Movie Recommender Systems via

User- Item Subgroups

**ABSTRACT**

Collaborative filtering (CF) is one of the most successful recommendation approaches. It typically associates a user with a group of like-minded users based on their preferences over all the items, and recommends to the user those items enjoyed by others in the group. However we find that two users with similar tastes on one item subset may have totally different tastes on another set. In other words, there exist many user-item subgroups each consisting of a subset of items and a group of like-minded users on these items. It is more natural to make preference predictions for a user via the correlated subgroups than the entire user-item matrix. Our approach can be seen as an extension of traditional clustering CF models. Systematic experiments on three real world data sets have demonstrated the effectiveness of our proposed approach.

**EXISTING SYSTEM**

Collaborative filtering techniques have been successful in enabling the prediction of user preferences in the recommendation systems. There are three major processes in the recommendation systems: object data collections and representations, similarity decisions, and recommendation computations. Collaborative filtering aims at finding the relationships among the new individual and the existing data in order to further determine the similarity and provide recommendations. To improve correlation-based collaborative filtering and performing clustering on movie ratings from users. The ratings are extracted from movie Recommendations.

**PROPOSED SYSTEM**

Collaborative filtering (CF) is one of the most widely adopted and successful recommendation approaches. Unlike many content-based approaches which utilize the attributes of users and items, CF approaches make predictions by using only the user-item interaction information. These methods can capture the hidden connections between users and items and have the ability to provide serendipitous items which are helpful to improve the diversity of recommendation. Recommender systems have been indispensable nowadays due to the incredible increasing of information in the world, especially on the Web. These systems apply knowledge discovery techniques to make personalized recommendations that can help people sift through huge amount of available articles, movies, music, web pages, etc. Popular examples of such systems include product recommendation in Amazon, music recommendation in Last.fm, and movie recommendation in Movie lens.

**MODULE DESCRIPTION:**

# **Number of Modules**

After careful analysis the system has been identified to have the following modules:

1. **Recommendation System Module**
2. **Collaborative Filtering Module.**
3. **Information Filtering Module.**

**1. Recommendation System Module:**

The website bases its recommendations on what the user provides to the website, including films the user likes, and what films other users with similar tastes prefer. When a user joins the website, they are given several randomly chosen movies and told to rate them from one to five stars, five being the best. The system then compares the user's ratings to those of other users with similar tastes, and then accordingly recommends films that the user has not yet seen. When a user looks for a specific film on the website, the system returns a prediction of what it believes the user will rate the film after watching it. The website suggests that users rate as many films that they have seen as possible so that the recommendations given will be more accurate, since the system would have a better sample of the user's film tastes. A recommendation based on the same data structure as user-item matrix having users and items consisting of their rating scores.

**2.Collaborative Filtering Module:**

Collaborative filtering (CF) is one of the most successful recommendation approaches. There are two methods in CF as user based collaborative filtering and item based collaborative filtering User based CF assumes that a good way to find a certain user’s interesting item is to find other users who have a similar interest. So, at first, it tries to find the user’s neighbors based on user similarities and then combine the neighbor users rating scores, which have previously been expressed, by similarity weighted averaging. And item based CF fundamentally has the same scheme with user based CF. It looks into a set of items; the target user has already rated and computes how similar they are to the target item under recommendation. After that, it also combines his previous preferences based on these item similarities.

**3.Information Filtering Module:**

Information filtering has two main methods. One is the content based filtering and the other is the collaborative filtering. Collaborative filtering (CF) has proved to be one of the most effective for its simplicity in both theory and implementation. While filtering does make it easier for users to acquire information which is most relevant to them, it gives power to the advertisers and other corporations. Filtering through complex algorithms is a good way of getting the most relevant information to the user easily. Collaborative information filtering systems, have been implemented as to recommend to the user what is probably to be interesting for him or her. This is done based on the ratings that other correlated users have assign to the same object. Usually this idea has been developed for specific domains, like “Music” or “Films” as for introducing people.

**SOFTWARE REQUIREMENTS**:

Operating System : Windows

Technology : Java and J2EE

Web Technologies : Html, JavaScript, CSS

IDE : My Eclipse

Web Server : Tomcat

Tool kit : Android Phone

Database : My SQL

Java Version : J2SDK1.5

**HARDWARE REQUIREMENTS**:

Hardware : Pentium

Speed : 1.1 GHz

RAM : 1GB

Hard Disk : 20 GB

Floppy Drive : 1.44 MB

Key Board : Standard Windows Keyboard

Mouse : Two or Three Button Mouse

Monitor : SVGA