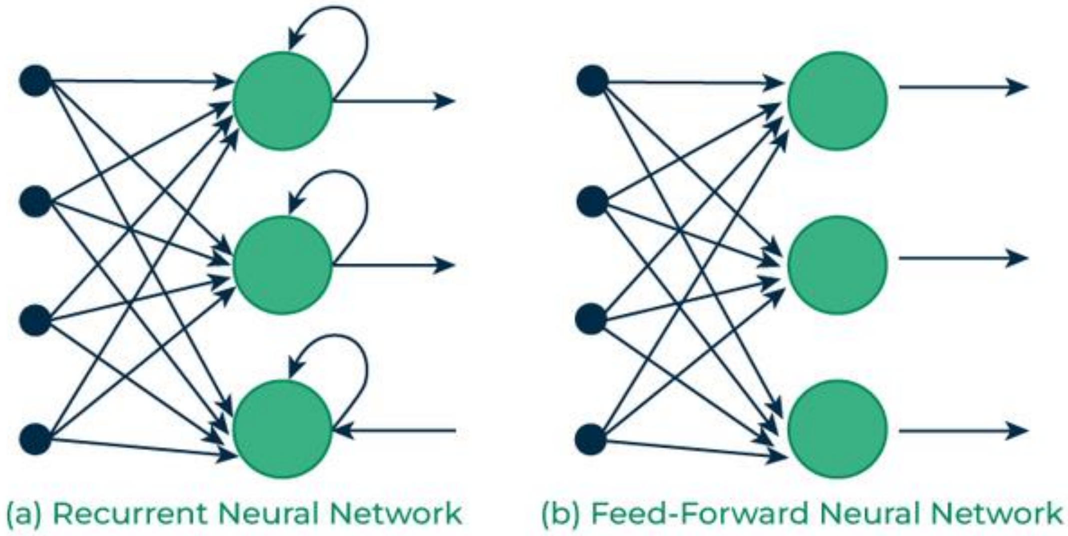


Image Source: https://www.geeksforgeeks.org/introduction-to-recurrent-neural-network/

Unsupervised Learning and Recommender Systems

Deep Learning Techniques Used in Content-Based Recommender Systems

- **Convolutional Neural Networks (CNNs):**
  - ✓ Effective for processing and learning features from visual content (e.g., images).
  - ✓ Example: A CNN can be used to analyze the visual style of movies or fashion products.

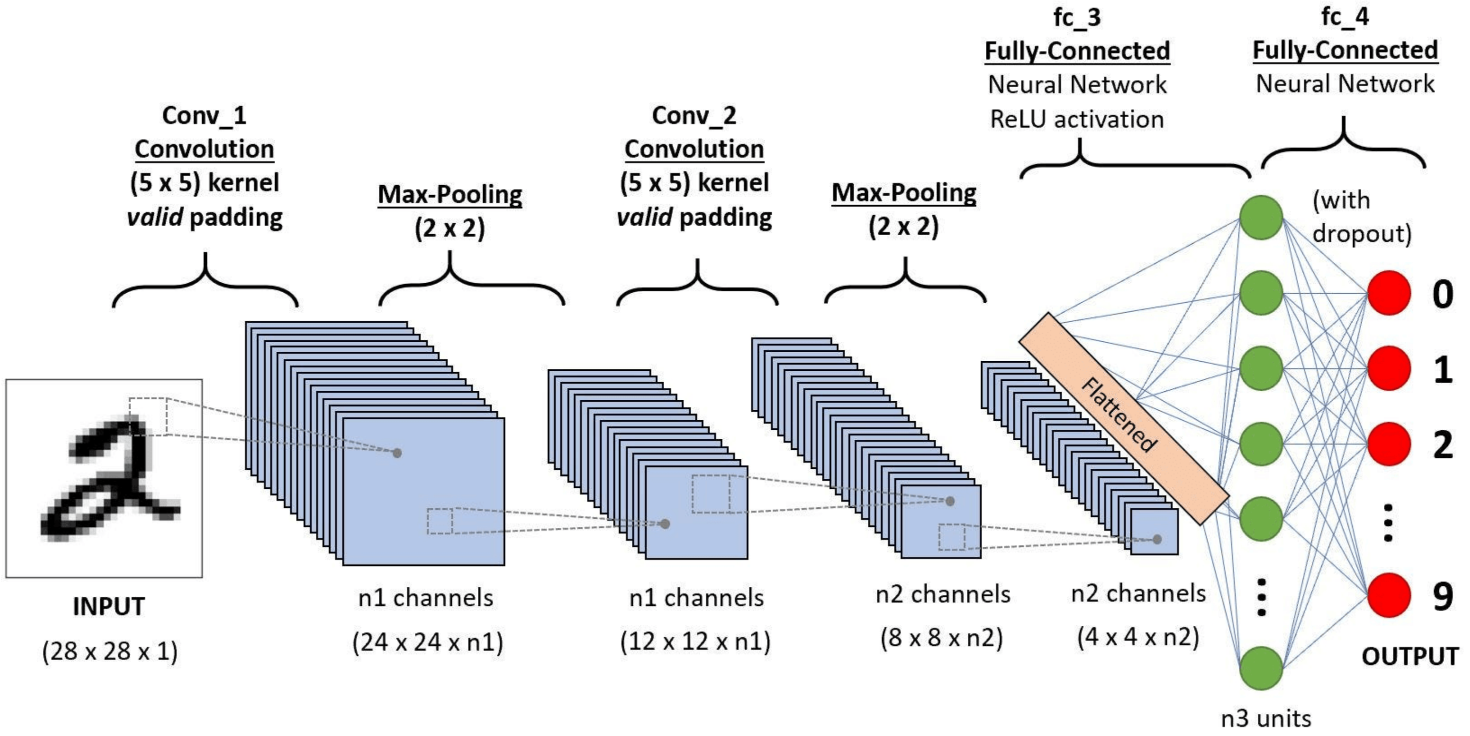


Image Source: https://dev.to/saidi-souhaieb/what-are-convolutional-neural-networks-cnn-the-art-of-computer-vision-for-beginners-34nn

Unsupervised Learning and Recommender Systems

Deep Learning Techniques Used in Content-Based Recommender Systems

- **Recurrent Neural Networks (RNNs):**
  - ✓ Suitable for sequential data, such as text or time-series data.
  - ✓ Example: An RNN can be used to analyze user reviews or descriptions of items to extract meaningful features.

