Movie Recommendation System Using Item Based Collaborative Filtering

Poonam Sharma, Lokesh Yadav

ABSTRACT- In today's digital world where there is an endless variety of content consumed such as books, videos, articles, Films, etc., finding material of one's choice has become an infallible task. Digital content on the other hand Providers want to engage more and more users in their service for maximum time. Where is it the recommender system comes into picture where content providers advise users by content User choice in this paper we have proposed a movie recommendation system .Purpose of movie recommendation system aims to provide users with accurate movie recommendations. Usually basic recommendation system to make recommendations consider one of the following factors; User preference known as content based Filtering or the preference of similar users known as collaborative filtering. To create a stable and accurate recommender system will use of content based filtering.

KEYWORDS- Movies, Recommendation system, CBF-Content-based filtering, CF- Collaborative filtering

I. INTRODUCTION

Recommendation system helps users to find and select items (e.g. books, movies, restaurants) from a large Number available on the web or other electronic information sources. Given a large set of objects and a Describing user needs, they offer the user a small set of items that are well suited description. Similarly, a film recommendation system provides a level of comfort and personalization Helps users to interact better with the system and watch movies that meet their needs. Providing this level of User comfort was our primary motivation in opting for the film recommendation system as our BE project. The main objective of our system is to recommend movies to our users based on their viewing history and ratings they provide. The system will also recommend various e-commerce companies to promote their products[1]. Depending on

the genre of films of their choice to specific customers. Personalized recommendation engines help Collaborative filtering and Content-based filtering is the predominant approach for providing recommendations to users[2]. They are both the best the specific scenarios are applied due to their respective fluctuations. Figure 1 & 2 show the examples of recommendation systems.
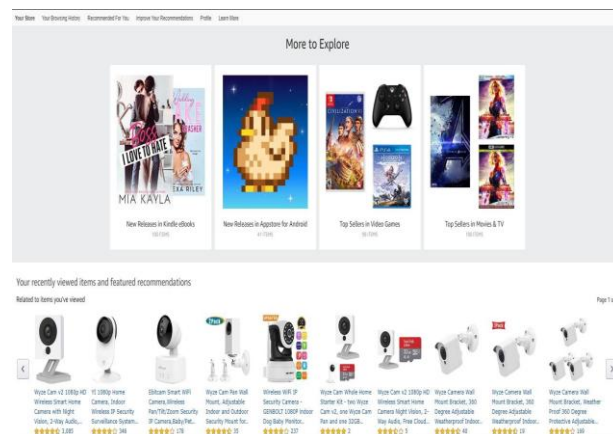

Fig 1: Recommended product on Amazon

Fig 2: Recommended movies on Netflix

II. OBJECTIVE

The Movie Recommendation System provides a mechanism to help users categorize users with similar interests. Basically the purpose of a recommendation system is to search for material that will be interesting to a person. Moreover, it involves a number of factors to create personalized lists of useful and interesting content specific to each user/individual. Recommendation

systems are Artificial Intelligence based algorithms that skim through all possible options and create a customized list of items that are interesting and relevant to an individual. These results are based on their profile; search/browsing history, what other people with similar traits/demographics are watching, and how likely are you to watch those movies. This is achieved by applying item-based collaborative filtering.

III. RELATED WORK

D.K. Yadav presented the movie recommendation system a MOVREC[10] which is based on collaborative filtering approach. Collaborative filtering makes use of information provided by user. That information is analyzed and a movie is recommended to the users which are arranged with the movie with highest rating first.

The system also has a provision for user to select attributes on which he wants the movie to be recommended. Luis M Capos et al. [5] has analyzed two traditional recommender systems i.e. content based filtering and collaborative filtering. As both of them have their own drawbacks he proposed a new system which is a combination of Bayesian network and collaborative filtering. The proposed system is optimized for the given problem and provides probability distributions to make useful inferences. A hybrid system has been presented by Harpreet Kaur et al. [9]. A mix of collaborative as well as content filtering algorithm is used by the system. The user specific information or item specific information is clubbed to form a cluster by Utkarsh Gupta et al. [12] using chameleon. This is an efficient technique based on Hierarchical clustering for recommender system. Urszula Kużelewska et al. [6] proposed clustering as a way to deal with recommender systems. Two methods of computing cluster representatives were presented and evaluated. Centroid-based solution and memory-based collaborative filtering methods were used as a basis for comparing effectiveness of the proposed two methods.

