Available Online at www.ijcsmc.com

International Journal of Computer Science and Mobile Computing

A Monthly Journal of Computer Science and Information Technology

ISSN 2320–088X

IJCSMC, Vol. 3, Issue. 4, April 2014, pg.765 – 770

RESEARCH ARTICLE

MOVIE RECOMMENDATION SYSTEM BASED ON USERS' SIMILARITY

Gaurav Arora[1], Ashish Kumar[2], Gitanjali Sanjay Devre[3], Prof. Amit Ghumare[4]

[1,2,3]Student, Computer Engineering Department

[4]Assistant Professor, Computer Engineering Department

[1,2,3,4]Dr. D. Y. Patil College of Engineering, Ambi University of Pune

[1] K243.gsa@gmail.com, [2] ashishkumar00071@gmail.com, [3] geetpatil2@gmail.com, [4] amit.ghumare@gmail.com

Abstract – The growth of e-Commerce has given the birth to recommendation engine. There are many recommendation engines existing in the market to recommend different stuffs to the users. These recommendations are based on different aspects such as interest of users, history of users, location of users and many more. In all the above aspects, one thing is common and that is individuality. The engine recommends users on the basis of users' perspective; but there are things in market which are worth concerned and that a user is unaware of. These things must also recommended to the users by the engine; but due to the limitation of "individuality", these engines do not recommend things that are out of the box. The hybrid movie recommendation engine has overcome this limitation of individuality. The engine will recommend movies to the users as per their interest as well as it will recommend movies rated by other users who are similar to the user. In addition to that, there are web services which will act as an adornment to the application.

Keywords – Recommendation system, Machine learning, Mahout, Euclidean algorithm, Web Services, Precise recommendation

I. INTRODUCTION

As we know that, the world is growing faster like never before. Everyone is rushing for their ultimate goals. This thirst results into the development of almost every sector. Online business is one of them. We people, don't have time to shop from market and this is not the end. We don't even have time to choose the object from the collection .This created the embryo of online shopping, which nowadays, became a huge tree, of tons of branches.

As the online market grows exponentially, it's obvious that competition will entered in this field also. Now, owners of their respective sites need to attract their users by providing attractive facilities. Recommender Engines is one of the facilities given to users.

Recommender engine are the most immediately recognizable machine learning technique in use today. We will have seen services or sites that attempt to recommend books or movies or articles based on our past actions. They try to infer tastes and preferences and identify unknown items that are of interest.

Amazon.com is perhaps the most famous e-commerce site to deploy recommendations based on purchases and site activity, Amazon recommends books and other items likely to be of interest. Netflix similarly recommends DVDs that may be of interest, and famously offered a $1,000,000 prize to researchers who could improve the quality of their recommendations. Social networking sites like Facebook use variants on recommender techniques to identify people most likely to be as-yet-unconnected friends.

II. PROBLEM STATEMENT

This paper is based on recommendation system that recommends different things to users. This system will recommend movies to users. This system will provide more precise results as compared to the existing systems. The existing system works on individual users' rating. This may be sometime useless for the users who have different taste from the recommendations shown by the system as every user may have different tastes. This system calculates the similarities between different users and then recommend movie to them as per the ratings given by the different users of similar tastes. This will provide a precise recommendation to the user. This is a web based as well as android system where there is a movie web service which provides services to user to rate movies, see recommendations put comments and see similar movies.

III. LITERATURE SURVEY

A. BACKGROUND

Over the past decade, a large number of recommendation systems for a variety of domains have been developed and are in use. These recommendation systems use a variety of methods such as content based approach, collaborative approach, knowledge based approach, utility based
approach, hybrid approach, etc.

Most of the online recommendation systems for a variety of items use ratings from previous users to make recommendations to current users with similar interests. One such system was designed by Jung, Harris, Webster and Herlocker (2004) for improving search results. The system encourages users to enter longer and more informative search queries, and collects ratings from users as to whether search results meet their information need or not. These ratings are then used to make recommendations to later users with similar needs.

