ABSTRACT

In recent years, with the exponential increase in the growth of Internet and Web usage, Recommendation Systems are being utilized by various E-commerce sites to give significant recommendations to their users, such that both the users and providers will be benefited. The overwhelming amount of data necessitates mechanisms for efficient information filtering. Recommender systems have the effect of guiding users in a personalized way to interesting objects in a large space of possible options. In this project, we will look at three different recommender system approaches namely Collaborative filtering (CF), Content-based filtering, Hybrid recommender systems that can be used on different e-commerce websites.

1.INTRODUCTION

The Data is rapidly growing online and it is a major problem to analyse this huge amount of data in different fields. The abundance of information available on the Web and in Digital Libraries, in combination with their dynamic and heterogeneous nature, has determined a rapidly increasing difficulty in finding what we want when we need it and, in a manner, which best meets our requirements. Recommender system address the problem of filtering information that is likely of interest to individual users. Typically, user profiles are employed to predict ratings for items that have not been considered. Depending on the application domain, items can be web pages, movies or any other products found on a web store. In this project we will work on movielens dataset.

Although many different approaches to recommender systems have been developed within the past few years, the interest in this area still remains high. This is due to growing demand on practical applications, which are able to provide personalized recommendation and deal with information overload.

Recommender systems are usually classified into content-based and collaborative filtering-based recommender systems. Content-based recommendations are typically based on item similarity to objects the user preferred in the past. In contrast, collaborative recommendation systems depend on the ratings given by individuals with similar taste and preference. However, both methods have their advantages, they also have certain disadvantages, some of which can be solved by combining both techniques to improve the quality of the recommendation. The system is known as a hybrid recommender system.

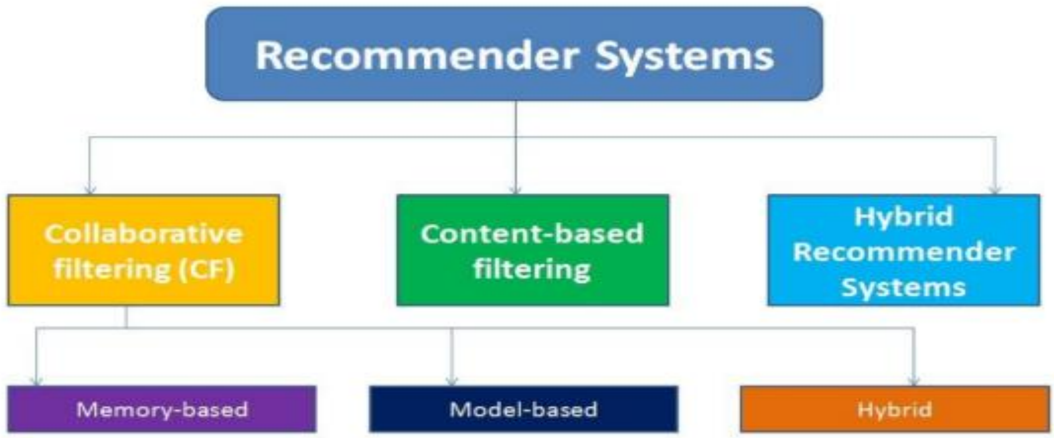

Fig. 1. Different types of Recommender systems

2.Recommender Systems

2.0. Technologies used:

- Python 3
- FLASK

2.1. Content-based Recommender Systems:

Systems implementing a content-based recommendation approach analyse a set of documents and/or descriptions of items previously rated by a user and build a model or profile of user interests based on the features of the objects rated by that user. The profile is a structured representation of user interests, adopted to recommend new interesting items. The recommendation process basically consists in matching up the attributes of the user profile against the attributes of a content object.

