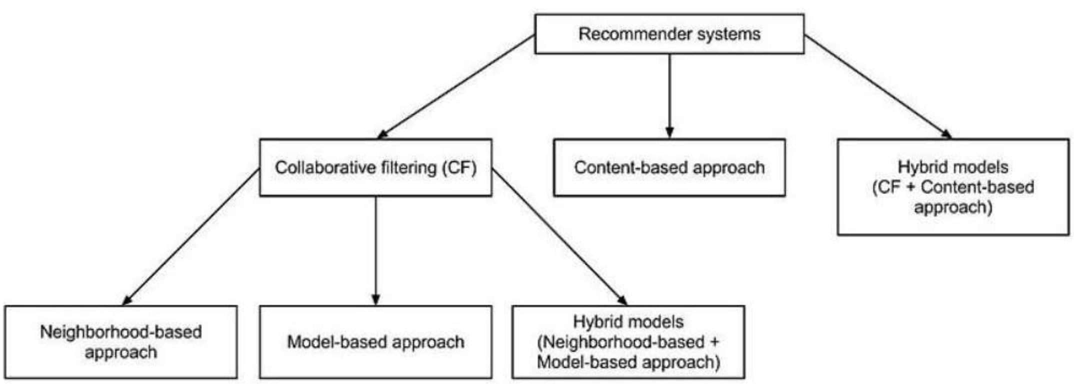
Recommender And Review System.

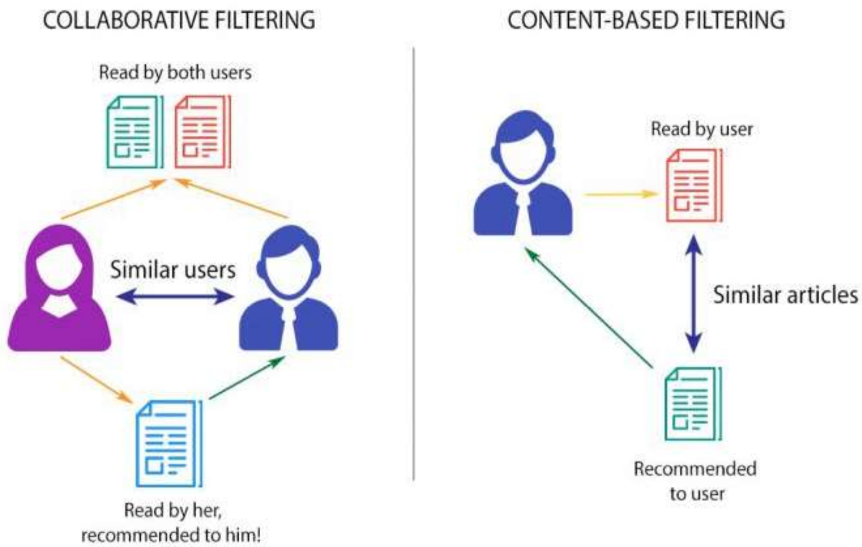
Team - DataMappers

Recommender System is a system that seeks to predict or filter preferences according to the user's choices. Recommender systems are utilized in a variety of areas including movies, music, news, books, research articles, search queries, social tags, and products, in general. A more formal definition of recommendation systems would be, *it is a subclass of information filtering system that seeks to predict the "rating" or "preference" a user would give to an item.*

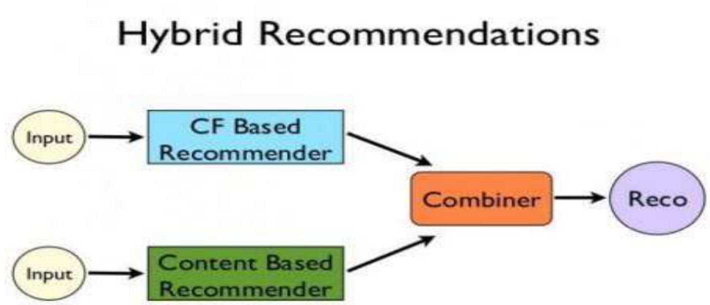
There are basically three important types of recommendation engines:



- **Collaborative filtering:** Collaborative Filtering algorithm considers "User Behaviour" for recommending items. They exploit the behaviour of other users and items in terms of transaction history, ratings, selection and purchase information. Other users behaviour and preferences over the items are used to recommend items to the new user.



