International Journal on Recent and Innovation Trends in Computing and Communication ISSN: 2321-8169
Volume: 4 Issue: 12 206 – 209

Movie Recommendations using Hybrid Recommendation Systems

Khyati Aggarwal
Department of Information Technology
Sardar Patel Institute of technology,
Mumbai, India
khyatiaggarwal@gmail.com

Yashowardhan Soni
Department of Information Technology
Sardar Patel Institute of technology,
Mumbai, India
soniyashowardhan@gmail.com

Gaurav Jain
Department of Information Technology
Sardar Patel Institute of technology,
Mumbai, India
gaurav199494@gmail.com

Abstract- A recommendation system for movies 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, and personal information. Although, there are many recommendation systems present, new recommender system technologies are needed that can quickly produce high quality recommendations, even for very large-scale information resources. The proposed system has the ability to recommend movies, to a new user as well as the others by using their Facebook data. It collects all the important information, such as, popularity, liking and disliking, required for recommendation. It also takes minimal information from new user without social network login. It generates recommendations for the user based on his/her behavior on social media. This system could also be used by movie producers to get an idea about the response of their upcoming movies since all the data is extracted from the user's behaviors.

Keywords - *Movie; Recommender System; Social Network.*

*****

I. INTRODUCTION

With the rise in the availability of online products, consumers today face the problem of choice.

Especially in case of Movies, wherein a number of movies are released worldwide, the audience long for the best ones which suit their interests. Personalization in choices is what a sophisticated system needs to satisfy consumer demands. Content-based and collaborative are two main paradigms used to recommend items to user based on their interest. Our system proposes a hybrid of the two to predict, rate and rank movies for the user. The system is implemented in Python and uses a Web based UI to get user data and display top ranked movies.

II. BACKGROUND AND RELATED WORK

Since the last decade, there has been a substantial increase in demand for efficient recommender systems. E-commerce websites have widely used recommender systems in order to catalogue items according to customer interests. Some systems link the customer's social media accounts in order to get personal data and provide recommendations accordingly. The approach used by these systems may be collaborative, content-based or hybrid.

Last.fm is a music streaming service which recommends music tracks to users by observing the bands or other attributes of the songs previously played by the user. Then the attributes are compared to other users' and users with similar interests are found. The songs are then recommended through the playlist of the similar users. This is collaborative filtering approach.

Another music streaming service, Pandora, uses content-based approach. It creates a station which has songs with similar attributes. The user can rate a song as 'liked' or 'disliked' depending on the interest. The feedback is then used to improvise and modify the list.

MovieLens is a popular online movie recommendation system by Grouplens Research Group, which asks user to rate a few movies on the first login. Through these ratings, the movies are recommended to the user. Collaborative recommendations are also provided based on the ratings of similar users. Grouplens also provides an extensive dataset for movies and ratings in different sizes.

III. METHODOLOGY

A. Data Extraction

Movie data was extracted using the OMDB API in JSON format. The JSON data was later converted to CSV format in order to import it in the database. Movie details included the title, cast, language, rating on IMDB, director, plot, poster url etc.

B. Facebook Login

User data was extracted from Facebook using the Graph API and PHP. User's age, gender, language, liked movies etc. were extracted after the user logged in using Facebook. User's posts and comments containing movie tags were also collected. Natural Language processing is performed on the posts and comments, to know if the post is positive or negative. According to the data obtained from the processing, the database is updated.

C. Login without social media

The user has to fill a registration page with details like Name, Age, Gender, and Profession. The user also has an option to select a few genres that the user likes. This information about the user is used to solve the cold start problem that may arise. Thus, users having a cold start are firstly recommended movies popular in their own age group. Secondly, if the user has specified some genres that the user likes, content based recommendations are provided to the user.

D. Collaborative Recommendation

Collaborative Filtering is a recommendation approach wherein the ratings of the peers are taken into consideration. Thus, the predictions of a user's liking for a particular movie are generated by taken into account the ratings of other users which are similar to the user's rating. Pearson Correlation Coefficient is used to find similarity between users. This coefficient varies from -1 to +1, where -1 indicates dissimilar users and +1 indicates similar users.

$$sim(a,b) = \frac{\sum_{p \in P}(r_{a,p} - \bar{r}_a)(r_{b,p} - \bar{r}_b)}{\sqrt{\sum_{p \in P}(r_{a,p} - \bar{r}_a)^2}\sqrt{\sum_{p \in P}(r_{b,p} - \bar{r}_b)^2}}$$

Where r is the similarity measure.

The similarity obtained is used as a weight for calculating the rating of the unrated movie. The rating is predicted using the following function:

$$pred(a,p) = \overline{r_a} + \frac{\sum_{b \in N} sim(a,b) * (r_{b,p} - \overline{r_b})}{\sum_{b \in N} sim(a,b)}$$

A python package named "mrjob" was used to perform MapReduce in python.

