**Assignment 3**

**Name:** Sahil Kaundal **UID:** 21BCS8197

**Semester:** 4th  **Section/Group:** 807/B

**Subject:** Principles of AI **Subject Code:** 20CST-258

**Q1.**

**What are Recommender Systems? Explain with example.**

**Answer.**

**A recommendation engine** is a system that suggests products, services, information to users based on analysis of data. Notwithstanding, the recommendation can derive from a variety of factors such as the history of the user and the behaviour of similar users.

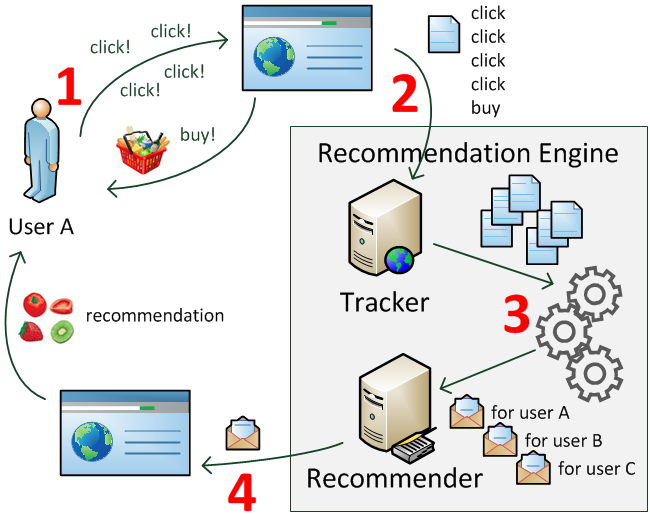
Recommendation systems are quickly becoming the primary way for users to expose to the whole digital world through the lens of their experiences, behaviours, preferences and interests. And in a world of information density and product overload, a recommendation engine provides an efficient way for companies to provide consumers with personalised information and solutions

A recommendation engine can significantly boost revenues, Click-Through Rates (CTRs), conversions, and other essential metrics. It can have positive effects on the user experience, thus translating to higher customer satisfaction and retention.

**Let’s take Netflix as an example.** Instead of having to browse through thousands of boxes sets and movie titles, Netflix presents you with a much narrower selection of items that you are likely to enjoy. This capability saves you time and delivers a better user experience. With this function, Netflix achieved lower cancellation rates, saving the company around a billion dollars a year.

Although recommender systems have been used for almost 20 years by companies like Amazon, it has been proliferated to other industries such as finance and travel during the last few years.

Recommendation engines need to know you better to be effective with their suggestion. Therefore, the information they collect and integrate is a critical aspect of the process. This can be information relating to **explicit interactions**, for example, information about your past activity, your ratings, reviews and other information about your profile, such as gender, age, or investment objectives. These can combine with **implicit interactions** such as the device you use for access, clicks on a link, location, and dates.



There are three main types of techniques for Recommendation systems; **content-based filtering**, **collaborative filtering**, and **knowledge-based system**.