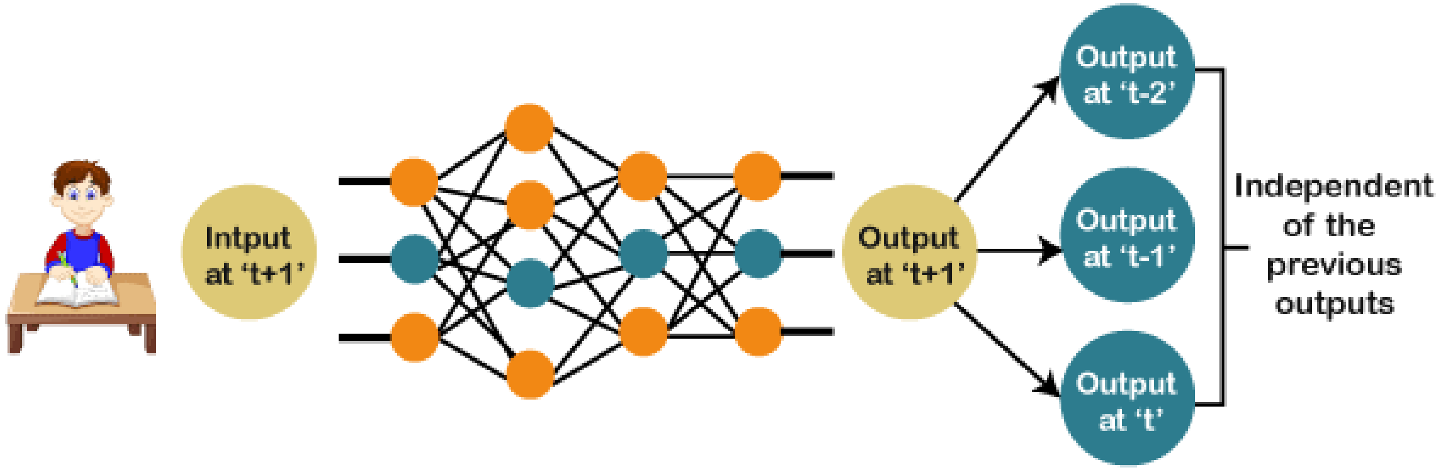


Image Source: https://www.linkedin.com/posts/sandeep-sharma-00389755_ml-dl-ai-activity-7212787087639498753-jnBI

Unsupervised Learning and Recommender Systems

Recommender System Using a Content-Based Deep Learning Method

- **Autoencoders:**
  - ✓ Unsupervised learning technique used to compress and encode item features into a lower-dimensional space.
  - ✓ Helps in learning compact representations that capture the most relevant information about items.

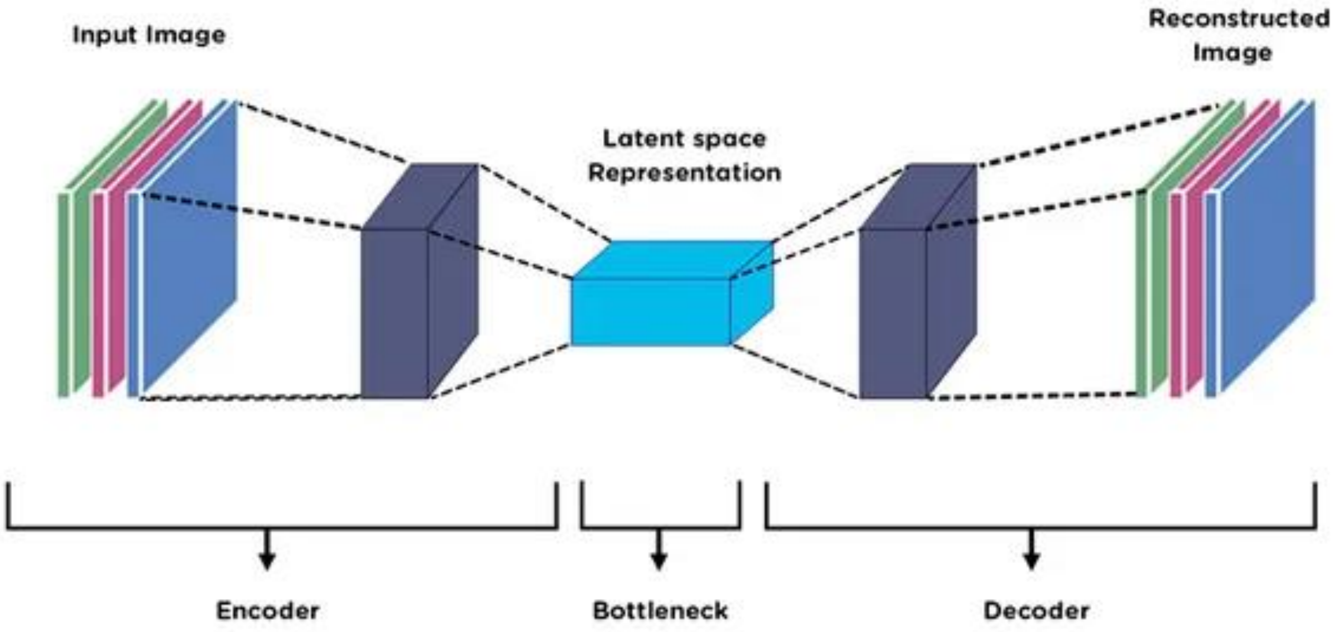


Image Source: https://arxiv.org/html/2405.00142v1

Unsupervised Learning and Recommender Systems

Recommender System Using a Content-Based Deep Learning Method

- **Transformers:**
  - ✓ Used for modeling long-range dependencies in sequential data, particularly useful for natural language processing tasks.
  - ✓ Example: Transformers can enhance text-based recommendations by understanding complex contextual information from descriptions or reviews.

