[IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY Volume-2, Issue-3, 538 – 545

IMPROVING ACCUMULATED RECOMMENDATION SYSTEM A COMPARATIVE STUDY OF DIVERSITY USING PAGERANK ALGORITHM TECHNIQUE

Dasari Siva Krishna[1], K. Rajani Devi[2]

[1]*M.Tech (IT), Nalanda institute of engineering and technology (NIET), Sattenapalli (M.D.), Guntur(D.T),India,* ***d.sivakrishna6@gmail.com***

[2]*Assoc.professor (HOD), Dept of IT, Nalanda institute of engineering and technology (NIET), Sattenapalli(M.D.), Guntur(D.T),India,* ***rajanidevik@gmail.com***

Abstract

Recommender systems can use data mining techniques for making recommendations using knowledge learnt from the action and attributes of users. In a world where the number of choices can be overwhelming, recommender systems help users find and evaluate items of interest. Previous works introduce and explore a number of item ranking techniques that can generate recommendations that have substantially higher aggregate diversity across all users while maintaining comparable levels of recommendation accuracy. In this paper we implement PageRank algorithm and recommendation tool with rating techniques along user studies exploring users' perceptions and acceptance of the diversity metrics as well as the users' satisfaction with diversity sensitive recommender systems would be an important step in this line of research.

Index terms*: Pagerank algorithm, Item-based collaborative filtering of purchase statistics, User-based Collaborative Filtering***,** *celebrative techniques, recommender system approach's.*

1. INTRODUCTION

Recommender systems or **recommendation systems** are a subclass of information filtering system that seek to predict the 'rating' or 'preference' that a user would give to an item (such as music ,books, e commerce applications).Now a day's recommender system place a vital role to find out the user information germane data and recommender systems technologies have been introduced to help people deal with the vast amounts of information. and they have been widely used in research as well as e-commerce applications, such as the ones used by Amazon and Netflix. In real world settings, recommender systems generally perform the following two tasks in order to provide recommendations to each user. First, the ratings of unrated items are estimated based on the available information (typically using known user ratings and possibly also information about item content or user demographics) using some recommendation algorithm. And second, the system finds items that maximize the user's utility based on the predicted ratings, and recommends them to the user. Typically, a recommender system compares the user's profile to some reference characteristics. These characteristics may be from the information item (the content-based approach) or the user's social environment (the collaborative filtering approach). In the content-based approach, the system must be capable of matching the characteristics of an item against relevant features in the user's profile. In order to do this, it must first construct a sufficiently-detailed model of the user's tastes and preferences through preference elicitation. This may be done either explicitly (by querying the user) or implicitly (by observing the user's behaviour). In the collaborative filtering approach, the recommender system would identify users who share the same preferences (e.g. rating patterns) with the active user, and propose items which the like-minded users favoured (and the active user has not yet seen). Content-based approach is unable to exploit quality judgments of other users. many studies developing new algorithms that can improve the conjecturing accuracy of recommendations. The goals of recommender systems is to provide a user with highly distinctive or personalized items, and more diverse recommendations result in more opportunities for users to get recommended such items. With this motivation, some studies proposed new recommendation methods that can increase the diversity of recommendation sets for a given *individual* user, often measured by an average divarication between all pairs of recommended items, while maintaining an acceptable level of accuracy. These studies measure recommendation diversity from an individual user's perspective(i.e., *individual diversity*). high individual diversity of recommendations does not necessarily imply high aggregate diversity while the benefits of recommender systems that provide higher aggregate diversity would be ostensibly to many users. Recommender systems on aggregate diversity in real-world e-commerce applications has not been well understood. For example, using data from online optical retailer, confirms the "long tail" phenomenon that refers to the increase in the tail of the sales distribution (i.e., the increase in aggregate diversity) attributable to the usage of the recommender system.

