**MINI PROJECT**

(2020-21)

**Movie Recommendations System**

**MID-TERM REPORT**

****

**Institute of Engineering & Technology**

**Submitted by-**

**Diya Gupta**

(181500228 )

**Khushi Verma**

(181500324)

**Shivam Tripathi**

(181500675)

**Vaibhav Kumar**

(181500774)

**Shivam Rajpoot**

(181500669)

**Supervised By: - Mr. Piyush Vashistha** (Asst. Professor)

Department of Computer Engineering & Applications

**Content**

**.**

**Abstract**

**1. Introduction**   
 1.1 General Introduction to the topic

1.2 Area of Computer Science

1.3 Hardware and Software Requirements

**2. Problem definition**

**3. Objectives**

**4. Implementation Details**

**5. Progress till Date & The Remaining work**

**6. Some Screenshots**

**7. References**

**ACKNOWLEDGEMENT**

I would like to thank Dr. Piyush Vashistha for helping, and guiding us and helping us in mini project. I would also like to thank my team member who played their role honestly and make this project a successful project .

**Abstract**

In today's digital world **recommender** systems have become ubiquitous in our lives. Yet, currently, they are far from optimal. In this project, we attempt to understand the **movie recommendation systems** .

**Recommender systems** are the **systems** that are designed to recommend things to the user based on many different factors. These systems predict the most likely movies that the users are most likely to watch and are of interest to.

This**system** attempts to solve the problem unique**recommendations** which results from ignoring the data specific to the user. The psychological profile of the user, their watching history and the data involving**movie** Scores from  other websites is collected. They are based on aggregate similarity calculation.

**Introduction**

* 1. **General Introduction to the topic**

The **algorithm** segregates the list of **movies** from the dataset according to the inputs provided by user and finally displays the list of **movies**. ... This is achieved by using **collaborative filtering** approach, wherein the system will provide **recommendations** to other like- minded users which have the same taste.

A **movie recommendation** is important in our social life due to its strength in providing enhanced entertainment. Such a system can suggest a set of movies to users based on their interest, or the popularities of the movies.

Although, a set of movie recommendation systems have been proposed, most of these either cannot recommend a movie to the existing users efficiently or to a new user by any means. In this paper we propose a movie recommendation system that has the ability to recommend movies to a new user as well as the others.

It mines movie databases to collect all the important information, such as, popularity and attractiveness, required for recommendation. It generates movie swarms not only convenient for movie producer to plan a new movie but also useful for movie recommendation. Experimental studies on the real data reveal the efficiency and effectiveness of the proposed system.

**Here's a high-level basic overview of the steps required to implement a user-based collaborative recommender system.**

1. Collect and organize information on users and products.
2. Compare User A to all other users.
3. Create a function that finds products that User A has not used, but which similar users have.

**Implementation Details**

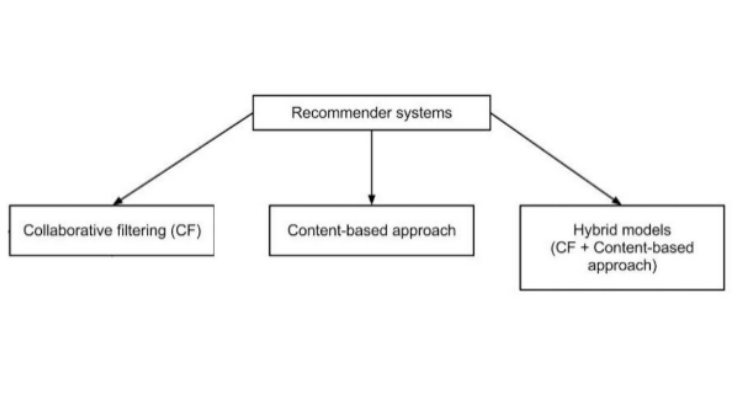
* Recommendation system produces a rank list of items on a which a user might

be interested, in the context of his current choice of an item.

* Subclass of information filtering system that seek to predict the ‘rating’

or ‘preferences’ that a user would give to them.