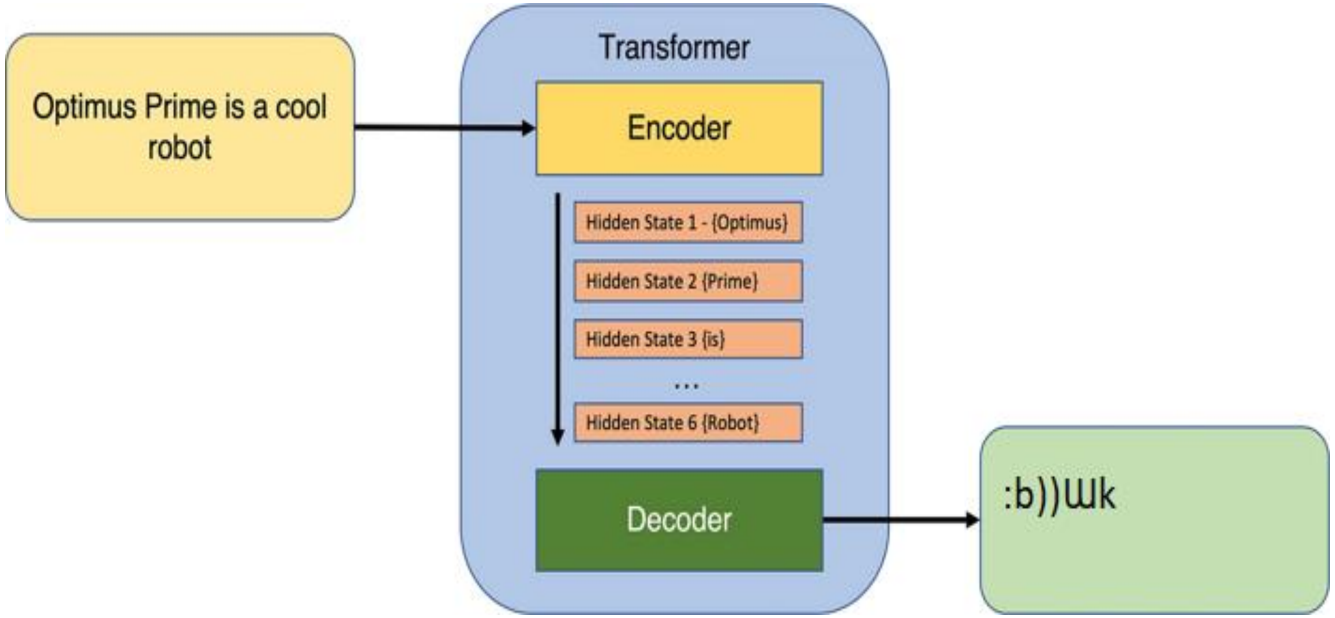


Image Source: https://mpost.io/top-30-transformer-models-in-ai-what-they-are-and-how-they-work/

Unsupervised Learning and Recommender Systems

Recommender System Using a Content-Based Deep Learning Method

- **Advantages of Content-Based Deep Learning Methods**

  - **High Dimensional Feature Learning:**
    - ✓ Can learn complex, non-linear relationships between user preferences and item attributes.
  - **Personalization:**
    - ✓ Offers personalized recommendations by adapting to individual user preferences over time.
  - **No Cold Start for Items:**
    - ✓ Can recommend new items based on their content features without requiring prior user interaction data.
  - **Improved Scalability:**
    - ✓ Suitable for large-scale datasets and can handle a variety of data types (text, image, video, etc.).

Unsupervised Learning and Recommender Systems

Recommender System Using a Content-Based Deep Learning Method

- **Challenges and Limitations**
  - **Cold Start Problem for Users:**
    - ✓ Requires sufficient user data to accurately learn preferences, making it challenging to recommend items to new users.
  - **Content Dependency:**
    - ✓ The quality of recommendations is highly dependent on the quality and completeness of item content data.
  - **High Computational Cost:**
    - ✓ Training deep learning models is computationally expensive and requires significant resources, especially with large datasets.
  - **Overfitting:**
    - ✓ Deep learning models can be overfit to the training data, particularly when the dataset is small or imbalanced.

Unsupervised Learning and Recommender Systems

Recommender System Using a Content-Based Deep Learning Method

- **Tools and Frameworks for Implementation**

  - **Python Libraries:**

✓ **TensorFlow** and **Keras:** Popular libraries for building and training deep learning models.

✓ **PyTorch:** Widely used for deep learning tasks, offering flexibility and ease of use.

✓ **Hugging Face Transformers:** For state-of-the-art natural language processing models.

✓ **spaCy and NLTK:** Libraries for natural language processing and text preprocessing.

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

Hybrid Recommender Systems

✓ Combines both collaborative filtering and content-based approaches to leverage the strengths of both methods.

Benefits:

- Overcomes the limitations of each individual approach.
- Provides more accurate and diverse recommendations.

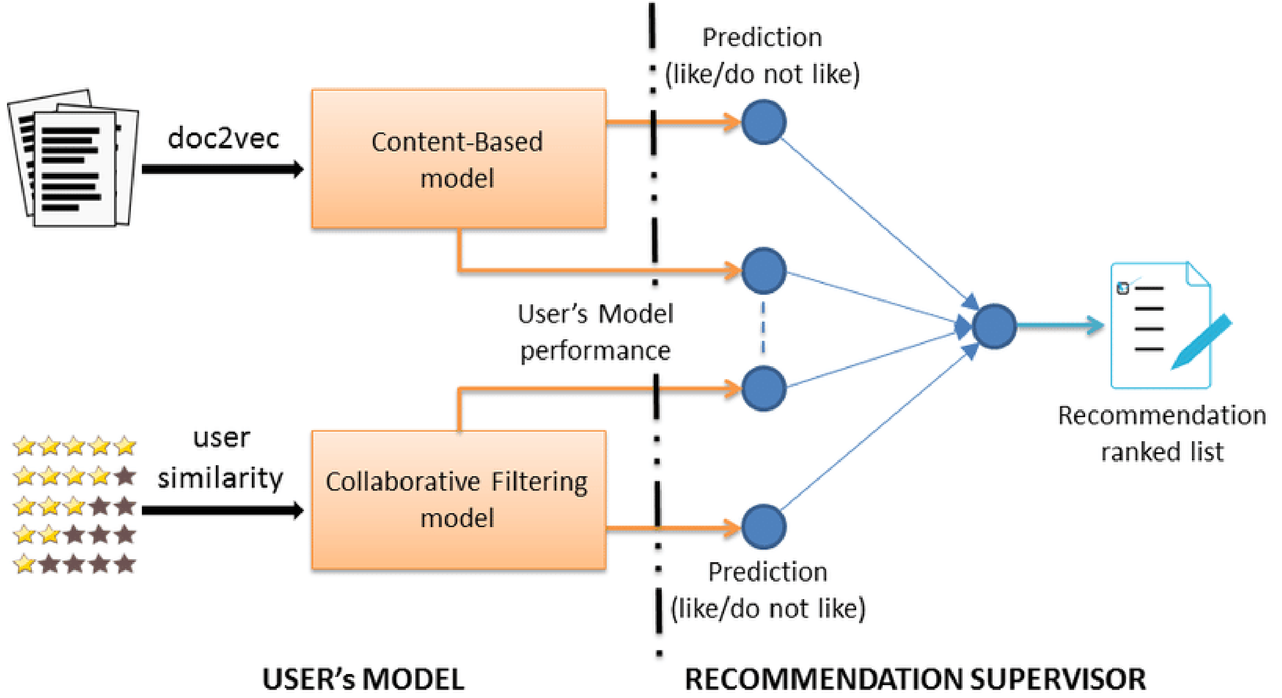


Image Source: https://www.researchgate.net/figure/Hybrid-recommendation-model_fig1_319045879