As seen from this recent debate, there is a growing awareness of the importance of aggregate diversity in recommender systems. Recent study focus on developing algorithmic techniques for improving aggregate diversity of recommendations which can be instinctively measured by the number of distinct items recommended across all users. Higher diversity (both individual and aggregate) produce the expense of accuracy. As known well, there is a trade-off between accuracy and diversity because high accuracy may often be obtained by safely recommending to users the most popular items, which can clearly lead to the abatement in diversity, i.e., less personalized recommendations .The recent study proposed ranking techniques are extremely *efficient*, because they are based on scalable sorting-based heuristics that make decisions based only on the "local" data (i.e., only on the candidate items of each individual user) with having to keep track of the "global" information(item based user based), such as which items have been recommended across all users and how many times without having user satisfaction and user needs. The techniques are also premeditate, since the user has the control to choose the acceptable level of accuracy for which the diversity will be maximized. Also, ranking techniques provide a *flexible* solution to improving recommendation diversity because: they are applied *after* the unknown item ratings have been estimated and, thus, can achieve diversity gains in conjunction with a number of different rating prognosis techniques.

When it comes to recommendation systems, everybody's looking to increase accuracy: the Netflix Prize was awarded last July for an algorithm that improved the accuracy of the service's recommendation algorithm by 10 percent. However, computer scientists are finding a new metric to improve upon: recommendation diversity. Accuracy has long been the most prized measurement in recommending content, like movies, links, or music. However, computer scientists note that this type of system can narrow the field of interest for each user the more it is used. Improved accuracy can result in a strong filtering based on a user's interests, until the system can only recommend a small subset of all the content it has to offer. To widen the potential field of user interest, the authors developed a hybrid of two algorithms. One combined an algorithm that based its recommendations on random walks between highly connected users and material; the other mirrored the process of heat diffusion, spreading ratings at a decreasing level of potency as the recommendation had to travel further. The heat diffusion algorithm can be thought of as a system that has users connected in a network with the objects they have interacted with and evaluated, and values are passed among the items in this network to develop ratings. While the accuracy of recommendations has been the prized focus (literally) in these systems, diversity and novelty are prized measures too (think of all those friends who boast about liking bands or movies before they were popular). The algorithms are still largely experimental, and the authors note that there is a significantly higher computational cost associated with using a hybrid algorithm. Nonetheless, diversity of suggestions seems to be the next horizon in refining recommendation systems.

In this paper we illustrate item based algorithm and page rank algorithm techniques and system can be recommended based on the top overall sellers on a site, on the demographics of the consumer, or on an analysis of the past buying behaviour of the consumer as a prediction for future buying behaviour. To address these issues we have explored several collaborative filtering techniques such as the item based approach, which identify relationship between items and indirectly compute recommendations for users based on these relationships. The user based approach was also studied; it identifies relationships between users of similar tastes and computes recommendations based on these relationships. In this paper, we introduce the topic of recommender system. It provides ways to evaluate efficiency, scalability and accuracy of recommender system. The paper also analyzes different algorithms of user based and item based techniques for recommendation generation.

2. BACKGROUND

Recommending systems are the bread and butter of large e-commerce web sites. Potential customers are presented with highly relevant product recommendations based on their demonstrated likes; thus increasing the probability of purchase. Customers purchasing one or more items are presented with like-items increasing the likelihood of cross-selling. And returning customers are retained by the quality of recommendations based on their previous purchases. In general, the goal of any recommendation system is to present users with a highly relevant set of items.

ISSN: 2250–3676

[IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY Volume-2, Issue-3, 538 – 545

There are two general types of recommendation algorithms. One is based on the similarity between the currently active user and previous users. The similarity is usually computed using information about the items each user demonstrated interest (or purchased). The other type is based on the similarity of items selected by the current user to other items. Usually described as user-based and item-based collaborative filtering, respectively, these algorithms differ in the data collected and the associations formed between the data items.

Both types of systems can be further classified as either memory- or model-based collaborative filters. The memory-based approach relies on the entire database of observations being in memory when a recommendation is requested. The request is then satisfied by searching for neighbours who exhibit similar behaviour and then combining the preferences of these neighbours into a set of ranked items. The model-based systems rely on building a model offline (usually by means of a machine learning algorithm) and then using the learned model to classify the user and recommend relevant items.