**1. Content-based filtering**

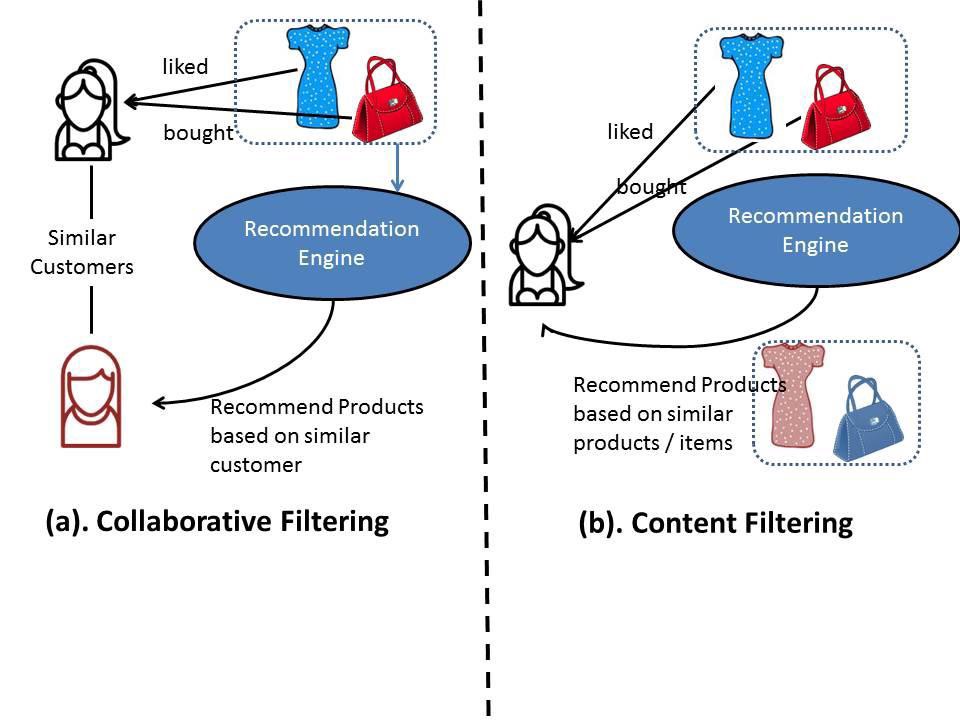
Content-based filtering is based on a single user’s interactions and preference. Recommendations are based on the metadata collected from a user’s history and interactions. For example, recommendations will be based on looking at established patterns in a user’s choice or behaviours. Returning information such as products or services will relate to your likes or views. With an approach like this, the more information that the user provides, the higher the accuracy.

Given the privacy and regulatory issues are important in some industries’ services, personal metadata and individual transactional data can be missing at the outset. These issues are commonly known as ‘cold start’ problems for recommender systems using this approach. Cold start occurs when a recommender system cannot draw inferences for a query due to lack of sufficient information. A particular form of the content-based recommendation system is a case-based recommender. These evaluate items’ similarities and have been extensively deployed in e-commerce.

A recommendation like ‘products similar to this’, is a typical instance of this type of approach. Overall, these are limited though to the specific domain and the level of categorisation available.

**2. Collaborative filtering**

Collaborative filtering is another commonly used technique. Collaborative filtering casts a much wider net, collecting information from the interactions from many other users to derive suggestions for you. This approach makes recommendations based on other users with similar tastes or situations. For example, by using their opinion and actions to recommend items to you or to identify how one product may go well with another. ‘Next buy’ recommendations are a typical usage. Collaborative filtering method usually has higher accuracy than content-based filtering; however, they can also introduce some increased variability and sometimes less interpretable results. They are especially weak in the absence of previously collected data. Without meaningful information on others, it becomes harder, naturally, to participate in any single person actions.



## 3. Knowledge-based system

Knowledge-based systems are systems where suggestions are based on an influence about a user’s needs and based on a degree of domain expertise and knowledge. Rules are defined that set context for each recommendation. This, for example, could be criteria that define when a specific financial product, like a trust, would be beneficial to the user. These do not, by default, have to use interaction history of a user in the same way as the content-based approach is, but can include these as well as customer products and service attributes, as well as other expert information. Given the way the system is built up, the recommendations can be easily explained. But building up this type of framework can be expensive. It tends to be better suited to complex domains where items are infrequently purchased or hence, data is lacking. Given this, it doesn’t suffer the same cold-start-up problems as others above.

**Q2.**

**How do you make sure which Machine Learning Algorithm to use?**

**Answer.**

For any given machine learning problem, numerous algorithms can be applied and multiple models can be generated. A spam detection classification problem, for example, can be resolved using a variety of models, including naive bayes, logistic regression and deep learning techniques like BiLSTMs.

Having a wealth of options is good, but deciding on which model to implement in production is crucial. Though we have a number of performance metrics to evaluate a model, it's not wise to implement every algorithm for every problem. This requires a lot of time and a lot of work.