One of the issues involved with collaborative filtering is that it requires all the rated items in order to consider them for ranking. If only a few rated items are available, the recommendation will be done with respect to the small set.

E. Content-based Recommendation

Users profile is created i.e. the information about the user like (Name, Age, Liked Genres, Liked Movies, Liked Actors, etc.). Also, the properties of movie i.e. (Similarity and overlapping of the genres, overlapping of the Actors and directors, Similarity of the movie plots) are considered. The user is rated movies based on his user profile. Thus first movies are filtered by the genres liked by the user, followed by filtering these movies by the favorite actors and directors of the user, thereby giving the user recommendations based on what the user has liked before. The movies were filtered according the plot using similarity. Content based has few issues, such that it cannot do an in depth analysis of movie, for example the scenes or the tone of the movie are not taken into consideration. Content based gives recommendation only according to user profile. Therefore user is forced to see the movies only similar to his experience.

F. Hybrid Recommendation

Hybrid Recommendations involves combining traditional approaches i.e. Collaborative Filtering and Content Based Recommendations to overcome the drawbacks of one of the approaches. In a Pipelined Hybrid Recommendation System, the input is passed on to a recommender system and the output generated by this system is passed as input to the next recommender system. For example, first the movie recommendations are found out by collaborative filtering, and then these recommendations are passed to the content based recommender to generate movie recommendations which are similar in content. This approach allows the system to recommend movies that should be rated high but are not recommended by collaborative filtering approach since no or very few users have rated this movie.

G. Natural Language Processing

The Facebook user login helps extraction of user posts related to movies. The posts containing the tagged movie pages are considered for Natural Language Processing (NLP). The weight age to the every term (word, phrase and emoticons) is given which is used to perform NLP and calculate the approximate rating (disliking and liking) for a movie. Python NLTK library is used to perform the semantic analysis.

(Refer Figure 1)

IV. CHALLENGES

Perhaps the biggest issue facing recommender systems is that they need a lot of data to effectively make recommendations. It is no coincidence that the companies most identified with having excellent recommendations are those with a lot of consumer user data: Google, Amazon, Netflix, and Last.fm. A

good recommender system firstly needs item data (from a catalog or other form), then it must capture and analyze user data (behavioral events), and then the magic algorithm does its work. The more item and user data a recommender system has to work with, the stronger the chances of getting good recommendations. But it can be a chicken and egg problem – to get good recommendations, you need a lot of users, so you can get a lot of data for the recommendations.

The recommender involves extraction of user data from social network account, mainly Facebook. However, in the last few years, Facebook has limited the access to user data through the Graph API. On querying for the user's friend list, only the list of friends with developer account is displayed. The troublesome part for the user is to rate the movies. Since the system mainly depends upon the rating given by the user, having the movies rated is a need. But users may not want to rate movies always.

H. The cold start problem

The cold start problem appears in two variants. The first variant is related to new users. Whenever a user logs into the system for the first time, the system does not have information about the preferences of that user to provide the recommendations. The second variant is related to new items. When a new item is added to the system, it doesn't have the user ratings. Due to lack of ratings, the item may be placed lower in the recommended items list. To calculate the similarities for neighborhood, the historic ratings are necessary, and they are unavailable in the cold-start scenario. This makes it difficult to predict ratings for items.

I. Movie data Collection

An extensive dataset is required for a recommender system to be efficient and accurate. Services like Google, Last.fm, Pandora, Netflix etc. provide accurate recommendations through the use of extensive and reliable dataset. The behavior of a real system can only be validated and verified through a reliable dataset.

The datasets freely available online consisted of large number of ratings, but they lacked the required features about the movie for the recommendation systems. Hence the dataset had to be preprocessed and modified before being implemented into the system.

V. RESULTS

The user has to log into the system using the Facebook login or through the registration page. The registration page consists of minimal compulsory questions (age, gender and profession) and remaining optional questions. After login, the user can browse through the catalogue of movies. The collaborative recommendations section predicts the ratings calculated using the Pearson correlation coefficient and displays the top ranked movies accordingly.

The Content based recommendations section compares the feature vector of the user profile with the features of the movies. The Hybrid Recommendations section gets the movie array from the collaborative section and passes it as an input to the content based module. The movie array is then refined accordingly and displayed. The proposed system is better than the current systems as minimal user input is taken and quality recommendations are provided. The feedback of the user to these recommendations is an important factor which helps the system decide how accurate the recommendations are.

VI. CONCLUSION

This paper explains how quality recommendations can be given to the user by taking minimum input from users. The system consists of Collaborative Filtering, Content Based Recommendations and Hybrid Recommendations which provide recommendations to the user. Hybrid Recommendations is one of the main modules of the system which helps overcome the drawbacks of the traditional Collaborative and Content Based Recommendations. We have obtained promising results using our current model. The system can further be extended to recommend music, books, web services etc. The rating demographics can be compiled for the directors, producers in order to improvise movies.