The Collaborative Network Library described in this article is a simplistic item-centric, memory-based, collaborative filtering algorithm. It is not a probabilistic approach and is solely based on observed associations between pairs of items; thus, it does not handle unobserved items or features. This means that if the user selects an item that has not been a selection of any previous users, then the system will not be able to produce a recommendation. For an item to be part of a recommendation, it must have been associated with other items by one or more previous users.

3. ARCHITECHTURE

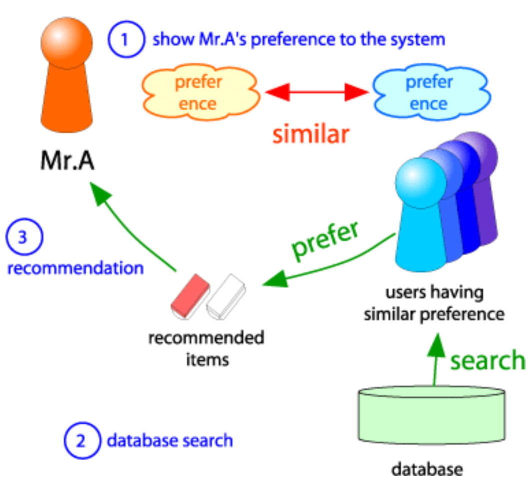


Figure 1: Architecture Diagram

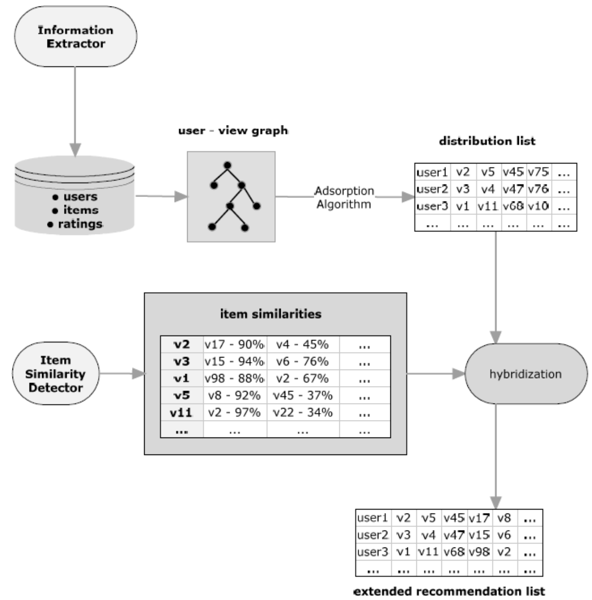


Figure 2:

4. RELATED WORK

4.1. Item-based collaborative filtering of purchase statistics

We are not always given ratings: when the users provide only binary data (the item was purchased or not), then Slope One and other rating-based algorithm do not apply. Examples of binary item-based collaborative filtering include Amazon's item-to-item patented algorithm which computes the cosine between binary vectors representing the purchases in a user-item matrix.

Being arguably simpler than even Slope One, the Item-to-Item algorithm offers an interesting point of reference. Let us consider an example.

Sample purchase statistics

Customer	Item 1	Item 2	Item 3
John	Bought it	Didn't buy it	Bought it
Mark	Didn't buy it	Bought it	Bought it
Lucy	Didn't buy it	Bought it	Didn't buy it

In this case, the cosine between items 1 and 2 is:

$$\frac{(1,0,0)\cdot(0,1,1)}{\|(1,0,0)\|\|(0,1,1)\|}=0,$$

The cosine between items 1 and 3 is:

$$\frac{(1,0,0)\cdot(1,1,0)}{\|(1,0,0)\|\|(1,1,0)\|}=\frac{1}{\sqrt{2}},$$

Whereas the cosine between items 2 and 3 is:

$$\frac{(0,1,1)\cdot(1,1,0)}{\|(0,1,1)\|\|(1,1,0)\|}=\frac{1}{2}.$$

Hence, a user visiting item 1 would receive item 3 as a recommendation, a user visiting item 2 would receive item 3 as a recommendation, and finally, a user visiting item 3 would receive item 1 (and then item 2) as a recommendation. The model uses a single parameter per pair of item (the cosine) to make the recommendation. Hence, if there are n items, up to $n(n-1)/2$ cosines need to be computed and stored. Systems Pearson correlation only computes the similarity between the two users who rate a same item.