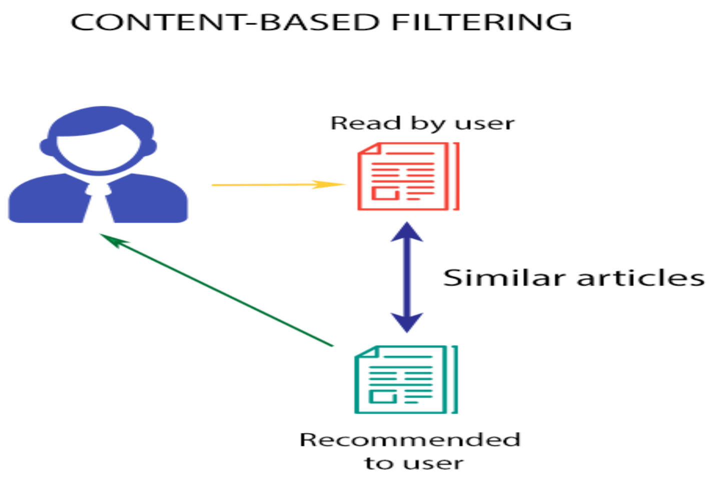

Fig 2.1: content-based filtering

Content-based filtering methods are based on a description of the item and a profile of the user's preference. These algorithms try to recommend items that are similar to those that a user liked in the past or is examining in the present. In particular, various candidate items are compared with items previously rated by the user and the best-matching items are recommended. This approach has its roots in information retrieval and information filtering research. Basically, these methods use an item profile characterizing the item within the system. The system creates a content-based profile of users based on a weighted vector of item features. The weights denote the importance of each feature to the user and can be computed from individually rated content vectors using a variety of techniques. Some of approaches use the average values of the rated item vector.

2.2.1. Collaborative Filtering:

Collaborative filtering approach assumes that if a person X has the same opinion as a person Y on an issue, X is more likely to have Y's opinion on a different issue 'a' than to have the opinion on 'a' of a person chosen randomly. For example, a collaborative filtering recommendation system for laptop tastes could make predictions about which laptop brand a user should like given a partial list of that user's tastes (likes or dislikes).

Typical workflow of a collaborative filtering system is as under:

- A user expresses his or her preferences by rating items (e.g. articles, videos or books) of the system. These ratings can be viewed as an approximate representation of the user's interest in the corresponding domain.
- The system matches these users' ratings against other users and finds the people with most similar tastes.
- With similar users, the system recommends items that the similar users have rated highly but not yet being rated by this user.

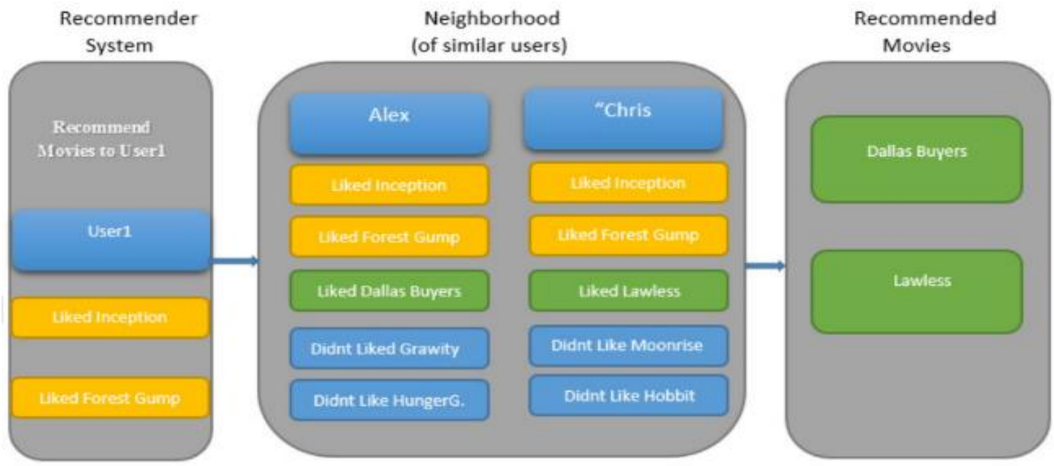

Fig 2.2.1: Collaborative filtering Systems applied in User-based Recommender System.

2.2.2Memory-based collaborative filtering: This mechanism uses user rating data to compute similarity between users or items. This is used for making recommendations. This was the earlier mechanism and is used in many commercial systems. It is easy to implement and is effective. Typical examples of this mechanism are item-based/user-based top-N recommendations. The Representative techniques are Item-based/user-based top-N recommendations.

Advantages:

- easy implementation
- new data can be added easily and incrementally
- need not consider the content of the items being recommended
- scale well with co-rated items

2.2.3: Implementation

Normalising the level optimism of various user by finding the adjusted mean. The algorithm calculates the similarity between two users or items and produces a prediction for the user by taking the weighted average of all the ratings. Similarity computation between items or users is an important part of this approach.