* Helps deciding in what to wear, what to buy, what stocks to purchase etc.
* Recommendation systems has mainly two elements Item and User.



**Movie Recommendation System**

1. Content Based: The recommendation system recommends other movies which are similar to selected movie.

f(movie) 🡪 {movies}

1. Collaborative: The recommendation system recommends movies which are rated highly by the similar users.

f (movies, user)🡪{movies}

**System Description**

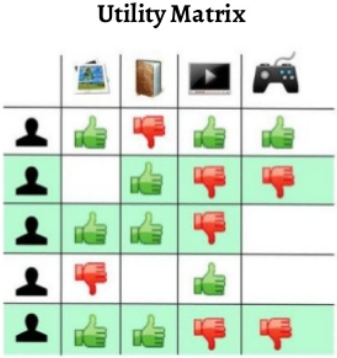
Owing to the various demerits of pure content-based and pure CF based systems, we have proposed a hybrid recommender system which is known as content-boosted collaborative filtering system. This hybrid system takes advantage from both the representation of the content as well as the similarities among users. The intuition behind this technique is to use a content-based predictor to fill the user-rating matrix that is sparsely distributed. After the pre-processing the movie content database is stored. The dataset consists of a user-rating matrix. Content-based predictions are used to train each user-rating vector in the user-rating matrix and convert it into a pseudo rating matrix which combines actual rating with the predicted ratings. Collaborative filtering is then applied to this full pseudo user-rating matrix to make recommendation for an active user.



**Collaborative Filtering**

**Utility Matrix:** Users have preferences for certain items and these preferences must be discovered from the data. The data is represented as a utility matrix, a value that represents the rating given by that user for that item and is given for each user-item pair.

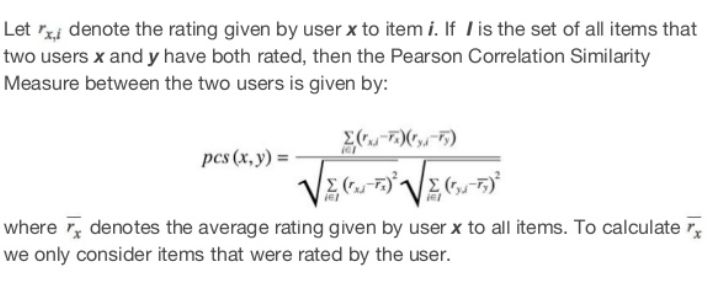
The goal of the recommendation engine is to predict the blanks in a utility matrix.



**Similarity Measures**

**Pearson Correlation Similarity Measure**: Measure of similarity of users or items from the rows and columns of the Utility Matrix. It is easy to interpret and tends to give better results than other similarity measures. Normalize the ratings.