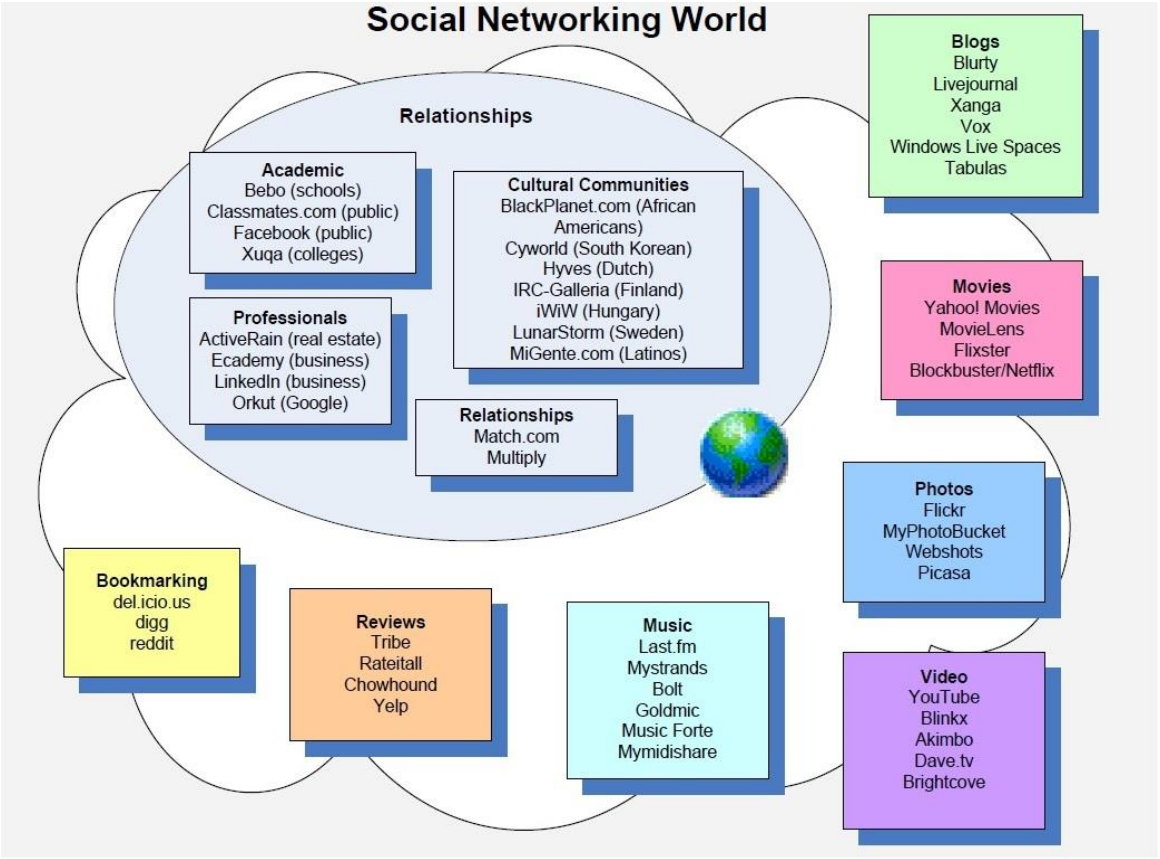
B. EXISTING PRODUCTS AND SYSTEMS

Current Social Networking World

Internet social networking sites, which began in 1995 with Classmates.com, have surged in popularity and use through word-of-mouth advertising. Since then, a wide range of virtual communities have formed serving different purposes and targeting varying niche audiences:

C. Social Movie Platforms

In particular, we've chosen to explore the movie niche as this is an area where our project can provide significant improvements compared to existing products and systems. Traditional movie websites (IMDB, AOL Movies) function by proving global user ratings on movies in their database. Movies are categorized by metadata such as genre, era, directors, and so on. Users can search for movies, browse lists and read reviews written by critics or other users. However, most of these services lack any personal recommendation system and haven't taken advantage of social-networking communities or crowd wisdom. Some websites, such as Blockbuster, do provide individualized recommendations based on a user's ratings but do not include any social networking component. Yahoo! Movies goes further and uses personal ratings to suggest movies currently playing in theatre, on TV, and out on DVD. It also draws upon its vast user base to give lists of similar movie fans, their ratings, and reviews. Other movie sites, like Flixster, take a different approach. Flixster forms web-based communities around movies and suggests movies to watch based on what your friends have rated.



IV. PROPOSED SYSTEM

The system is built on windows 2007 operating system. The system uses advanced java technology along with machine learning concepts. MySQL is used for storing data. This system uses three-tier architecture. The web service layer provides the android user to rate movies, view similar recommendations given by the system and comment on it.

The proposed system is a better system than any other existing systems. This system has added the positive features of existing systems and has overcome the drawbacks of existing systems. The system uses all the existing algorithms i.e. content based, context based and collaborative based algorithms. All these algorithms are combined to give more precise result. The following modules are developed as:

A. Admin

The system admin will add movie in a database, view movies and update it.