The result was a significant increase in the accuracy of the generated recommendations when compared to just centroid-based method. Costin-Gabriel Chiru et al. [3] proposed Movie Recommender, a system which uses the information known about the user to provide movie recommendations. This system attempts to solve the problem of unique recommendations which results from ignoring the data specific to the user. To predict the difficulty level of each case for each trainee Hongli LIn et al. proposed a method called contentboosted collaborative filtering (CBCF). There are various types of recommender systems with different approaches and some of them are classified in this paper:

IV. PROCEDURE AND RECOMMENDATIONS

A. Content-Based Filtering

In content-based filtering, items are recommended based on comparisons between item profile and user profile[7]. A user profile is content that is found to be relevant to the user in form of keywords (or features). A set of assigned keyword (terms, feature) might be seen by user profile which

algorithm collects from the user interest. A set of keywords (or features) of an item is the Item profile. For example, consider a scenario in which a person goes to buy his favorite cake 'X' to a pastry. Unfortunately, cake 'X' has been sold out and as a result of this the shopkeeper recommends the person to buy cake 'Y' which is made up of ingredients similar to cake 'X'.
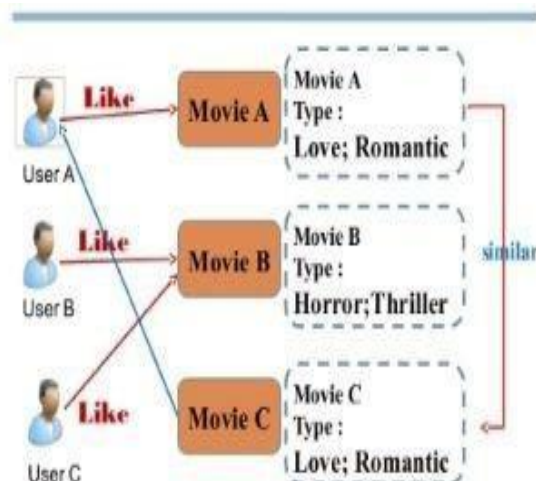

Fig 3: Recommended movies on Netflix

We will be using the cosine similarity to calculate a numeric quantity that denotes the similarity between two movies. We use the cosine similarity score since it is

$$\text{similarity} = \cos(\theta) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\|\|\mathbf{B}\|} = \frac{\sum\limits_{i=1}^{n} A_i B_i}{\sqrt{\sum\limits_{i=1}^{n} A_i^2}\sqrt{\sum\limits_{i=1}^{n} B_i^2}},$$

independent of magnitude and is relatively easy and fast to calculate. Mathematically, it is defined as follows:

These are the following steps we'll follow:-
- Get your index of the movie.
- Get a list of cosine similarity scores for that particular film with all films.
- Convert it into a list of tuples where the first element is its position and the second is the similarity score.
- Sort the aforementioned list of tuples based on the similarity scores; that is, the second element.
- Get the top 10 elements of this list.
- Return the titles corresponding to the indices of the top elements.

Advantages of content-based filtering are:
- They capable of recommending unrated items.
- We can easily explain the working of recommender system by listing the Content feature of an item.
- Content-based recommender system use needs only the rating of the concerned user, and not any other user of the system.

Disadvantages of content-based filtering are:

- It does not work for a new user who has not rated any item yet as enough rating are required content based recommender evaluates the user preferences and provide accurate recommendations.
- No recommendation of serendipitous items.
- Limited Content Analysis- The recommend does not work if the system fails to distinguish the items that a user likes from the items that he does not like.

B. Collaborative filtering based systems

Content based engine algorithm has several limitations. It is only capable of suggesting movies which are close to a certain movie. Also, the engine that we built is not really personal in that it doesn't capture the personal tastes and biases of a user [11]. Collaborative filtering system recommends items based on similarity measures between users and/or items. The system recommends items preferred by similar users. This is based on the scenario where a person asks his friends, who have similar tastes, to recommend him some movies.
It is basically of two types:-

C. User based filtering

These systems recommend products to the user that similar users may have liked. We can use either Pearson correlation or cosine similarity to measure similarity between two users. Each row represents a user and the column represent the different movie in the matrix and the last one record that show the similarity between the user and the target user is except in the matrix. The rating which is given by the user to that movie is represented by each cell. Although computing user-based Collaborative filtering is very simple, it suffers from several problems. A main issue is that users' preferences may change over time[8]. It indicates that precompiling the matrix based on their neighboring users may lead to bad performance. To tackle this problem, we can apply item-based Collaborative filtering.
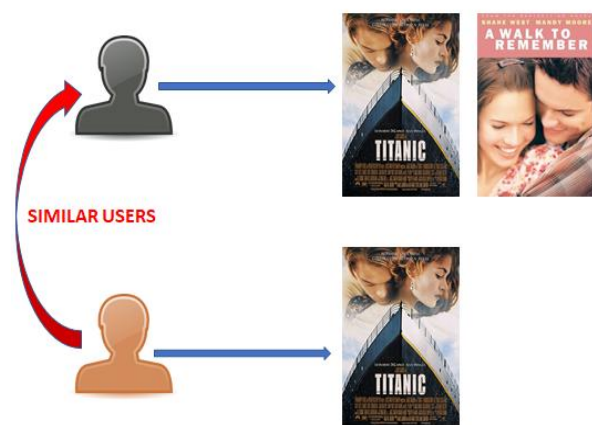