**PCS Measure**



**Objectives**

**\*** The aim of recommendation systems is just the same---

**1.** Recommendation system is to achieve customer loyalty by providing relevant content.

**2**  Movie recommendation systems provide a mechanism to assist users in classifying users with similar interests..

**Progress Till date & Remaining Work**

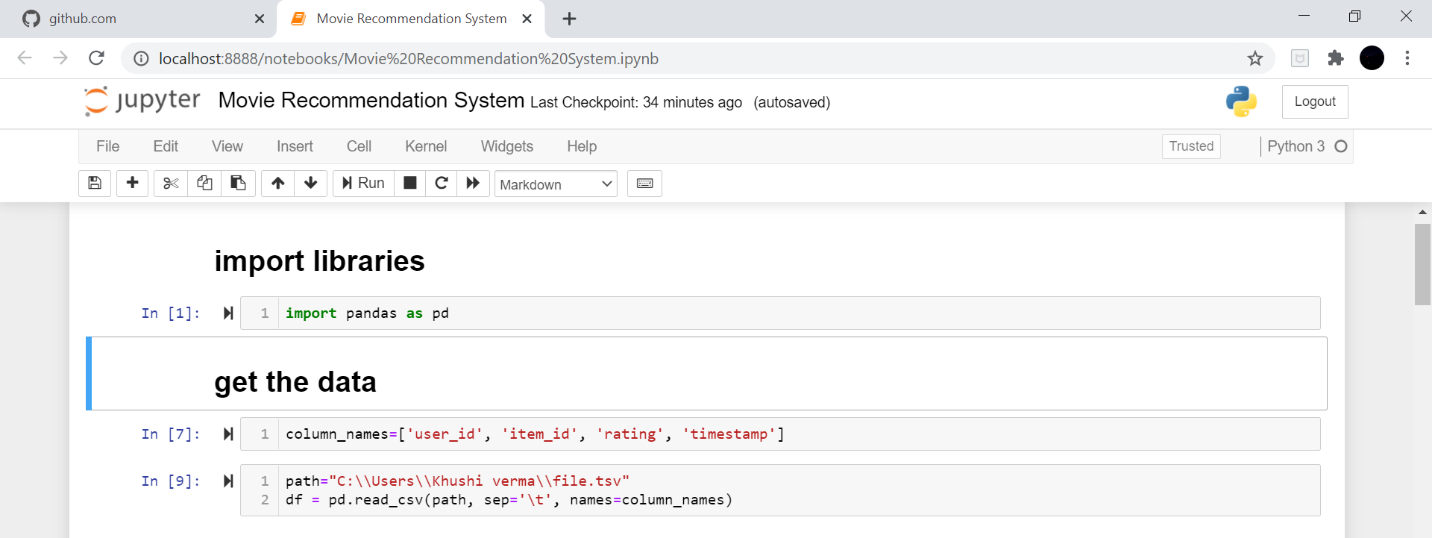
Progress till date-

\*Collection of data

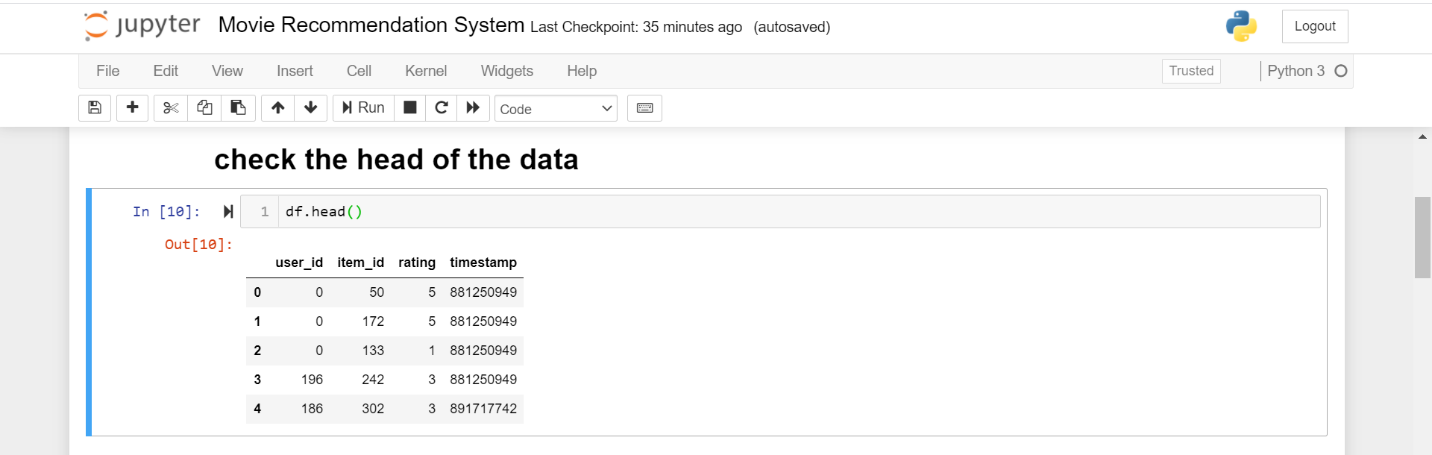
\*Visualization of data

**Some Screenshots**

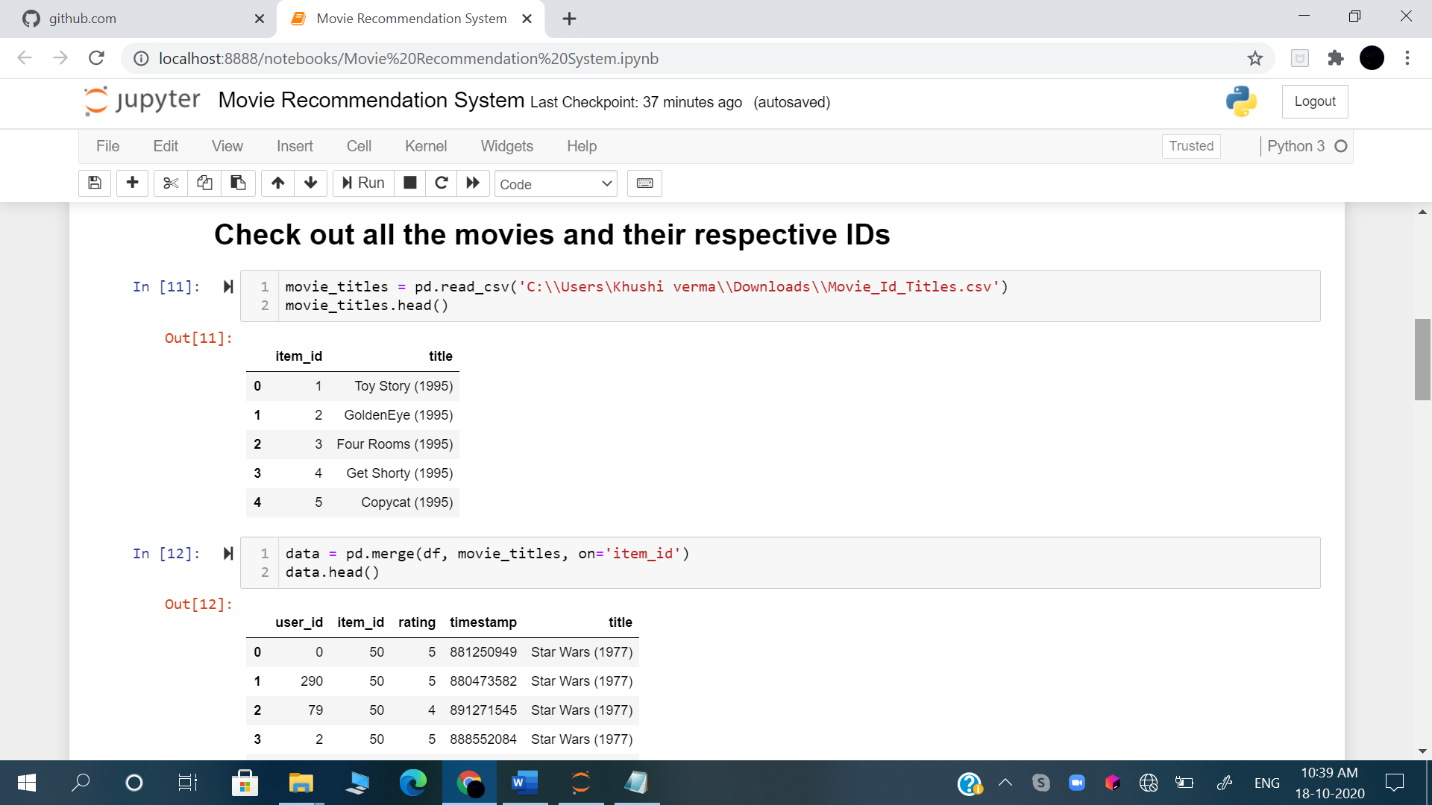
1.