B. Recommendation Engine

This recommendation engine will calculate the similarities between the different users. On the basis of that similarities calculated, this engine will recommend movie to a user.

C. Movie Web Service

This will allow user to rate movies, comments on movies. This service will also show the movie recommendation to the users.

D. Android User

The android user can rate a movie, can comment on any movie, and can see similar movies recommended by other users who are similar to this user.

V. METHODOLOGY

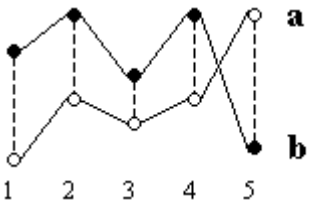
A. City Block Distance

The *City block distance* between two points, *a* and *b*, with *k* dimensions is calculated as:

$$\sum_{j=1}^{k} |a_j - b_j|$$

The *City block distance* is always greater than or equal to zero. The measurement would be zero for identical points and high for points that show little similarity.

The figure below shows an example of two points *a* and *b*. Each point is described by five values. The dotted lines in the figure are the distances (a_1-b_1), (a_2-b_2), (a_3-b_3), (a_4-b_4) and (a_5-b_5) which are entered in the equation above.



In most cases, this distance measure yields results similar to the *Euclidean distance*. Note, however, that with *City block distance*, the effect of a large difference in a single dimension is dampened (since the distances are not squared).

The name *City block distance* (also referred to as *Manhattan distance*) is explained if you consider two points in the xy-plane. The shortest distance between the two points is along the hypotenuse, which is the *Euclidean distance*. The *City block distance* is instead calculated as the distance in x plus the distance in y, which is similar to the way you move in a city (like Manhattan) where you have to move around the buildings instead of going straight through.

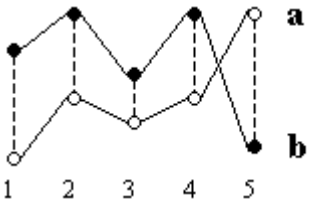
B. Euclidean Distance

The *Euclidean distance* between two points, *a* and *b*, with *k* dimensions is calculated as:

$$\sqrt{\sum_{j=1}^{k}(a_j - b_j)^2}$$

The Euclidean distance is always greater than or equal to zero. The measurement would be zero for identical points and high for points that show little similarity.

The figure below shows an example of two points *a* and *b*. Each point is described by five values. The dotted lines in the figure are the distances (a_1-b_1), (a_2-b_2), (a_3-b_3), (a_4-b_4) and (a_5-b_5) which are entered in the equation above.



VI. CONCLUSION AND FUTURE SCOPE

A recommendation system has been implemented based on hybrid approach of collaborative filtering engine and context based engine. We have tried to combine the existing algorithms for recommendation to come up with a hybrid one. It improves the performance by overcoming the drawbacks of traditional recommendation systems.

It describes the conventional Content, Collaborative Filtering and Context Filtering recommendation approaches along with their precision, recall and accuracy parameters. This paper has presented a number of utilized evaluation metrics, from which some were used to

measure quality, while others to measure performance. Recommender systems make the selection process easier for the users. Hybrid recommendation engine is a competent system to recommend Movies for e-users, whereas the other recommender algorithms are quite slow with inaccuracies. This recommender system will assuredly be a great web application, which can be clubbed with today's high demanding online purchasing web sites. Our approach can be extended to various domains to recommend books, music, etc.

REFERENCES

[1] G. Adomavicius and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," *IEEE Trans. Knowl. Data Eng.*

[2] G. Linden, B. Smith, and J. York, "Amazon recommendations: Itemto-item collaborative filtering," *IEEE Internet Comput.*, Feb. 2003.

[3] Michael Hashler, "Recommender Lab: A Framework for Developing and Testing Recommendation Algorithms" *Nov. 2011.*

[4] R. Bell, Y. Koren, and C. Volinsky, "Modeling relationships at multiple scales to improve accuracy of large recommender systems" KDD '07: Proceedings of the 13th ACM SIGKDD international conference on Knowledge discovery and data mining, New York, NY, USA, 2007, ACM.

[5] O. Celma and P. Herrera, "A new approach to evaluating novel recommendations", RecSys '08: Proceedings of the 2008 ACM conference on Recommender systems, New York, NY, USA, 2008, ACM.

[6] C. N. Ziegler, S. M. McNee, J. A. Konstan, and G. Lausen, "Improving recommendation lists through topic diversification", WWW '05: Proceedings of the 14th international conference on World Wide Web, New York, NY, USA, 2005, ACM.