**1-Categorize the problem**The next step is to categorize the problem.  
**Categorize by the input:**If it is a labelled data, it’s a supervised learning problem. If it’s unlabelled data with the purpose of finding structure, it’s an unsupervised learning problem. If the solution implies to optimize an objective function by interacting with an environment, it’s a reinforcement learning problem.  
**Categorize by output:**If the output of the model is a number, it’s a regression problem. If the output of the model is a class, it’s a classification problem. If the output of the model is a set of input groups, it’s a clustering problem.

**2-Understand Your Data**Data itself is not the end game, but rather the raw material in the whole analysis process. Successful companies not only capture and have access to data, but they’re also able to derive insights that drive better decisions, which result in better customer service, competitive differentiation, and higher revenue growth. The process of understanding the data plays a key role in the process of choosing the right algorithm for the right problem. Some algorithms can work with smaller sample sets while others require tons and tons of samples. Certain algorithms work with categorical data while others like to work with numerical input.

**Analyze the Data**In this step, there are two important tasks which are understand data with descriptive statistics and understand data with visualization and plots.

**Process the data**The components of data processing include pre-processing, profiling, cleansing, it often also involves pulling together data from different internal systems and external sources.

**Transform the data**The traditional idea of transforming data from a raw state to a state suitable for modelling is where feature engineering fits in. Transform data and feature engineering may, in fact, be synonyms. And here is a definition of the latter concept. Feature engineering is the process of transforming raw data into features that better represent the underlying problem to the predictive models, resulting in improved model accuracy on unseen data.

**3-Find the available algorithms**After categorizing the problem and understand the data, the next milestone is identifying the algorithms that are applicable and practical to implement in a reasonable time. Some of the elements affecting the choice of a model are:

* The accuracy of the model.
* The interpretability of the model.
* The complexity of the model.
* The scalability of the model.
* How long does it take to build, train, and test the model?
* How long does it take to make predictions using the model?
* Does the model meet the business goal?

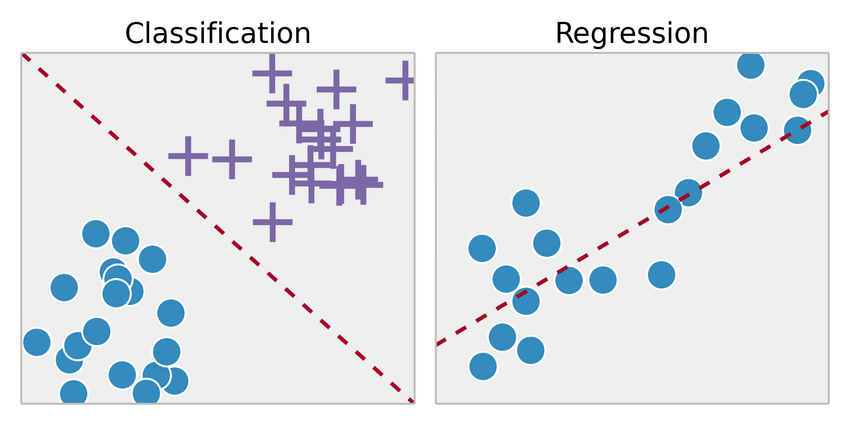
**4-Implement machine learning algorithms.**Set up a machine learning pipeline that compares the performance of each algorithm on the dataset using a set of carefully selected evaluation criteria. Another approach is to use the same algorithm on different subgroups of datasets. The best solution for this is to do it once or have a service running that does this in intervals when new data is added.

**5-Optimize hyperparameters.**There are three options for optimizing hyperparameters, grid search, random search, and Bayesian optimization.

**Types of machine learning tasks**

* **Supervised learning**
* **Unsupervised learning**
* **Reinforcement learning**

**Supervised learning**Supervised learning is so named because the human being acts as a guide to teach the algorithm what conclusions it should come up with. Supervised learning requires that the algorithm’s possible outputs are already known and that the data used to train the algorithm is already labelled with correct answers. If the output is a real number, we call the task regression. If the output is from the limited number of values, where these values are unordered, then it’s classification.