Unsupervised Learning and Recommender Systems

Recommender Systems: Collaborative Filtering and Content-Based Deep Learning

Key Takeaways:

- ✓ Collaborative filtering relies on user-item interactions but struggles with new data (cold start).
- ✓ Content-based filtering with deep learning leverages item content, solving the cold start problem and providing more nuanced recommendations.
- ✓ Combining both methods in a hybrid system can lead to better performance and a more personalized user experience.

Image Source : https://www.flaticon.com/free-icon/take-away_3595217

Reinforcement Learning Model

Reinforcement Learning

- Reinforcement learning (RL) is a type of machine learning where an agent learns to make decisions by interacting with an environment.

- The goal of the agent is to take actions that maximize cumulative rewards over time.

- RL works on a trial-and-error basis, where the agent tries different actions, receives feedback in the form of rewards or penalties, and uses this feedback to improve its future decisions.

Reinforcement Learning

Key Concepts/Terminology

- ✓ **Agent**: The decision-maker or learner.
- ✓ **Environment**: The world the agent interacts with.
- ✓ **Action**: Choices the agent can make.
- ✓ **Reward**: Feedback from the environment based on the agent's action.
- ✓ **State**: The current situation or condition the agent is in.
- ✓ **Policy**: A strategy used by the agent to determine its actions based on the current state.
  - **The agent’s objective is to learn an optimal policy that maximizes the total reward over time**

Reinforcement Learning

Imagine you're playing a video game where you're controlling a character to collect as many coins as possible while avoiding obstacles.

- **You (the player)** are the **agent**.
- The **game world** is the **environment**.
- Every time you press a button to move left, right, or jump, that's an **action**.
- When you collect a coin, you get points (a **reward**), and when you hit an obstacle, you lose points (a **penalty**).

At first, you don't know the best way to play, so you try different actions. Sometimes you succeed and collect coins, sometimes you fail and lose points. Over time, you start to figure out which actions (like jumping at the right time) help you get more coins and which ones make you lose points.

This process of learning by **trying things out and improving based on rewards and penalties** is what reinforcement learning is all about!

The goal is to keep learning the best moves to get the highest score, just like how an agent in reinforcement learning learns to make the best decisions to maximize its rewards.

Reinforcement Learning

Key Algorithms

- Q-Learning
- Deep Q-Networks (DQN)
- Policy Gradient Methods
- Actor-Critic Methods
- Monte Carlo Methods

Reinforcement Learning

Q Learning

https://www.geeksforgeeks.org/q-learning-in-python/

Reinforcement Learning

Q Learning

Q-learning is a method where an agent learns the best actions to take in a given environment by **trying out different actions** and **learning from the results**.

- A model-free algorithm where the agent learns the value of actions (Q-values) for each state and uses this information to make decisions.

- It updates the Q-values using the equation:

$$Q(s,a) = Q(s,a) + \alpha[r + \gamma \max Q(s',a') - Q(s,a)]$$

- Used in simple environments like grid-worlds.

Reinforcement Learning

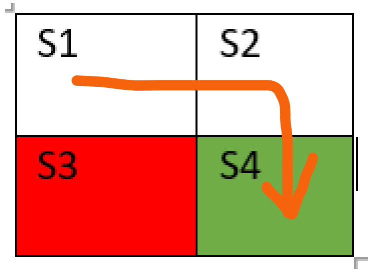
Q Learning Example

- ✓ **Environment**: A robot in a 2x2 grid world.
- ✓ **Goal**: The robot needs to reach the bottom-right corner (goal state) to get a reward.
- ✓ **Actions**: The robot can move **up**, **down**, **left**, or **right**.

Rewards:

- ✓ Reaching the goal (green cell) gives a reward of **+10**.
- ✓ Moving to an empty cell gives **0** reward.
- ✓ Moving to a red cell gives a **-1** penalty.

Reinforcement Learning



Initial Q value for all State-action pair is 0

State	Right	Left	Top	Down
S1	0	0	0	0
S2	0	0	0	0
S3	0	0	0	0
S4	0	0	0	0

Q value=0

Moving from S1 to S2 (S1 to right)

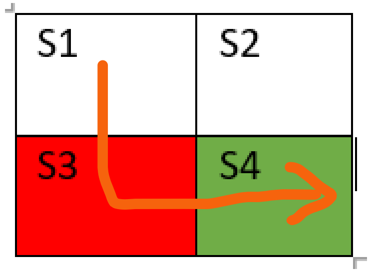
State	Right	Left	Top	Down
S1	0	0	0	0
S2	0	0	0	0
S3	0	0	0	0
S4	0	0	0	0

Updated Q value = 0

Moving from S2 to S4 (S2 to down)

State	Right	Left	Top	Down
S1	0	0	0	0
S2	0	0	0	10
S3	0	0	0	0
S4	0	0	0	0

Updated Q value =10



Initial Q value for all state action pair

State	Right	Left	Top	Down
S1	0	0	0	0
S2	0	0	0	0
S3	0	0	0	0
S4	0	0	0	0

Q value=0

Moving from S1 to S3

State	Right	Left	Top	Down
S1	0	0	0	0
S2	0	0	0	-1
S3	0	0	0	0
S4	0	0	0	0

Updated Q value = -1

Moving from S3 to S4

State	Right	Left	Top	Down
S1	0	0	0	0
S2	0	0	0	-1
S3	10	0	0	0
S4	0	0	0	0

Updated Q value =9

Path s1-s2-s4 maximizes the Q value in the process of reaching s4

Deep Reinforcement Learning Model (Deep Q Network)

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- ✓ **A Deep Q Network (DQN)** is a reinforcement learning algorithm that combines Q-learning with deep neural networks to handle environments with high-dimensional state spaces.