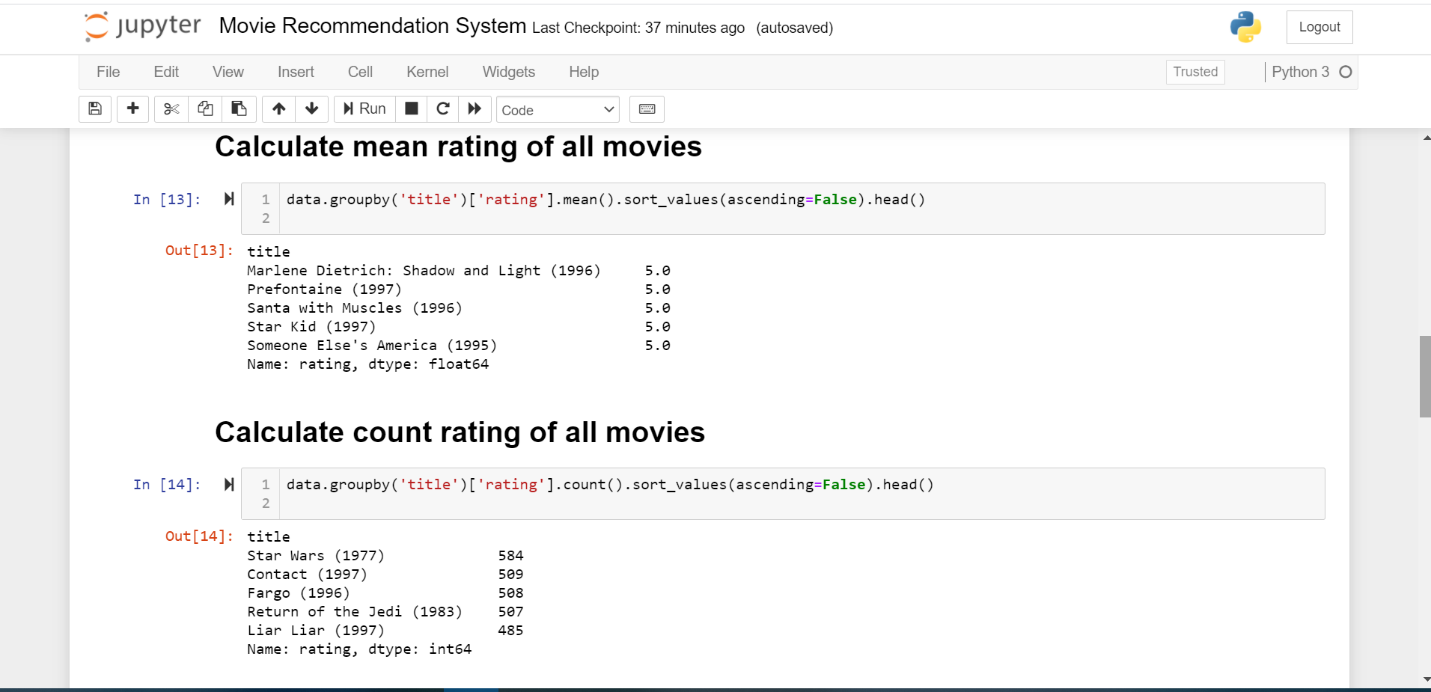


2.