Considering as the most used algorithm in Collaborating Filtering, there are some limitations in user-based approach. The first limitation is the scalability of the algorithm. The computation of user-base CF is more complex when the number of users gets bigger .Therefore, it is difficult to use user-based CF in big online service companies as Amazon and Netflix. User-based CF recommender systems can work very well with a small dataset, but they usually don't work well with a large dataset like Netflix's dataset. Second limitation of user based CF is performance. Its performance is slow because User-based CF needs to recomputed the similarity of user-user every time it gives new recommendation.

4.2. User-based Collaborative Filtering

User-based Collaborative Filtering is one of the most chosen algorithms to use in recommender systems by online companies . It relies on the similarly behaviours between each users in the group. These behaviours are including buying or ratings items. The behaviours of various users in one group can help recommending other users in same group to buy or rate different items. There are many algorithms to calculate the similarity between the two users in CF systems. One of them is Pearson correlation algorithm. It is a most chosen algorithm to use in CF systems . Pearson correlation only computes the similarity between the two users who rate a same item.

5. RECOMMENDER SYSTEM-MODULES

5.1 Posting the opinion

In this module, we get the opinions from various people about business, e-commerce and products through online. The opinions may be of two types. Direct opinion and comparative opinion. Direct opinion is to post a comment about the components and attributes of products directly. Comparative opinion is to post a comment based on comparison of two or more products. The comments may be positive or negative.

5.2 Recommendation Technique:

However, the quality of recommendations can be evaluated along a number of dimensions, and relying on the accuracy of recommendations alone may not be enough to find the most relevant items for each.

User, these studies argue that one of the goals of recommender systems is to provide a user with highly personalized items, and more diverse recommendations result in more opportunities for users to get recommended such items. With this motivation, some studies proposed new recommendation methods that can increase the diversity of recommendation sets for a given *individual* user. They can give the feedback of such items.

5.3 Rating Prediction:

First, the ratings of unrated items are estimated based on the available information (typically using known user ratings and possibly also information about item content) using some recommendation algorithm. Heuristic techniques typically calculate recommendations based directly on the previous user activities (e.g., transactional data or rating values). For each user, ranks all the predicted items according to the predicted rating value ranking the candidate (highly predicted) items based on their predicted rating value, from lowest to highest (as a result choosing less popular items.

5.4 Ranking Approach:

Ranking items according to the rating variance of neighbours of a particular user for a particular item. There exist a number of different ranking approaches that can improve recommendation diversity by recommending items other than the ones with topmost predicted rating values to a user. A comprehensive set of experiments was performed using every rating prediction technique in conjunction with every recommendation ranking function on every dataset for different number of top-*N* recommendations.

6. Recommendation ALGORITHUM

6.1. Collaborative Filtering

One approach to the design of recommender systems that has seen wide use is collaborative filtering. Collaborative filtering methods are based on collecting and analyzing a large amount of information on users' behaviours, activities or preferences and predicting what users will like based on their similarity to other users. User-based collaborative filtering attempts to model the social process of asking a friend for a recommendation. A key advantage of the collaborative filtering approach is that it does not rely on machine analyzable content and therefore it is capable of accurately recommending complex items such as movies without requiring an "understanding" of the item itself.

One of the most famous examples of Collaborative Filtering is item-to-item collaborative filtering (people who buy x also buy y), an algorithm popularized by Amazon.com's recommender system. Other examples include:

- As previously detailed, Last.fm recommends music based on a comparison of the listening habits of similar users.
- Facebook, MySpace, LinkedIn, and other social networks use collaborative filtering to recommend new friends, groups, and other social connections (by examining the network of connections between a user and their friends).

