**MOVIE RECOMMENDER SYSTEM**

**Aegis School of Data Science**

**Course : Python**

**Manavi Agrawal**

**Urvi Parekh**

**Abstract :**

Here we have tried to build simple movie recommender system using Python.The approach used is Popluar based approach and content based approach.The data is scraped from various imbd links, and stored in SQLite databse.Then future the project is deployed in Dash plotly,presently the deployment done is in static form.This project can be future developed.

**Keywords** :web-scraping , movie recommendation system , sqllite database , popular based approach,content based approach, dash , plotly

**Background** :

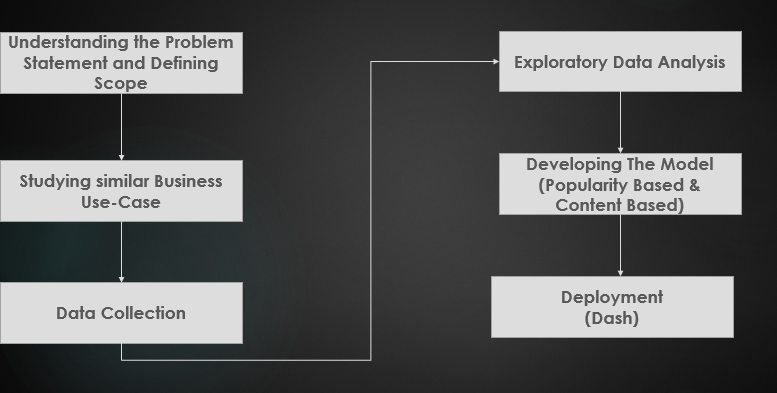
* In Today’s world Machine Learning and Artificial Intelligence plays a great role in our day-to-day lives.
* In n number of examples of Machine Learning and Artificial Intelligence examples Recommender System plays an important role now-a-days.
* Recommender System are divided into two major parts :
  + Content Based Approach
    - A content based recommender works with data that the user provides, either explicitly (rating) or implicitly (clicking on a link). Based on that data, a user profile is generated, which is then used to make suggestions to the user. As the user provides more inputs or takes actions on the recommendations, the engine becomes more and more accurate.
  + Collaborative Based Approach
    - * Collaborative filtering is the process of filtering for information or patterns using techniques involving collaboration among multiple agents, viewpoints, data sources, etc Collaborative filtering methods have been applied to many different kinds of data including: sensing and monitoring data, such as in mineral exploration, environmental sensing over large areas or multiple sensors; financial data, such as financial service institutions that integrate many financial sources; or in electronic commerce and web applications where the focus is on user data, etc.
* Many more Approach are there like Popular Based Approach , Knowledge Based Approach etc.
* Examples : Movie Recommendation System , Job Recommendation System , Product-Based Recommendation System and many more.

**Project Scope**

**Simple Movie Recommender System**

* Project Definition : We have tried to build the recommender System by using popularity based approach and content based approach.
* Data Collected : Scraped from IMDB website.
* Front-end : Dash
* Back-end : SQLITE Database
* Developed : Anaconda : Spyder

**Project Pipeline**



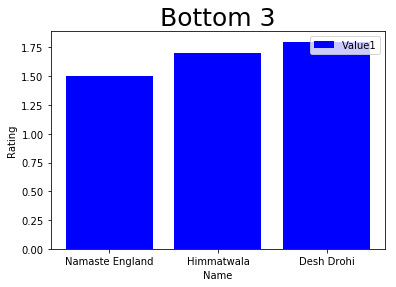
**Business Case Study**

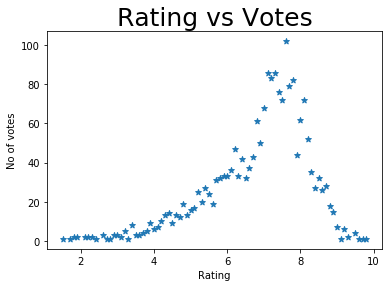
**NetFlix Use-Case**

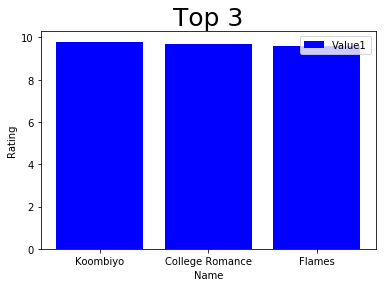
* They recommend on the basis of :
  + The interactions with the service (such as viewing history and how we rated other titles),
  + other members with similar tastes and preferences on service,
  + information about the titles, such as their genre, categories, actors, release year, etc.
* Also They catch the Information like :
  + the time of day we watch,
  + the devices we are watching Netflix on, and
  + how long we watch.
* When we create our account into Netflix , they ask us to select few titles and know our interested movie.
* If we don't select titles then they randomly just show us titles on the page.
* And once we start watching movies and rating them they get the data and is inputted into their algorithms.
* In addition to this they rank their title rows on the basis of 3 level personalization :
* the choice of row (e.g. Continue Watching, Trending Now, Award-Winning Comedies, etc.)
* which titles appear in the row, and
* The ranking of thoese rows
* The most strongly recommended rows go to the top. The most strongly recommended titles start on the left of each row and go right -- unless you have selected Arabic or Hebrew as your language in our systems, in which case these will go right to left.
* They don’t consider demographic information like age and gender.
* They take feedback from the user from time to time and on the basis of that they try to improve their system.