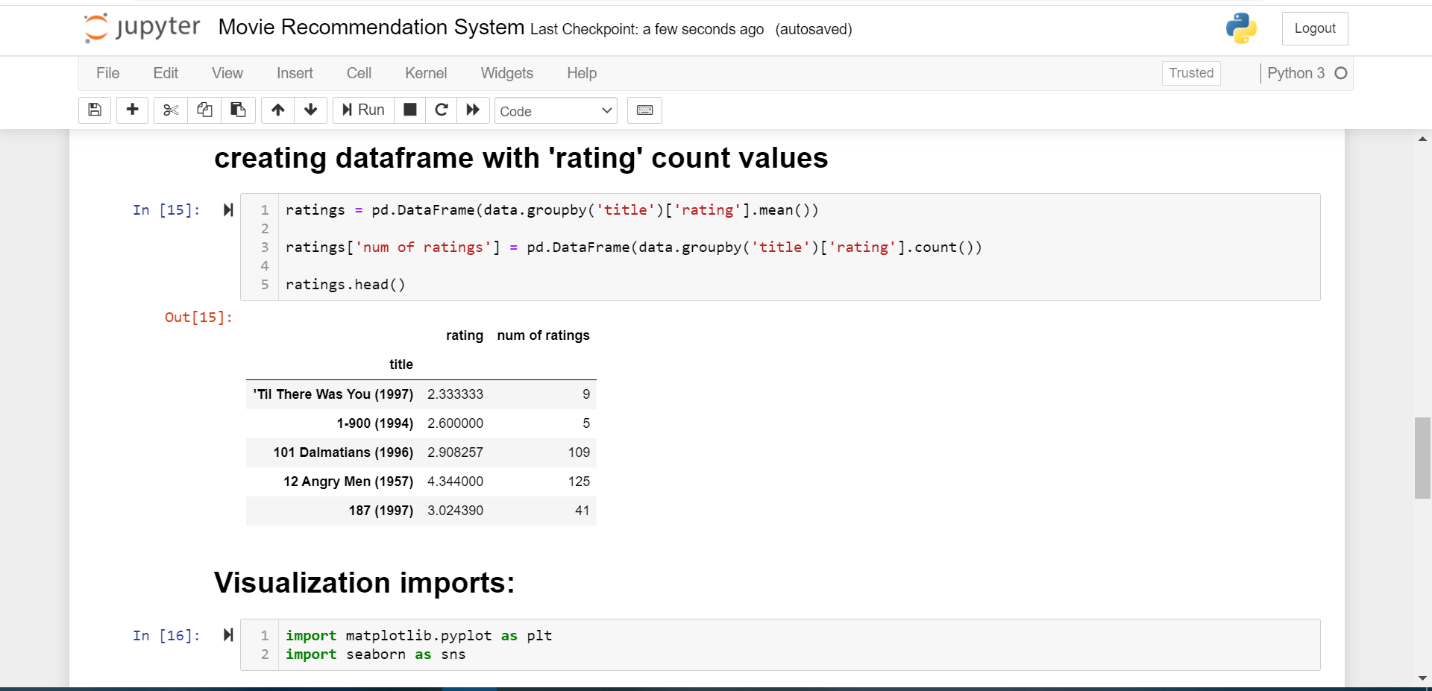


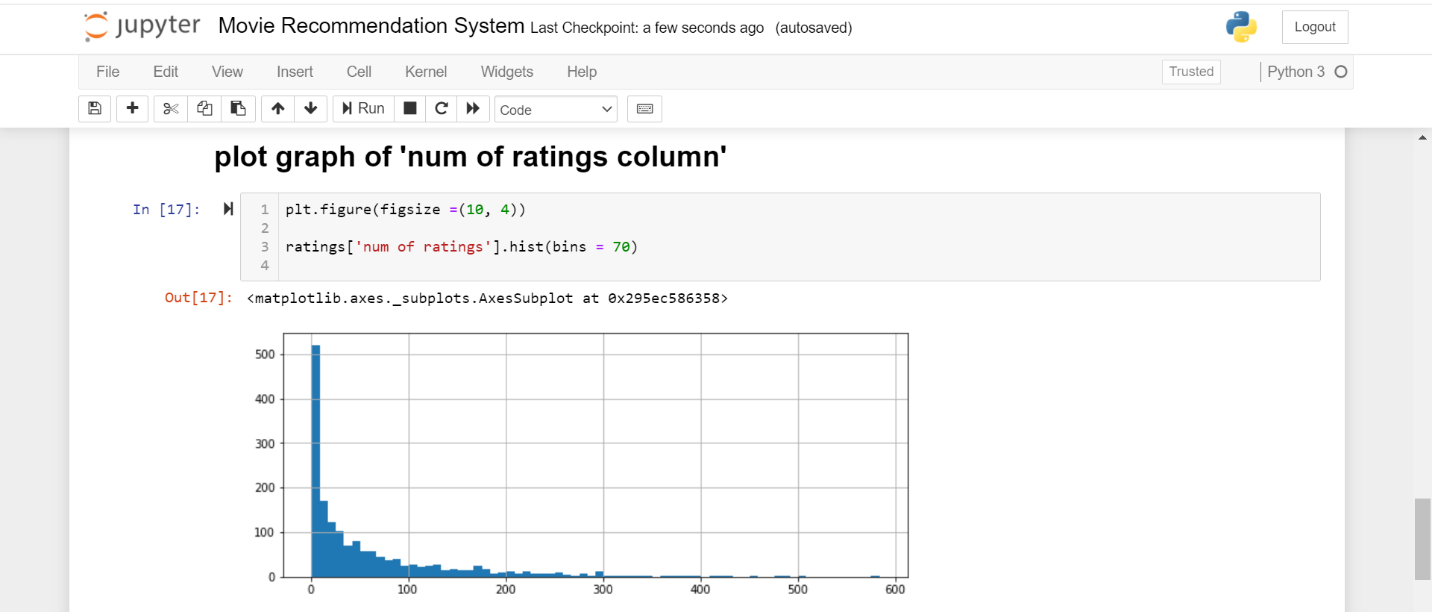
3.