6.2. Content-Based Filtering

Another common approach when designing recommender systems is content-based filtering. Content-based filtering methods are based on information about and characteristics of the items that are going to be recommended. In other words, 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 (i.e., a set of discrete attributes and features) 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. Simple approaches use the average values of the rated item vector while other sophisticated methods use machine learning techniques such as Bayesian Classifiers, cluster analysis, decision trees, and artificial neural networks in order to estimate the probability that the user is going to like the item.

6.3 Hybrid Recommender Systems

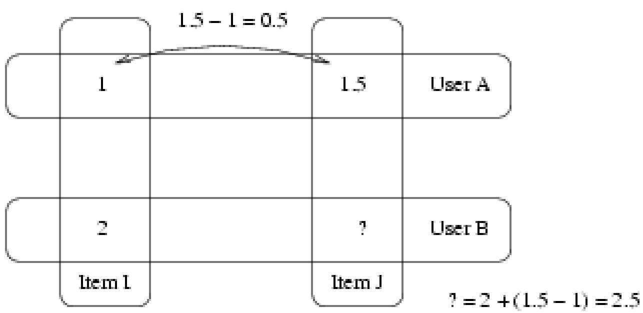
Recent research has demonstrated that a hybrid approach, combining collaborative filtering and content-based filtering could be more effective in some cases. Hybrid approaches can be implemented in several ways: by making content-based and collaborative-based predictions separately and then combining them; by adding content-based capabilities to a collaborative-based approach (and vice versa); or by unifying the approaches into one model . Several studies empirically compare the performance of the hybrid with the pure collaborative and content-based methods and demonstrate that the hybrid methods can provide more accurate recommendations than pure approaches. These methods can also be used to overcome some of the common problems in recommender systems such as cold start and the sparsity problem.

Netflix is a good example of a hybrid system, as it makes recommendations both by comparing the watching habits of similar users (i.e. collaborative filtering) as well as by offering movies that share characteristics with films that a user has rated highly (content-based filtering).

6.3. Item-based Rating-Based collaborative filtering
To drastically reduce overfitting, improve performance and ease implementation, the Item-based Rating-Based collaborative filtering algorithms was proposed.
We proposed new recommendation tool base on these combined three algorithm.

7. THE IMPLEMENTATION OF PAGERANK IN THERECOMMENDAION RANKING APPROACHES

Essentially, instead of using linear regression from one item's ratings to another item's ratings ($f(x) = ax + b$), it uses a simpler form of regression with a single free parameter ($f(x) = x + b$). The free parameter is then simply the average difference between the two items' ratings. It was shown to be much more accurate than linear regression in some instances[1], and it takes half the storage or less.



Example:1

User A gave a 1 to Item I and an 1.5 to Item J.

User B gave a 2 to Item I.

How do you think User B rated Item J?

The Slope One answer is to say 2.5 (1.5-1+2=2.5).

For a more realistic example, consider the following table.

Sample rating database

Customer	Item 1	Item 2	Item 3
John	5	3	2
Mark	3	4	Didn't rate it
Lucy	Didn't rate it	2	5

In this case, the average difference in ratings between item 2 and 1 is (2+(-1))/2=0.5. Hence, on average, item 1 is rated above item 2 by 0.5. Similarly, the average difference between item 3 and 1 is 3. Hence, if we attempt to predict the rating of Lucy for item 1 using her rating for item 2, we get 2+0.5 = 2.5. Similarly, if we try to predict her rating for item 1 using her rating of item 3, we get 5+3=8.

If a user rated several items, the predictions are simply combined using a weighted average where a good choice for the weight is the number of users having rated both items. In the above example, we would predict the following rating for Lucy on item 1:

$$\frac{2 \times 2.5 + 1 \times 8}{2 + 1} = \frac{13}{3} = 4.33$$

Hence, given n items, to implement Slope One, all that is needed is to compute and store the average differences and the number of common ratings for each of the n^2 pairs of items.