Then the rating for an item is predicted by the following formula:

$$p_{u,i} = \mu_u + \frac{\sum_{v \in N(u;i)} cos(u,v)(r_{v,i} - \mu_v)}{\sum_{v \in N(u;i)} |cos(u,v)|}$$

The Pearson correlation similarity of two users x, y is defined as

$$\mathrm{simil}(x,y) = \frac{\sum\limits_{i \in I_{xy}} (r_{x,i} - \bar{r_x})(r_{y,i} - \bar{r_y})}{\sqrt{\sum\limits_{i \in I_{xy}} (r_{x,i} - \bar{r_x})^2} \sqrt{\sum\limits_{i \in I_{xy}} (r_{y,i} - \bar{r_y})^2}}$$

where I_{xy} is the set of items rated by both user x and user y.

The user based top-N recommendation algorithm uses a similarity-based vector model to identify the k most similar users to an active user. After the k most similar users are found, their corresponding user-item matrices are aggregated to identify the set of items to be recommended.

2.3. HYBRID RECOMMENDER SYSTEMS

Hybrid recommender systems are based on the combination of Collaborative filtering and Content-based filtering. It improves the prediction performance. Importantly, it overcomes the CF problems such as sparsity and loss of information. Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model. These methods can also be used to overcome some of the common problems in recommender systems such as cold start and the sparsity problem.