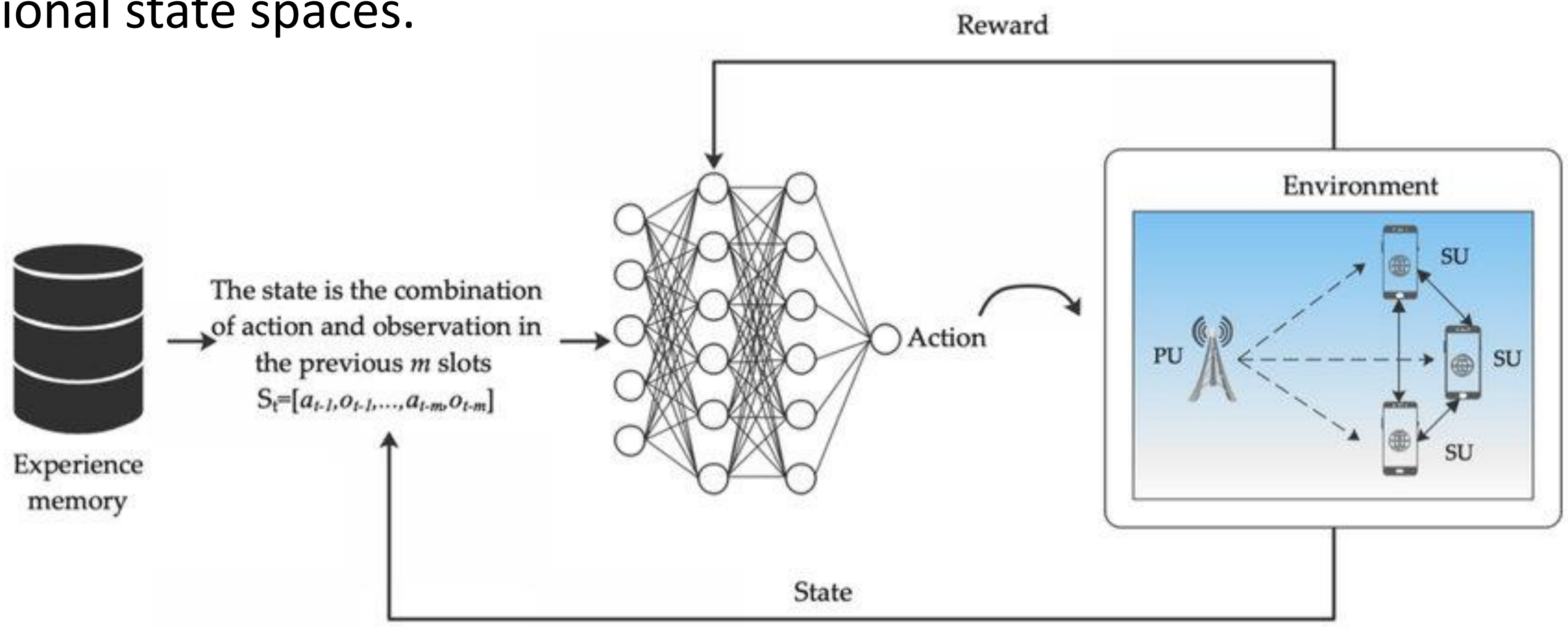


Image Source: https://www.researchgate.net/figure/The-framework-of-deep-Q-network-DQN_fig3_349500153

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

State

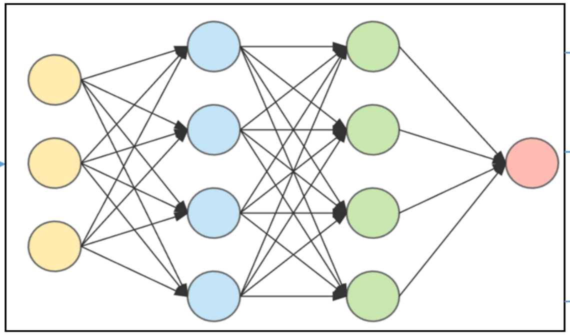
Action

Q Table	
State-Action	Value
-	0
-	0
-	0
-	0
-	0
-	0
-	0
-	0
-	0

Q-Value

Q Learning

State



Q-Value Action 1

Q-Value Action 2

-

-

Q-Value Action N

Deep Q Learning

Image Source : https://www.youtube.com/watch?app=desktop&v=bMQZjyP08Pc

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- **Key Components:**
  - ✓ **Q-Function Approximation:** Uses a neural network to approximate the Q-value function, **Q (s, a; θ)**, where **s** is the state, **a** is the action, and **θ** represents the network parameters.
  - ✓ **Replay Buffer:** Stores past experiences (state, action, reward, next state) to break correlations between sequential observations.
  - ✓ **Target Network:** A separate network used to compute the target Q-values to stabilize training.