8.COMPARATIVE STUDY OF DIVERSITY:RECOMMENDING ITEMS

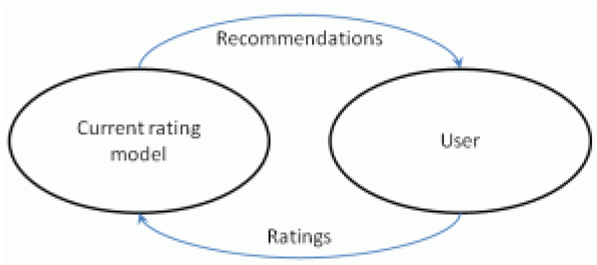
Ranking approaches can obtain significant diversity gains with a small amount of accuracy loss. In this paper we illustrate another interesting topic for ranking approach is to explore possibilities to improve *both* accuracy as well as diversity using item based collaborative filtering techniques and PageRank

Algorithm. Based on the findings described in this paper, a possible approach to improving both the accuracy and diversity of the standard technique would be to turn over new leaf the proposed recommendation re-ranking techniques, which are already known to produce diversity gains, in a way that increases their accuracy.

Imagine a recommender system that advises, for instance, movies to users. How and what you recommend to users, affects what movies they will watch, and how users behave when presented with recommendations affects what the system will recommend in the future. In other words, depending on what is recommended and how users behave, the dynamics of the whole system can become very different. What we investigate is the *impact* of these behaviours (both from the system and user side), and we are particularly interested in how *diversity* changes as time passes.

ISSN: 2250–3676

[IJESAT] INTERNATIONAL JOURNAL OF ENGINEERING SCIENCE & ADVANCED TECHNOLOGY Volume-2, Issue-3, 538 – 545



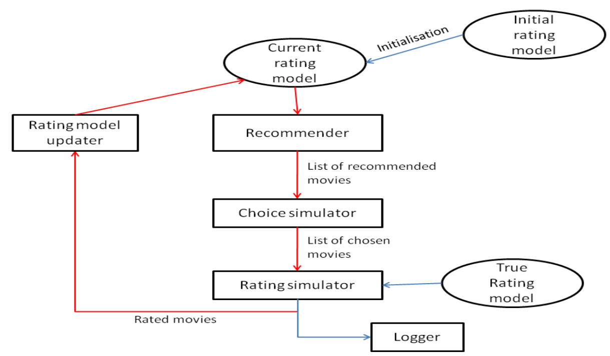
- What happens if we force users to rate a certain number of items in a period of time (e.g., everyone rates 5 movies a week)? Such a restriction is an example of how the owner of the recommender can ‘influence’ the behaviour of users.
- What is the effect of changing the type of information that a recommender gives to users? For example, a recommender can show to the user either most popular movies, or movies that match best the user’s preferences.

We considered diversity defined in various ways. We investigate, for example, entropy of rated movies, overall variance of ratings, variance of ratings per user, etc. These diversity values were measured using simulations that were based on the Netflix dataset (hence the use of the word ‘movies’ as opposed to the more general term ‘items’).

The main idea behind the simulation was that the circular process illustrated above can be broken down into ’rounds’. In one round, which we considered to be a month, a number of users rate a number of movies, and at the end of the month, the recommender is retrained. How users react to recommendations was simulated using several choice models. For instance, in one simulation, users selected and rated those and only those movies that the system recommended to them (we called them ‘Yes-men’). In another simulation, users accepted everything the system offered, but everyone had to rate the same number of movies (‘Uniform Yes-men’). At the other end of the scale were simulations with ‘Randomisers’, where users seemed to select and rate movies randomly.