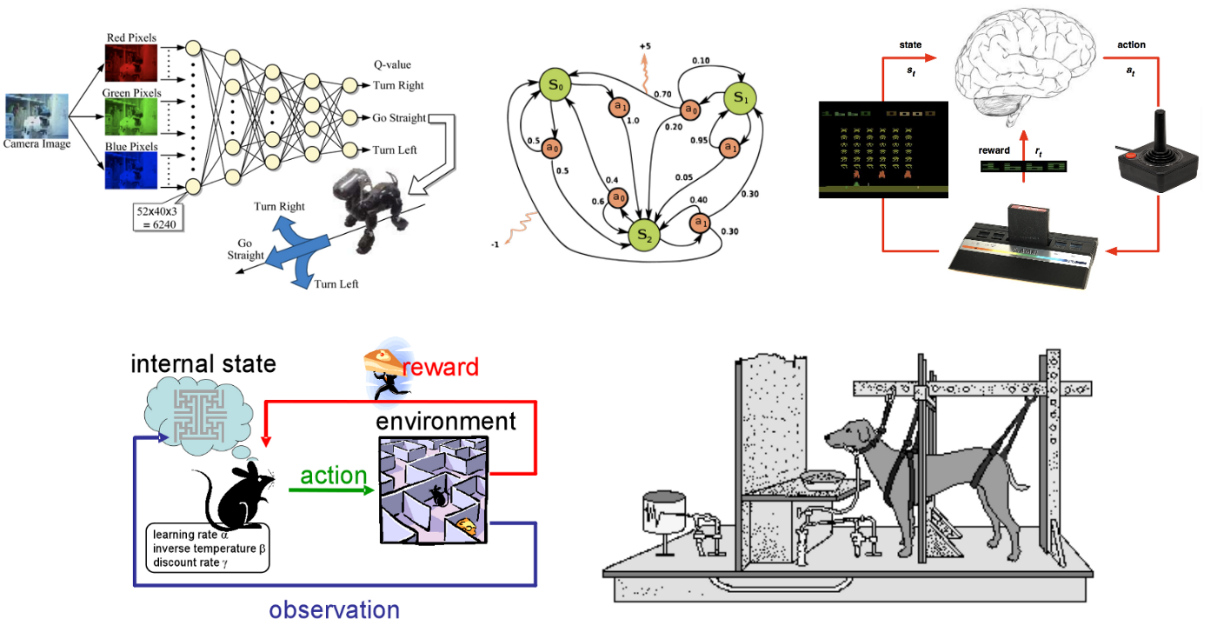
[**Supervised Learning**](https://www.researchgate.net/figure/Classification-vs-Regression_fig9_326175998)

**Unsupervised learning**Unsupervised machine learning is more closely aligned with what some call true artificial intelligence — the idea that a computer can learn to identify complex processes and patterns without a human to provide guidance along the way. There is less information about objects, in particular, the train set is unlabelled. It’s possible to observe some similarities between groups of objects and include them in appropriate clusters. Some objects can differ hugely from all clusters, in this way these objects to be anomalies.



[**Unsupervised Learning**](https://towardsdatascience.com/clustering-unsupervised-learning-788b215b074b)

**Reinforcement learning**Reinforcement learning refers to goal-oriented algorithms, which learn how to attain a complex objective or maximize along a particular dimension over many steps. For example, maximize the points won in a game over many moves. It differs from the supervised learning in a way that in supervised learning the training data has the role key with it so the model is trained with the correct answer itself whereas in reinforcement learning, there is no answer but the reinforcement agent decides what to do to perform the given task. In the absence of training dataset, it is bound to learn from its experience.



[**Reinforcement Learning**](https://www.eecs.tufts.edu/~jsinapov/teaching/comp150_RL/)