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- **How DQN Works:**

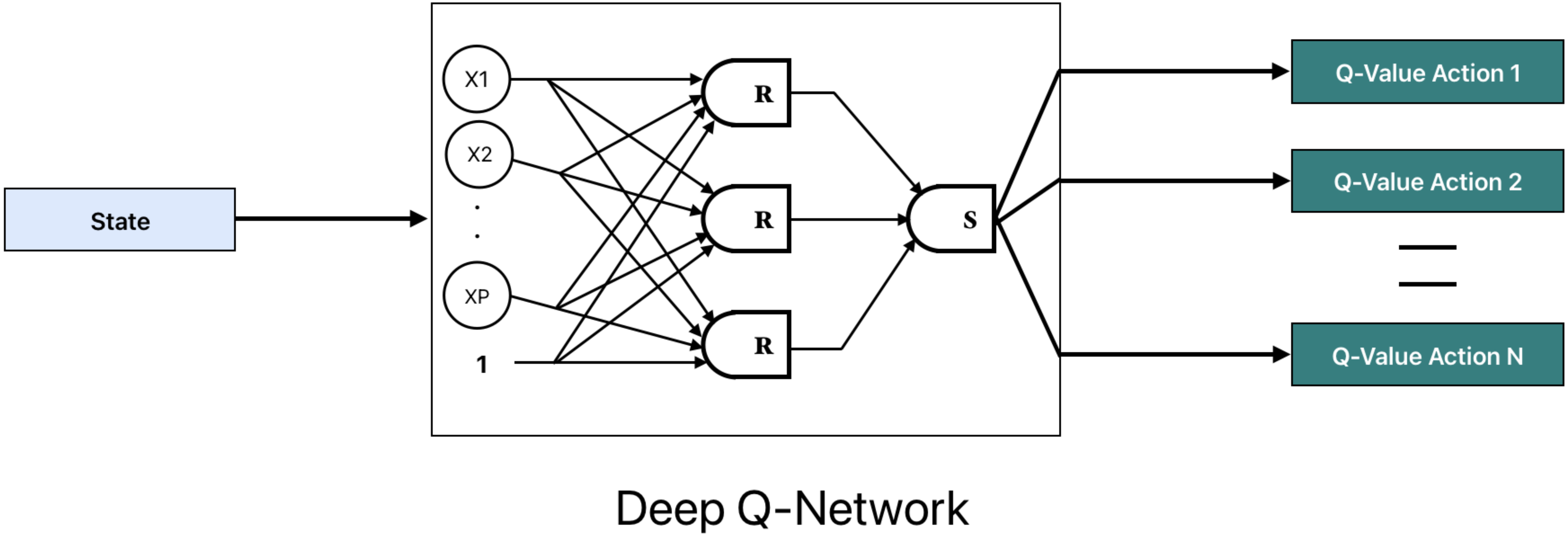


Image Source: https://www.linkedin.com/pulse/bxd-primer-series-deep-q-network-dqn-reinforcement-learning-mayank-k-

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- **How DQN Works..**

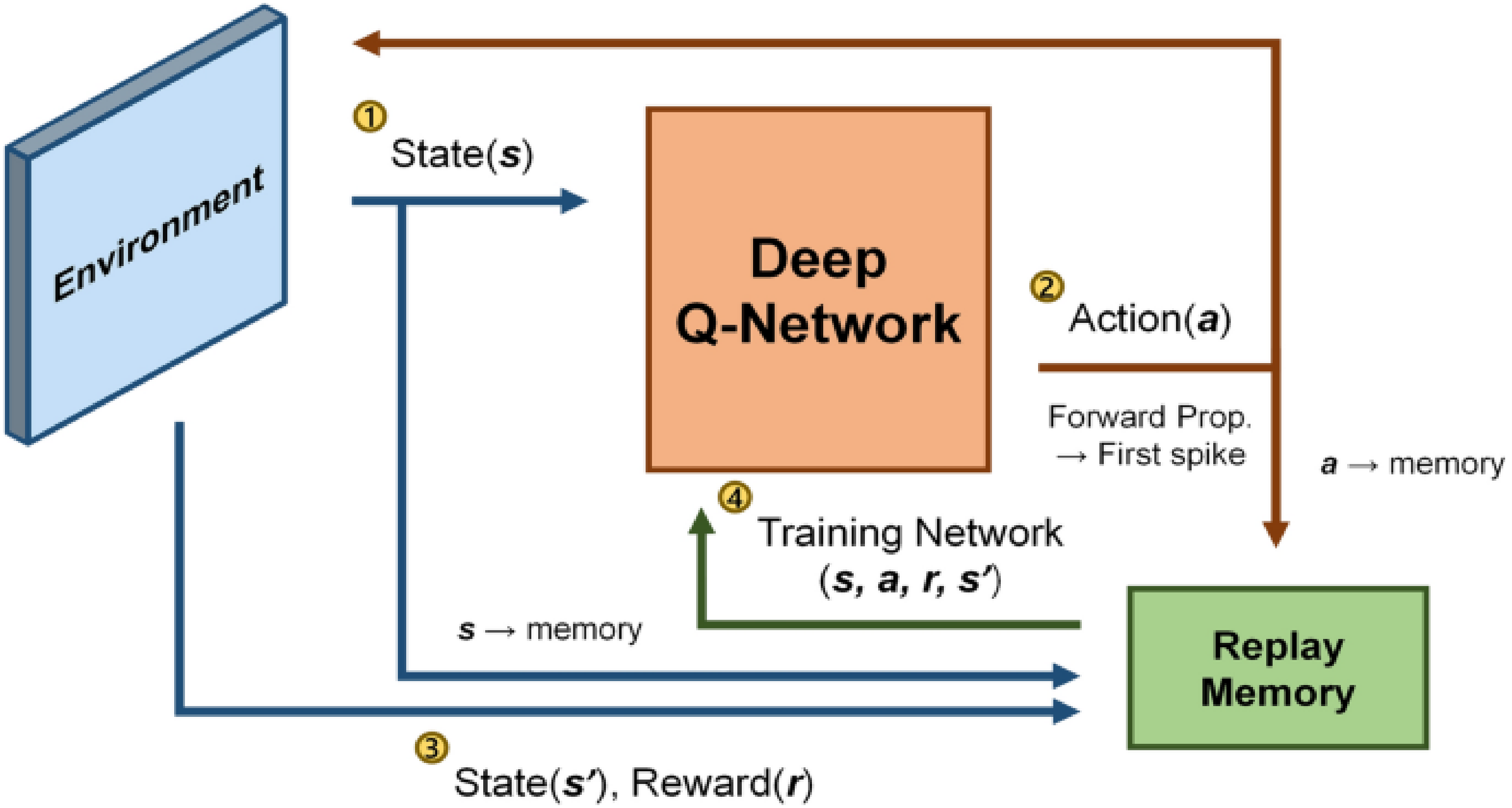


Image Source : https://www.researchgate.net/figure/Overall-training-process-of-the-hardware-based-DQN-consists-of-three-elements-and-four_fig3_349187272