❖ **Content-Based Filtering:** Content-based systems, recommends item based on a similarity comparison between the content of the items and a user's profile. The feature of items is mapped with the feature of users in order to obtain user – item similarity.



❖ **Hybrid Recommendation Systems:** This is the most sort after Recommender system that many companies look after, as it combines the strengths of more than two Recommender system and also eliminates any weakness which exists when only one recommender system is used.

So, then the question arises as to **How does a recommender system work?**

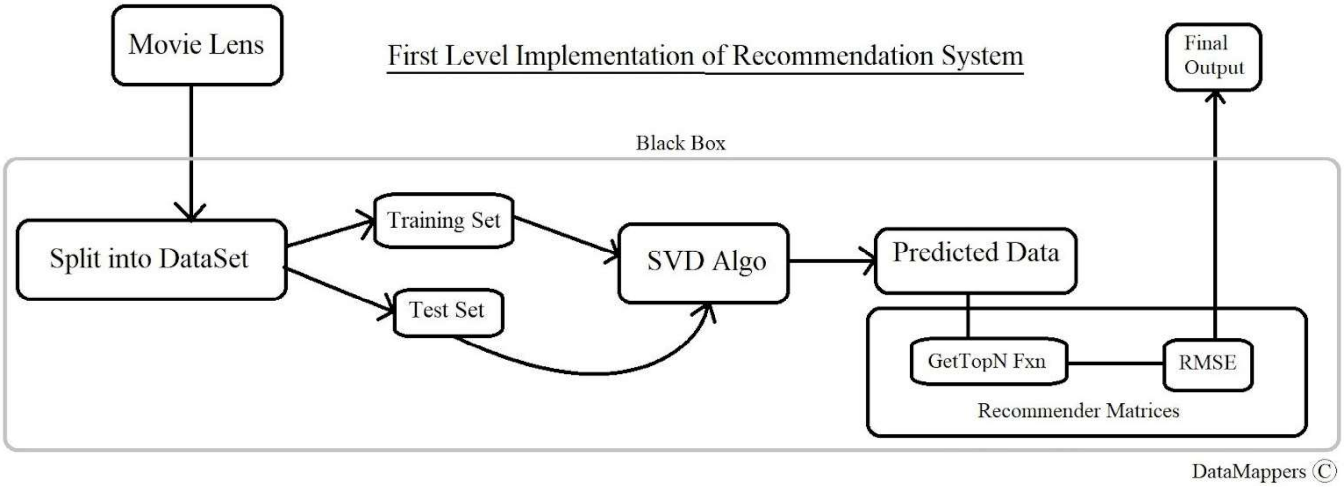
Well, a typical recommendation system processes data through a different kind of phases, namely **collection, storing, analyzing and filtering.**

- **Collection of data:** The first step in creating any recommender system is the gathering of data which can be either explicit (data inputted by users such as ratings and comments on products) or implicit (order history/return history, Cart events, Pageviews, search log etc.) which will be created for every user visiting the site.

- **Storing of data:** It is obvious that the more data which we provide to our algorithms, better the recommendations would be. The type of data that one uses to create recommendations can help one decide the type of storage one should use.

- **Analyzing the data:** We can provide immediate recommendations to the user as they are viewing the product by analyzing the data and by filtering data accordingly.