All simulations were tied to the Netflix dataset. For instance, the notion of a month came from the data, the list of available movies per month as well, and ‘true ratings’ were calculated using the same dataset. Simulation outline (red arrows demonstrate a single round of simulations)

As for the results, we have found that forced uniformity in terms of number of items rated does not necessarily result in users becoming more uniform, and the mean ratings they give to items will decrease, indicating lower system performance as perceived by the user. Also, we have identified how three kinds of choice models, i.e., Yes-men, Trend-followers (always watch currently highest rated movies) and Randomisers, result in different diversity and mean rating values. This is a particularly important result as these behaviours can directly be encouraged by recommender system owners, e.g., we might decide to offer more trending items to one particular kind of users, while other users might need different kind of recommendations.



10. FUTURE WORK

Possible future works might be: (1) Include the rating dates in the classifier to accommodate the change of user’s taste over time; [2] to remove several global effects including user rating average, movie rating average, rating dates, release year of the movie and etc.

11. CONCLUSION

In this paper, we designed a PageRank algorithm and Item based collaborative filtering technique to provide a quality recommendations to user, with user satisfaction, user needs and number of recommendation ranking techniques that can provide significant improvements in recommendation diversity. Recommender systems benefit users by enabling them to find items they like. Conversely, they help the business by generating more sales. It provides ways to evaluate efficiency, scalability and accuracy of recommender system.

REFERENCES

1. P. Resnick, N. Iakovou, M. Sushak, P. Bergstrom, and J. Riedl, "GroupLens: An Open Architecture for Collaborative Filtering of Netnews," *Proc. 1994 Computer Supported Cooperative Work Conf*., 1994.

2. B. M. Sarwar, G. Karypis, J. Konstan, and J. Riedl, "Analysis of Recommender Algorithms for E-Commerce," *ACM E-Commerce 2000 Conf.*, pp.158-167, 2000.

3. M. Zhang and N. Hurley, "Avoiding monotony: improving the diversity of recommendation lists," *Proc. of the 2008 ACM Conf. on Recommender systems*, pp. 123-130, 2008.

4. K. Greene, "The $1 million Netflix challenge," Technology Review. www.tech nologyreview.com/read_article.aspx?id=17587&ch=biztech, October 6, 2006.

5. D. Billsus and M. Pazzani, "Learning Collaborative Information Filters," *Proc. Int'l Conf. Machine Learning*, 1998.

6. S. Breese, D. Heckerman, and C. Kadie, "Empirical Analysis of Predictive Algorithms for Collaborative Filtering," *Proc. of the 14th Conf. on Uncertainty in Artificial Intelligence*, 1998.

7. M. Balabanovic and Y. Shoham, "Fab: Content-Based, Collaborative Recommendation," Comm. ACM, 40(3), pp. 66-72, 1997.

8. G. Linden, B. Smith, and J. York. Amazon. com Recommendations: Item-to-Item Collaborative Filtering. 2003.

9. Netflix. Netflix prize. http://www.netflixprize.com/.

10. L.J. Herlocker, A.J. Konstan, and J. Riedl. Explaining collaborative filtering recommendations. In Proceedings of the 2000 ACM conference on Computer supported cooperative work table of contents Philadelphia, Pennsylvania, United States, pages 241–250, 2000.

11. Google Video. http://video.google.com/.

12. Yahoo! Video. http://video.yahoo.com/.

13. Improving Aggregate Recommendation Diversity Using Ranking-Based Techniques-2011 Gediminas Adomavicius, *Member, IEEE*, and YoungOk Kwon

14. Youtube. http://www.youtube.com/.

15. Collaborative Filtering for the Netflix Prize Hao Zhang Partner: Sahand Negahban zhanghao@eecs.berkeley.edu Department of EECS, University of California, Berkeley

BIOGRAPHIES

DASARI SIVA KRISHNA, M.Tech (IT), NALANDA INSTITUTE OF ENGINEERING AND TECHNOLOGY (NIET), SATTENAPALLI(M.D.), GUNTUR (D.T), ANDHRA PRADESH, INDIA

K.RAJANI DEVI Assoc.Professor, H.O.D., Department of IT, N.I.E.T. SATTENPALLI (M.D), GUNTUR(D.T), ANDHRA PRADESH,INDIA