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

Algorithm

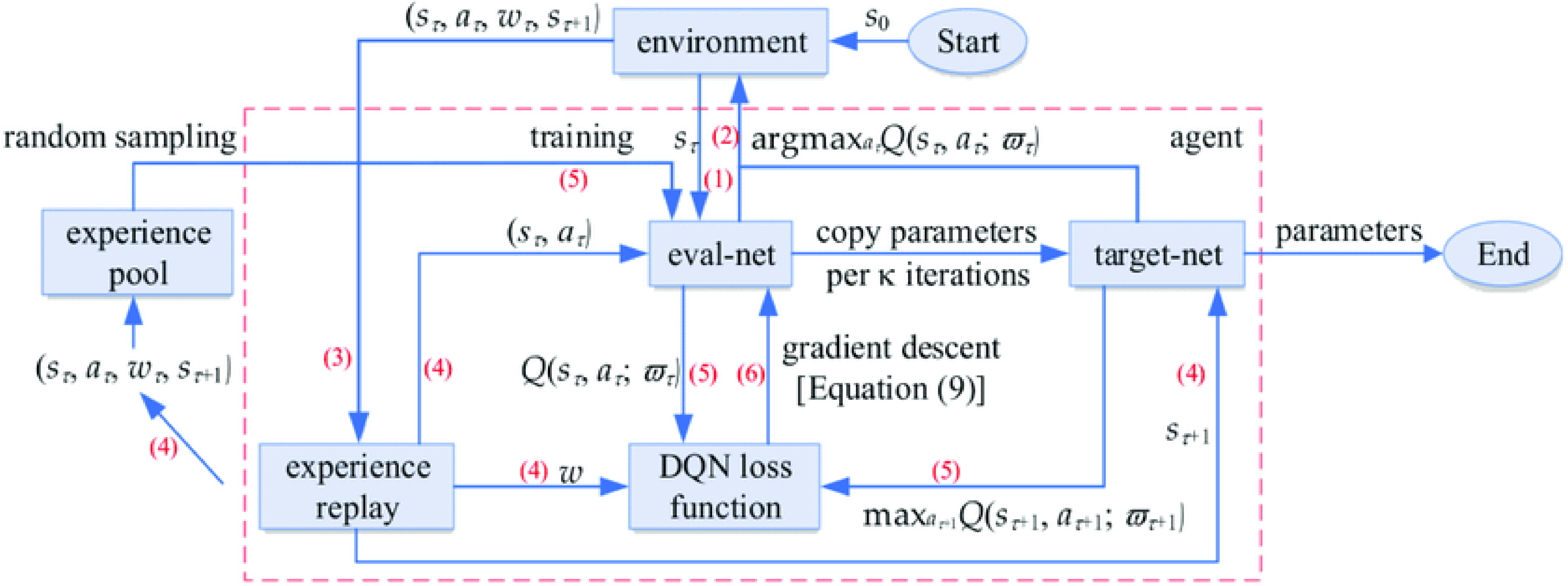


Image Source : https://www.researchgate.net/figure/Deep-Q-network-DQN-algorithm-flow-chart_fig1_350574788

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

Algorithm

1. Initialize the replay memory and the Q-network with random weights.
2. For each step in the environment:
   - Select an action a using an epsilon-greedy policy.
   - Execute action a and observe the reward r and next state s'.
   - Store the experience (s, a, r, s') in the replay buffer.
   - Sample a mini-batch of experiences from the replay buffer.
   - Update the Q-network weights by minimizing the loss:

$$\text{Loss} = \left(r + \gamma \max_{a'} Q(s', a'; \theta^{-}) - Q(s, a; \theta)\right)^2$$

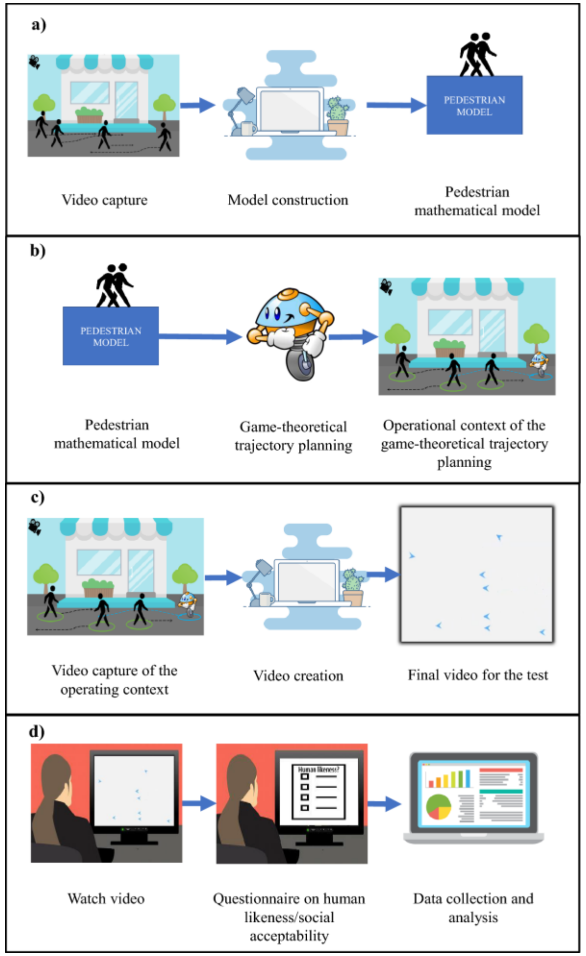
   where γ is the discount factor and θ^{-} are the target network parameters.
3. Periodically update the target network.

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- **Advantages of DQN**

- **Handles High-Dimensional State Spaces:**
  - ✓ Effective for complex environments, such as video games or robotics, where states are represented by high-dimensional data like images.



https://www.researchgate.net/figure/of-the-procedure-a-Construction-of-the-game-theoretical-model-for-human-motion-b_fig1_366427623

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- **Advantages of DQN**

- **Sample Efficiency:**
  - ✓ Uses experience replay to learn from past experiences multiple times, improving learning efficiency.