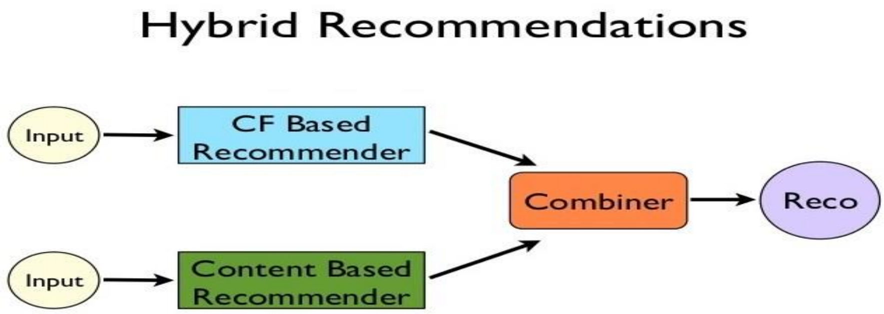

Fig 2.3: Hybrid recommender system

3. SEQUENCE DIAGRAM

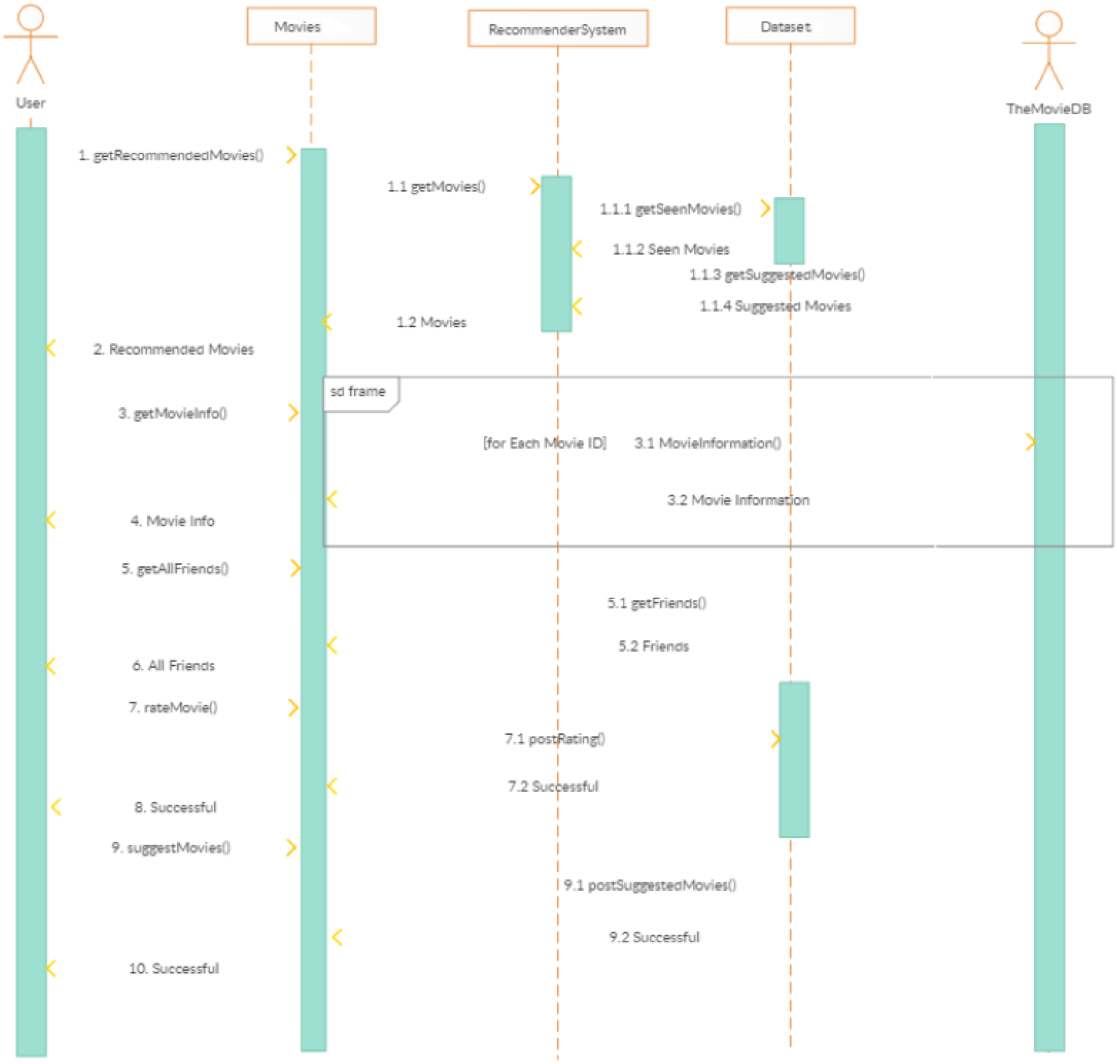

Fig 3.1 – Sequence Diagram of the recommendation system

4. Output

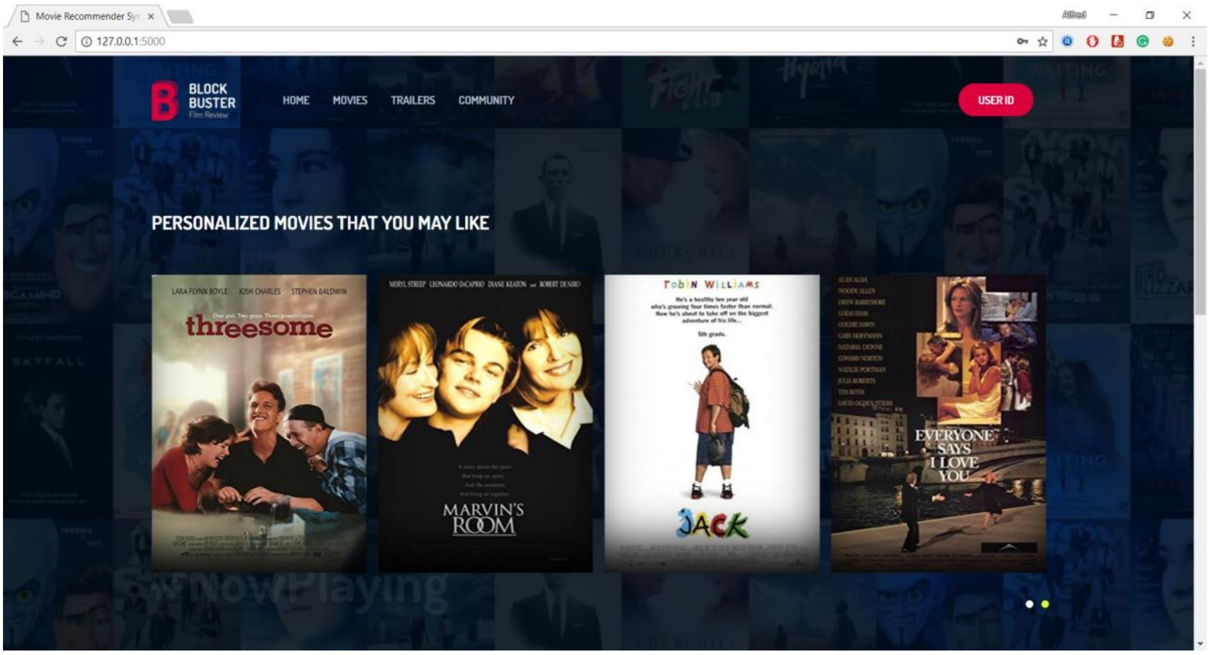

Fig 4.1 – Screenshot 1 – Homepage

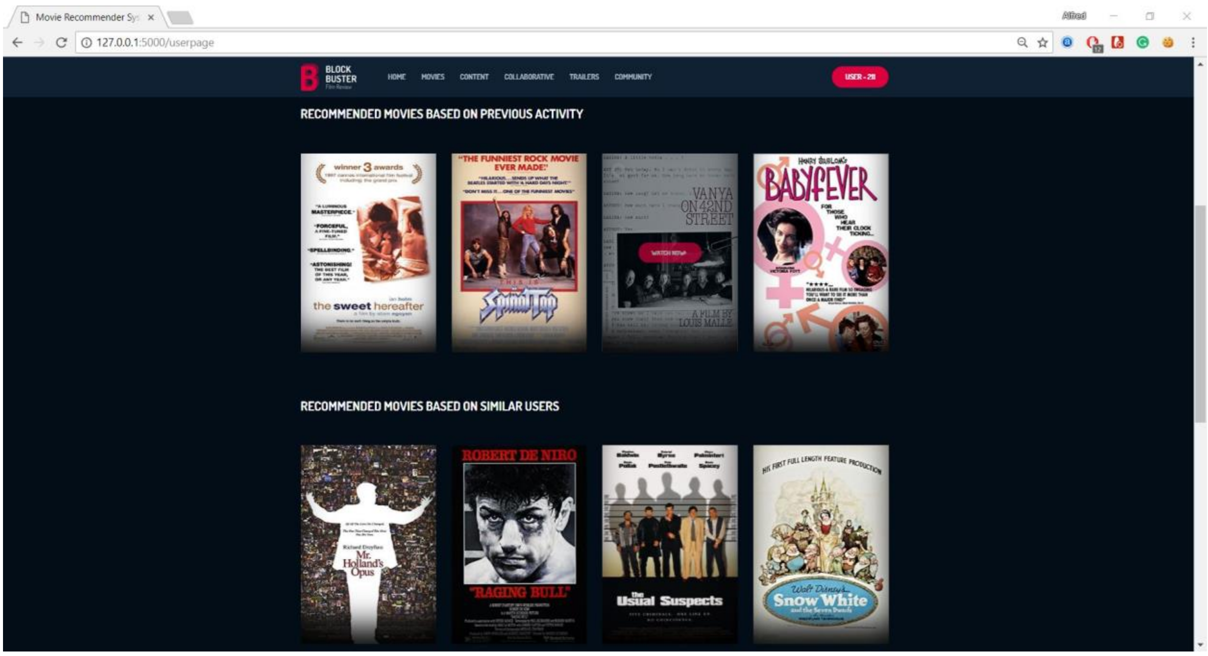

Fig 4.1 – Screenshot 1 – User Page with recommendations

4. CONCLUSION AND FUTURE SCOPE

The three recommender systems have their advantages and disadvantages in performing their job. Most of the limitations in each one of the approaches can be complimented by the other. A good recommender system should be able to provide positive and relevant recommendations from time to time and also provide alternative recommendations to break the fatigue of the users seeing the same items in the recommendation list. Future recommendation systems should be dynamic, and the profiles should be able to be updated in real time. This and the synchronization of various profiles implies the need of huge amount of computational power, network bandwidth etc. Current algorithms and techniques all have relatively high memory computational complexity, and that leads to long system processing time and data latency. Therefore, new algorithms and techniques that can reduce memory computational complexity eventually eliminate synchronization problems will be the one of development orientations.

5.REFERENCES

1. Francesco Ricci and Lior Rokach and Bracha Shapira, Introduction to Recommender Systems Handbook, Recommender Systems Handbook, Springer, 2011, pp. 1-35.
2. Xiaoyuan Su, Taghi M. Khoshgoftaar, A survey of collaborative filtering techniques, Advances in Artificial Intelligence archive, 2009.
3. Pasquale Lops, Marco de Gemmis, Giovanni Semeraro, Content-based Recommender Systems: State of the Art and Trends, Recommender Systems Handbook, Springer, Pages 73-105, 2011.
4. Burke, R.: Hybrid web recommender systems. In: The Adaptive Web, Springer Berlin / Heidelberg, Pages. 377–408, 2007.