FUTURE WORK

We plan to extend the model and incorporate graph database to link user profiles. Using a larger dataset of movies, the system will be able to provide accurate results

REFERENCES

[1] Eyrun A. Eyjolfsdottir, Gaurangi Tilak, Nan Li - MovieGEN: A Movie Recommendation System

[2] Christakou, C;Lefakis, L. ; Vrettos, S. ; Stafylopatis, A. - A Movie Recommender System based on Semi-supervised Clustering, Computational Intelligence for Modelling, Control and Automation, 2005 and International Conference on Intelligent Agents, Web Technologies and Internet Commerce, International Conference on (Volume:2).

[3] Shinahyun Ah,Chung-Kon Shi - Exploring Movie Recommendation System using cultural metadata, Cyberworlds, 2008 International Conference on.

[4] Ashraf M.Z, Chouwdhary D.K., Das R.L., Ghosal P. - An efficient and optimized recommendation system using social network knowledge base, Advances in Electrical Engineering (ICAEE), 2014 International Conference.

ISSN: 2321-8169
206 – 209

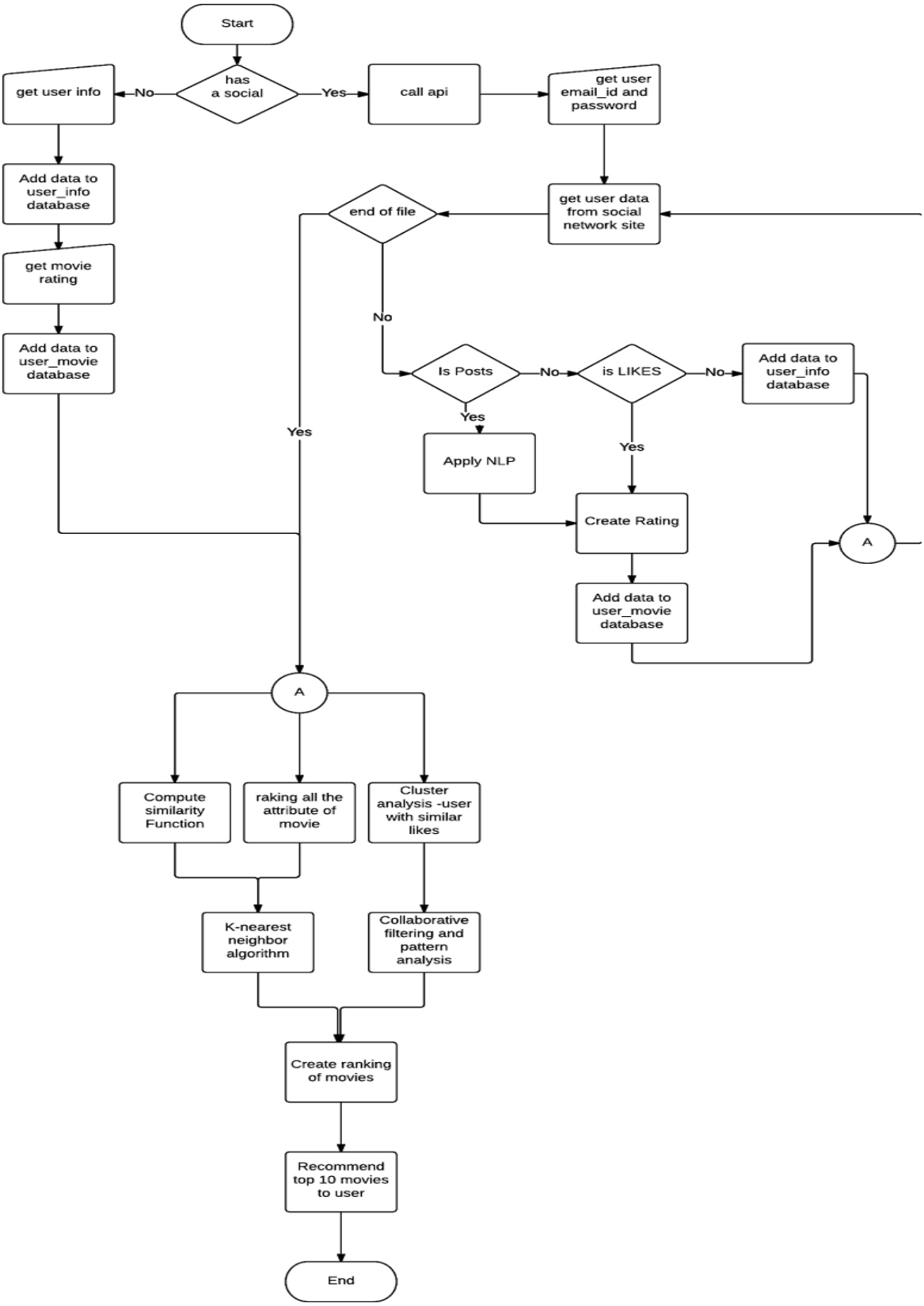

Figure 1. Flowchart for the system