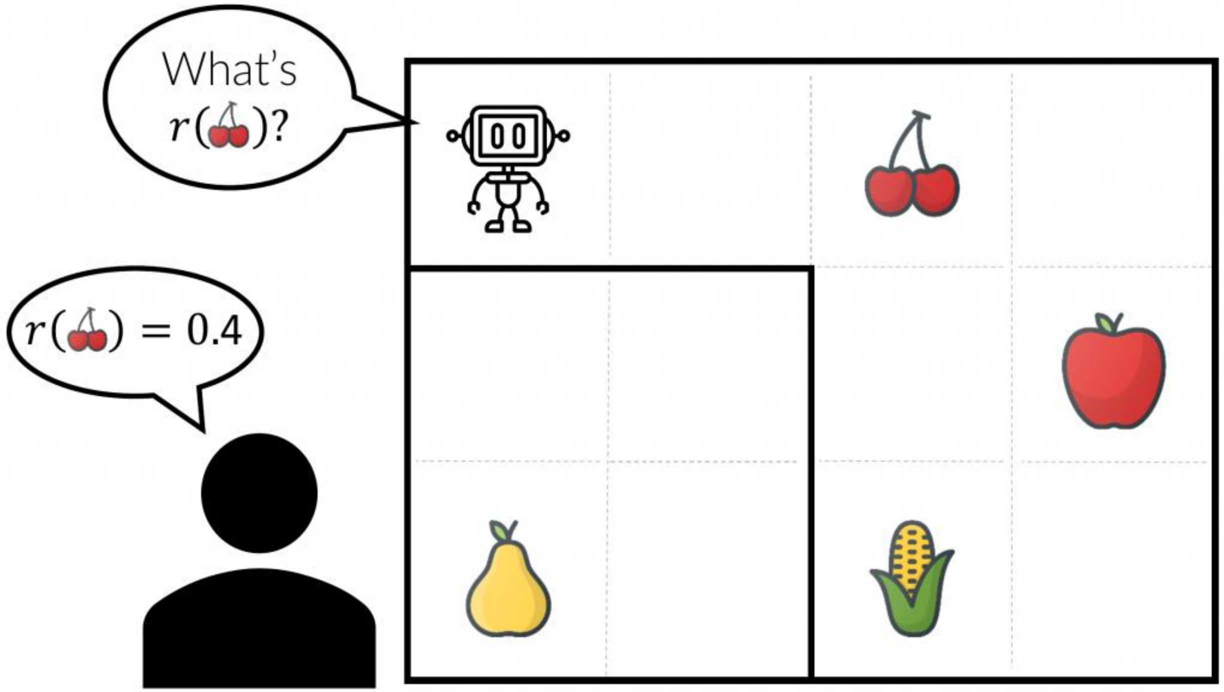


Image Source: https://www.researchgate.net/figure/of-the-procedure-a-Construction-of-the-game-theoretical-model-for-human-motion-b_fig1_366427623

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- **Advantages of DQN**

- **Improved Stability:**
  - ✓ The use of a target network helps stabilize training by reducing correlations between the Q-values being updated and the targets.

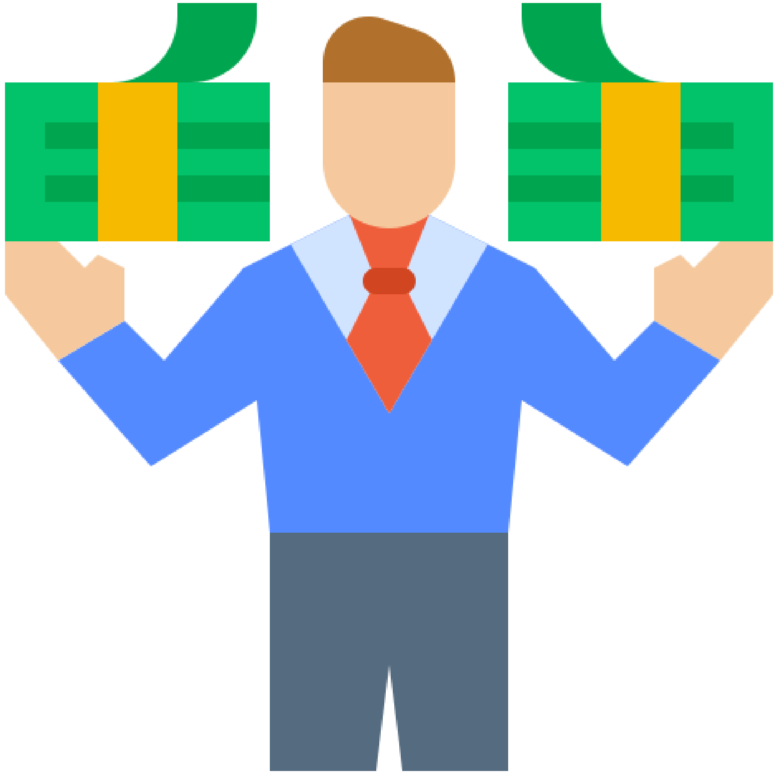


Image Source : https://www.flaticon.com/free-icon/stability_2362133

Unsupervised Learning and Recommender Systems

Deep Reinforcement Learning Model (Deep Q Network)

- **Applications of DQN**
  - ✓ **Gaming:**
    - Achieved human-level performance in Atari games by learning from raw pixel inputs.
  - ✓ **Robotics:**
    - Used for training robotic agents to perform complex tasks by interacting with a simulated or real environment.
  - ✓ **Finance:**
    - Applied in algorithmic trading to learn optimal trading strategies based on market data.

Image Source : https://www.youtube.com/watch?app=desktop&v=b9dWnCC-FvQ
Image Source : https://www.shutterstock.com/image-illustration/3d-illustration-ai-robot-serving-human-2304539565

Summary

Summary

K-Means Clustering: Anomaly Detection:	• Groups similar data points to discover underlying patterns without labels. • Identifies rare or unusual data points that deviate significantly from the norm.
Collaborative Filtering: Content-Based Deep Learning:	• Recommends items based on user behavior and preferences (e.g., Netflix, Amazon). • Suggests items by analyzing item features (e.g., text, audio) using deep learning (e.g., Spotify, YouTube).
Deep Q Network (DQN):	• Uses neural networks to approximate value functions, enabling agents to learn optimal actions in complex environments (e.g., gaming, robotics).

Knowledge Check

Unsupervised Learning and Recommender Systems

- **What is a key difference between collaborative filtering and content-based filtering in recommender systems?**

  A. Collaborative filtering uses item features, while content-based filtering uses user interactions.
  B. Content-based filtering uses item features, while collaborative filtering uses user interactions and similarities.
  C. Both techniques are identical in approach.
  D. Collaborative filtering is unsupervised, while content-based filtering is supervised.

Unsupervised Learning and Recommender Systems

- **What is a key difference between collaborative filtering and content-based filtering in recommender systems?**

  A. Collaborative filtering uses item features, while content-based filtering uses user interactions.
  B. Content-based filtering uses item features, while collaborative filtering uses user interactions and similarities.
  C. Both techniques are identical in approach.
  D. Collaborative filtering is unsupervised, while content-based filtering is supervised.

Answer: B. Content-based filtering uses item features, while collaborative filtering uses user interactions and similarities.

Unsupervised Learning and Recommender Systems

- **Deep Reinforcement Learning models, like Deep Q Networks (DQN), are particularly effective in:**

A. Image classification tasks

B. Situations requiring sequential decision-making under uncertainty

C. Solving linear regression problems

D. Detecting anomalies in static datasets

Unsupervised Learning and Recommender Systems

- **Deep Reinforcement Learning models, like Deep Q Networks (DQN), are particularly effective in:**

  A. Image classification tasks
  B. Situations requiring sequential decision-making under uncertainty
  C. Solving linear regression problems
  D. Detecting anomalies in static datasets

Answer: B. Situations requiring sequential decision-making under uncertainty

Thank You !!!