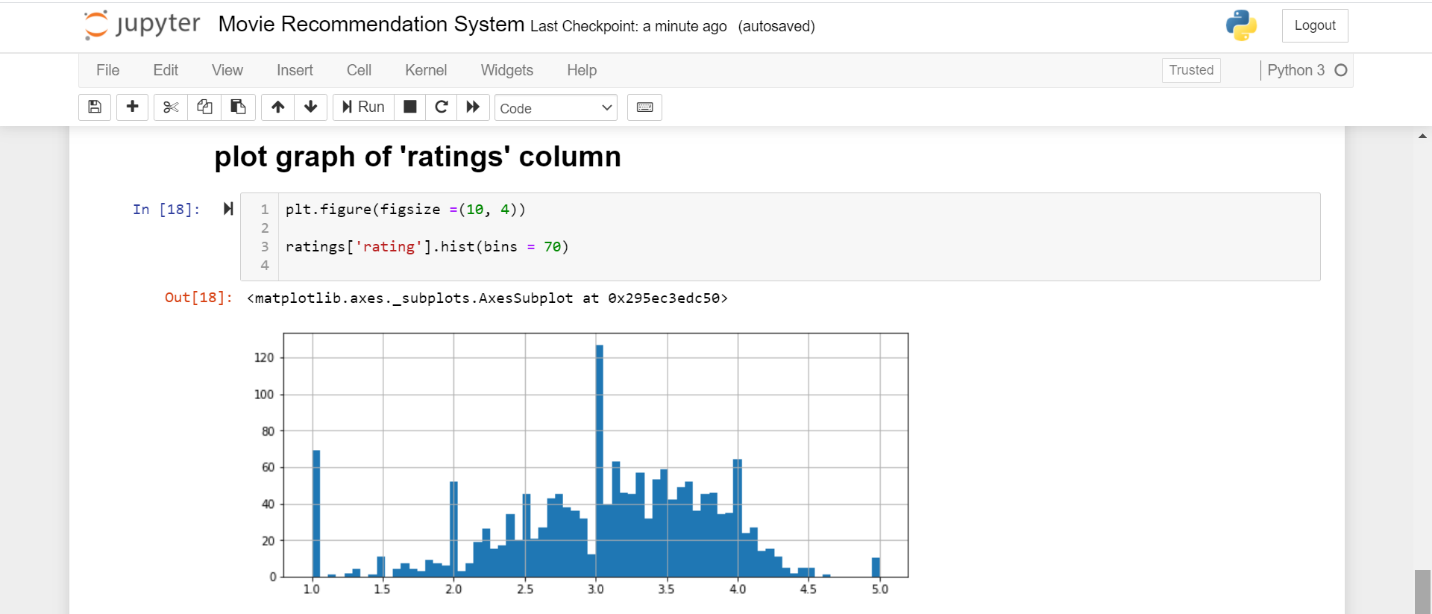
4.

5.



6.

7.



**Reference Websites we used-**

1. [www.kaggle.com](http://www.kaggle.com) (for data collection)
2. geeksforgeeks.org (for learning concepts)