Recommendation Systems

1. Recommendation systems are used in the context of different use cases, each aiming to solve a different business problem. (movie, stocks, product recommendations etc..).
2. The common aim is to increase customer satisfaction that drives to increased commissions, greater sales, more revenue.
3. Coming to the technical part , we’ll focus on three things - **DATA, APPROACH, EVALUATION METRICS**.

DATA - here we will be requiring USER and PRODUCT((Movie/Stock/Product) data

USER DATA - features can be demographics of the customer (age, gender, location, other customer info)

PRODUCT DATA - if we consider the movies use case, features will be director, actors, movie length, genre, etc..

LINKING DATA - this table is very important for our analysis , this has data like - Customer/User ID, Product ID (Movie/Stock/Product), No: of Units/Rating, Transaction Date etc…

* APPROACH - I understood below 2 approaches :-

1. CONTENT BASED - we **match** customers to the products they have liked or bought.We’ll create a ‘**profile**’ for each user and each product individually and once we have these profiles :-

* Compute users-products similarity by creating user and product vectors and using **cosine similarity method**.
* Compute products-products similarity by using product vectors and **Jaccard similarity method**.
* **need to study more on the similarity methods and their implementation!!**

1. CLUSTERING - Clustering is typically used when the recommendation problem is going to be unsupervised(THERE’S NO STRUCTURE IN THE DATA).

* We can first find a structure in the data , like creating clusters that are characterised by a group of customers and their product interests.(**k-means ,kmedoids ,kmodes** are the methods I have experience using)
* We can then create labels and create a labelled dataset and make it a supervised learning problem.
* We can then predict what will be the products a particular customer would like.
* EVALUATION METRICS - **go-to metrics** used for recommendation systems precision and recall, in the context of recommendations systems :-
* Precision = (No.of relevant recommendations)/Total recommendations
* Recall = No.of relevant recommendations/

Total possible relevant products (in the data)

**THAT’S ALL I HAVE GOT ABOUT RECOMMENDATION SYSTEMS ! PROBABLY I SHOULD TRY AND BUILD ONE NOW** 🤔 **!**