Fig 4: User based filtering

D. Item based filtering

Instead of measuring the similarity between users, the item-based Collaborative filtering recommends items based on their similarity with the items that the target user rated. Likewise, the similarity can be computed with Pearson Correlation or Cosine Similarity[4]. The major difference is that, with item- based collaborative filtering, we fill in the blank vertically, as oppose to the horizontal manner that user-based Collaborative filtering does. It successfully avoids the problem posed by dynamic user preference as item-based Collaborative filtering is more static. However, several problems remain for this method. First, the main issue is scalability. The computation grows with both the customer and the product. O(mn) with m users and n items is the worst case complexity. In addition, sparsity is another concern.
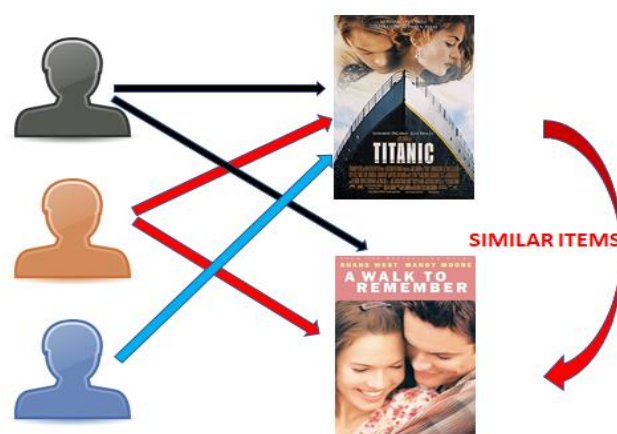

Fig 5: User based filtering

Advantages of collaborative filtering based systems:

- It is dependent on the relation between users which implies that it is content-independent.
- They can make real quality assessment of items by considering other peoples experience

Disadvantages of collaborative filtering are:

- Initial rater problem: Collaborative filtering systems cannot provide recommendations for new items because there is no user rating on which to base a prediction.
- Gray Sheep: For a CF based system to work; a group with similar characteristics is required. Even if such groups exist, it will be very difficult to recommend users who do not consistently agree or disagree to these groups.

V. RESULTS AND CONCLUSIONS

Recommended systems open up new opportunities to obtain personal information over the Internet. It also helps reduce the problem of information overload which is a very common phenomenon with information retrieval systems and allows users to access products and services which are not readily available to users on the system.
We come up with a strategy that focuses on dealing with user's personal interests and based on his previous reviews, movies are recommended to users. This strategy helps in improving accuracy of the recommendations. A

personal profile is created for each user, where each user has access to his own history, his likes, ratings, comments, password modification processes. It also helps in collecting authentic data with improved accuracy and makes the system more responsive.

This approach overcomes the drawbacks of each individual algorithm and improves system performance. Techniques such as clustering, similarity, and classification are used to obtain better recommendations, thus reducing MAE and increasing accuracy and precision.
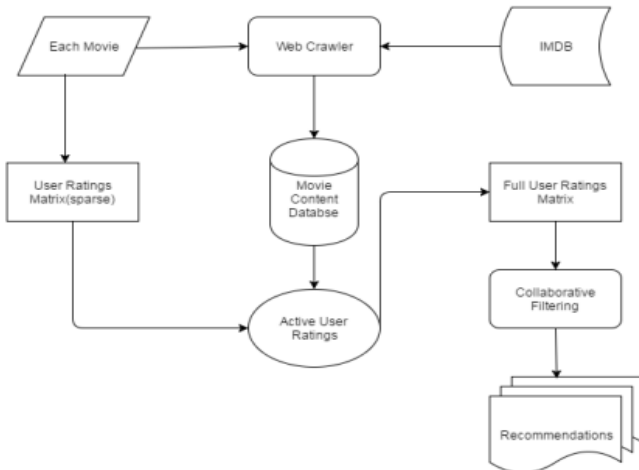
Fig 5: System Overview

```
ratings_mean_count_df['mean'].plot(bins=100, kind='hist', color = 'r')
```
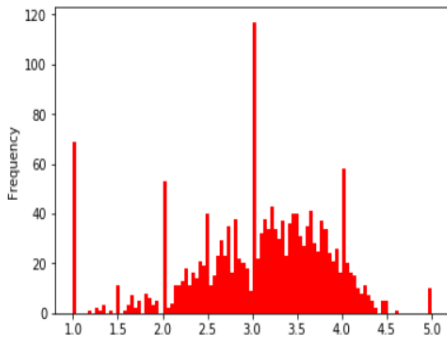```
<matplotlib.axes._subplots.AxesSubplot at 0x213f8cbe7b8>
```



Fig 6: Rating mean graph