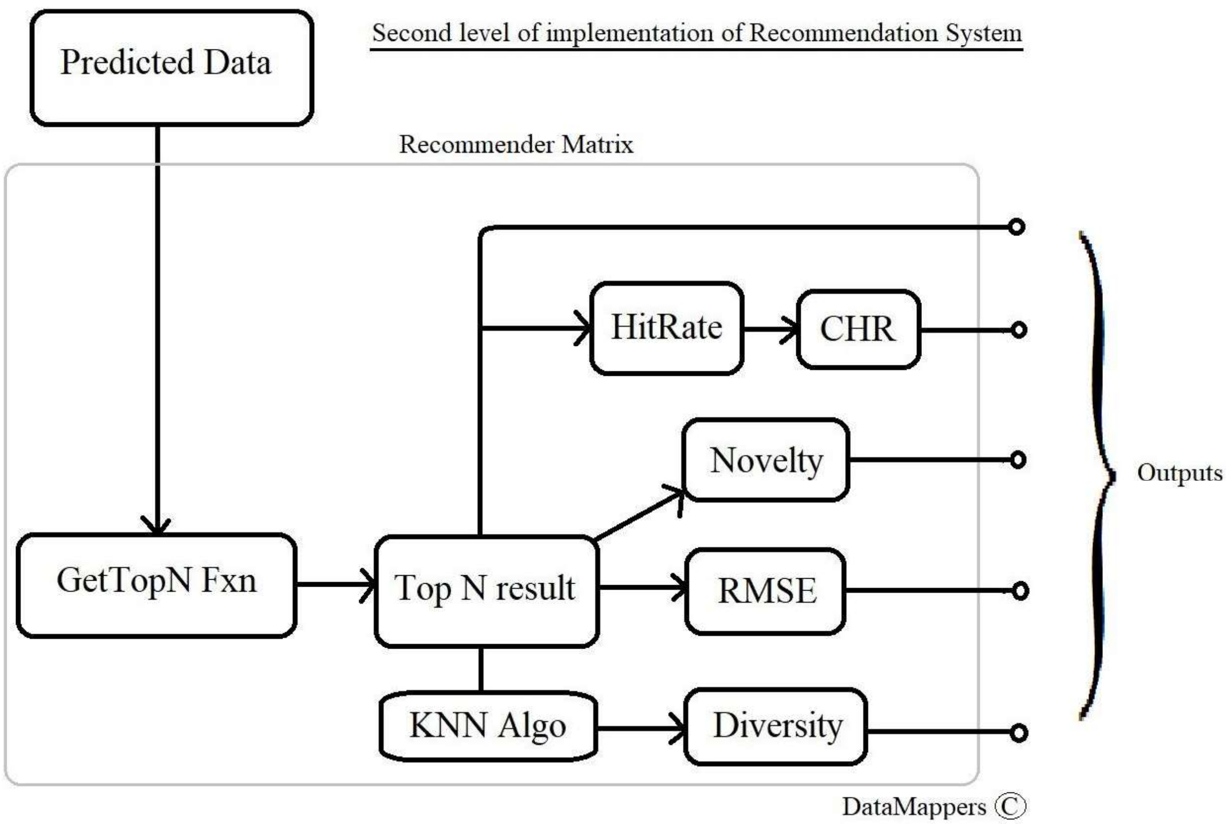
- **Filtering the data:** Finally, we need to filter the data to get the relevant data necessary to provide recommendations to the user.



- **Movie Lens:** It contains ratings and info about the movie and converting them into datasets that SurPRISE (Simple Python Recommendation System Engine) can use, also includes useful functions to quickly lookup movie titles & other utility functions.
    - Split into dataset
        - Training Set: It contains 75% of our split dataset.
        - Test Set: It contains 25% of our split dataset.

- **SVD (Singular Value Decomposition) Algorithm:** It is used across many matrices that we have designed and is used for decomposing a matrix. But it does not compute Items similarity in order to compute diversity. Thus, we need to use KNN Baseline.

- **Predicted Data:** We passed our training data to fit in the algorithm to train it. Later test dataset is used upon the algorithm to get respective prediction.



❖ **Functions applied on TopN results** :-

- **Diversity:** It is used to give diversity from our predicted data. We use KNN algorithm to compute it. We need a matrix of similarities scores between all pairs of items in data sets. It works on the concept of 1-S, where S is the average similarity score between every possible pair of recommendations for a given user. Higher means more diverse.

- **Novelty:** Average popularity rank of recommended items. Higher means more novel.We take a handy dictionary of popularity rankings of every item as a parameter and then go to user's Top N recommendation and compute avg of popularity ranking.

- **Hit Rate:** We pass a dictionary of top N movies for each user ID & the set of test movie that is left out of the training data set. It returns how often we are able to recommend a left-out rating. Higher value is better.

- **Cumulative Hit Rate (CHR):** Hit rate, confined to ratings above a certain threshold. It returns how often we recommend a movie the user already liked. Higher value is considered better.

- **RMSE:** Root Mean Squared Error. Lower values means better prediction accuracy.

Applications

The RRS (Review & Recommender System) will be used to recommend a specific object to a user based upon its earlier reviews collected over a period. This system can be applied over books in library of particular type, snacks at canteen etc. IMDB dataset will used as sample dataset to demonstrate movie recommendation as a prototype (downloaded from Kaggle)

MADE BY:-

ARPIT JAIN (17UEC028)
ADITYA RAJ (17UEC143)
AARYAMAN SHARMA (17UEC001)

Thank You.