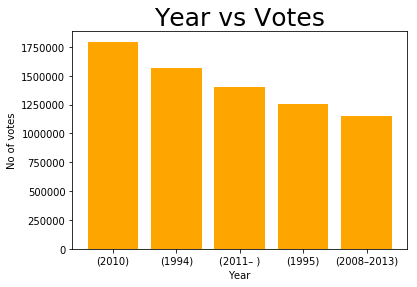
**Data Information**

The Data is collected from different imdb website. It consists of variety of movies and tv shows in different languages like Hindi , English , Chinese , Japanese , French , Telgu etc. There are about 2000 rows having columns Title , Year , Genre , Director , Actor , Rating and votes.There are no missing values or any outliers.It is stored in SQLite database.

**Exploratory Data Analysis**

****

****

****

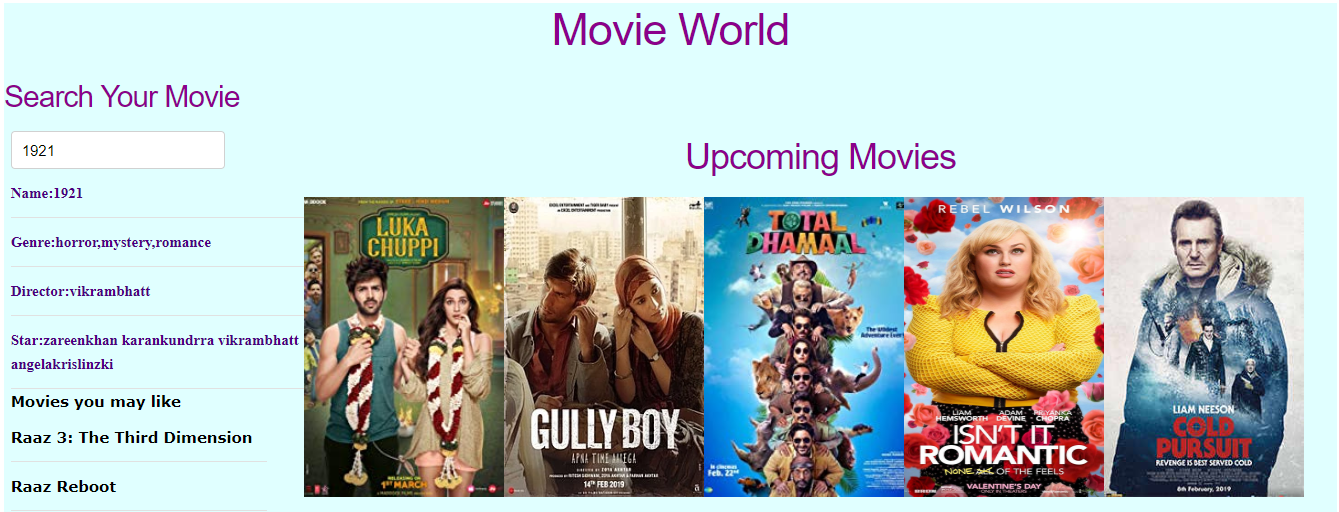
**Model Building**

1. First Approach :
   * Popularity Based Approach : In this approach on the basis of rating and votes the weighted average score is calculated and on that basis we get our most popular movies or we can get worst movies.
2. Second Approach :

* Content Based Approach : In this approach on the basis of genre , director and actor the similar kind of movies are suggested to users.

**Deployment**

* 1. We have deployed the recommender system in Dash Ploty.
  2. It is divided into 3 Parts **:**
     + Section 1 : **Search Movies** , Enter The Movie Name and gets its details and also view the similar kind of movies related to that movie on the basis of genre , director , actor.
     + Section 2 : **Upcoming Movies** , The upcoming movies are displayed and you can view trailers of it.
     + Section 3 : **Popular Movies** , The 10 most popular movies with their score is present in bar graph form.
* A sample Screen Shot of Deployment



**Future Works**

1. This can be made more user specific by creating the user profile.
2. There can be collaboration of content based and collaborative based approach and be more efficient.
3. The user log can be stored and useful insights can be derived from it.
4. More interactive User Interface can be built.
5. We can consider more features like plot description into account while building the recommended systems

**Conclusion**

This was basic recommended system as our python project which can be furthur improvised in future.