```
ratings_mean_count_df['count'].plot(bins=100, kind='hist', color = 'r')
```
```
<matplotlib.axes._subplots.AxesSubplot at 0x213f8dbb6d8>
```
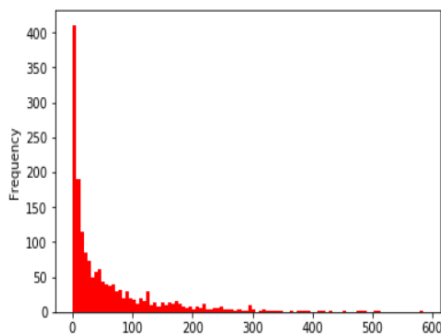


Fig 7: Rating count graph

```
In [39]: movie_correlations = userid_movietitle_matrix.corr(method = 'pearson', min_periods = 80)
         # pearson : standard correlation coefficient
         # Obtain the correlations between all movies in the dataframe
```
```
In [40]: movie_correlations
```



Fig 8: Pearson method

```
titanic_correlations[titanic_correlations['count']>80].sort_values('Correlation',ascending=False).head()
```

title	Correlation	count
Titanic (1997)	1.000000	350.0
River Wild, The (1994)	0.497600	146.0
Abyss, The (1989)	0.472103	151.0
Bram Stoker's Dracula (1992)	0.443560	120.0
True Lies (1994)	0.435104	208.0

Fig 9: Correlation of movies with count values

In future we can work on hybrid recommender using clustering and similarity for better performance. Our approach can be further extended to other domains to recommend songs, video, venue, news, books, tourism and e-commerce sites, etc.

REFERENCES

[1] James Bennett, Stan Lanning ; "The Netflix Prize", In KDD Cup and Workshop in conjunction with KDD,2007

[2] Mohammad Yahya H. Al-Shamri , Kamal K. Bharadwaj; "A Compact User Model for Hybrid Movie Recommender System " in International Conference on Computational Intelligence and Multimedia Applications 2007

[3] Costin-Gabriel Chiru, Vladimir-Nicolae Dinu , Ctlina Preda, Matei Macri ; "Movie Recommender System Using the User's Psychological Profile" in IEEE International Conference on ICCP, 2015.

[4] Christina Christakou, Leonidas Lefakis, Spyros Vrettos and Andreas Stafylopatis; "A Movie Recommender System Based on Semi-supervised Clustering ", IEEE Computer Society Washington, DC, USA 2015.

[5] Luis M. de Campos, Juan M. Fernández-Luna *, Juan F. Huete, Miguel A. Rueda-Morales; "Combining content-based and collaborative recommendations: A hybrid approach based on Bayesian networks",

International Journal of Approximate Reasoning, revised 2010.

[6] Urszula Kużelewska; "Clustering Algorithms in Hybrid Recommender System on MovieLens Data", Studies in Logic, Grammar and Rhetoric, 2014.

[7] Dietmar Jannach, Gerhard Friedrich; "Tutorial: Recommender Systems", International Joint Conference on Artificial Intelligence, Beijing, August 4, 2013.

[8] Gaurangi, Eyrun, Nan; "MovieGEN: A Movie Recommendation System", UCSB.

[9] Harpreet Kaur Virk, Er. Maninder Singh," Analysis and Design of Hybrid Online Movie Recommender System "International Journal of Innovations in Engineering and Technology (IJIET)Volume 5 Issue 2,April 2015.

[10] Manoj Kumar, D.KYadav, Ankur Singh, Vijay Kr. Gupta," A Movie Recommender System: MOVREC" International Journal of Computer Applications (0975 – 8887) Volume 124 – No.3, August 2015.

[11] Prerana Khurana , Shabnam Parveen; 'Approaches of Recommender System: A Survey'; International Journal of Computer Trends and Technology (IJCTT) – Volume 34 Number 3 - April 2016.

[12] Utkarsh Gupta1 and Dr Nagamma Patil2," Recommender System Based on Hierarchical Clustering Algorithm Chameleon" 2015 IEEE International Advance Computing Conference(IACC).