Metadata Based Recommender Systems

Paritosh Mittal
IIIT-Delhi
Email: paritosh10059@iiitd.ac.in

Aishwarya Jain
IIIT-Delhi
Email: aishwarya10007@iiitd.ac.in

Angshul Majumdar
IIIT-Delhi
Email: angshul@iiitd.ac.in

***Abstract*—For building a recommendation system the eCommerce portal gathers the user's ratings on various items in order to determine his/her choice regarding its merchandise. The portal also collects metadata for the user when he/she signs up and becomes a part of the system; therefore the portal has access to information such as user's age, gender, occupation, location, etc. Till date almost all prior studies used the metadata for alleviating the cold-start problem; this information was not used for improving the recommendations. For the first time in this work, we propose a simple neighborhood selection technique by giving importance to the metadata groups for improving the recommendations.**

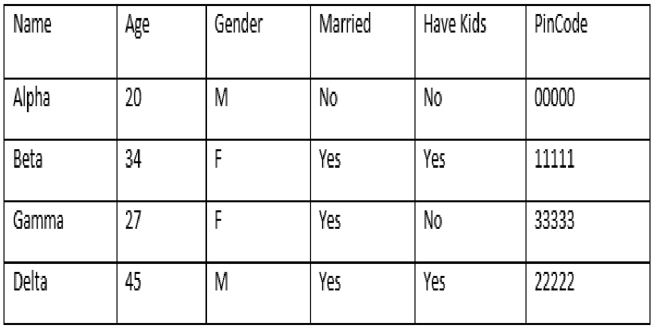
I. Introduction

In this era of e-commerce trade, recommender systems have been used tremendously to provide users with recommendations [17]. The recommendations can vary from information, articles, movies, jokes to services. These systems are customized such that they are as close to the preferences and liking of the user. Their aim is to recommend items that are more likely to be relevant to the user to improve the revenue of the company. Recommender systems use historical data that is given by the user over a period of time and tries to give a short and relevant summary of items from a huge number of unseen item lists.

Various types of systems have been made to recommend items. Each system uses different and specific types of information to recommend lists of items, which is as close as it can be to the user's preferences [6]. Each system will have its own pros and cons. Therefore, the choice of the system should be based on the demand of the user of the system.

Recommender systems can be broadly classified into two categories: content based filtering and collaborative filtering. Content based filtering [4] is done on the basis of qualities of items that are rated/ bought by the user. Under this system the user will rate equivalently, the objects which have similar qualities. In collaborative filtering[5], grouping is done on the historical data rated/bought by the user. Based on this historical data, it tries to find users which are similar to the user and a recommendation list is generated based on the neighborhood users. Some hybrid recommender systems[3] are also used to obtain better results.

All the above mentioned methods suffer from a cold-start problem irrespective of the algorithm [7] [8]. In this scenario, whenever a new user is introduced in the system, it has no prior item rating for this user. So recommendations using historical data become meaningless and an alternative is required that does not depend on historical ratings. To counter this problem, meta data [9] [10] such as age, gender, occupation, and ethnicity are used to initially cluster similar users in a group. An example of meta data is shown in Fig. 1. Recommendations of items are made using the item ratings provided by neighbors from the group. Fig. 2 shows how meta data is used whenever a new user in introduced in the system. Eventually, the users rate enough items so that similarity based techniques can be employed and metadata is not required for recommendations.



Name	Age	Gender	Married	Have Kids	PinCode
Alpha	20	M	No	No	00000
Beta	34	F	Yes	Yes	11111
Gamma	27	F	Yes	No	33333
Delta	45	M	Yes	Yes	22222

Fig. 1: An example of meta data

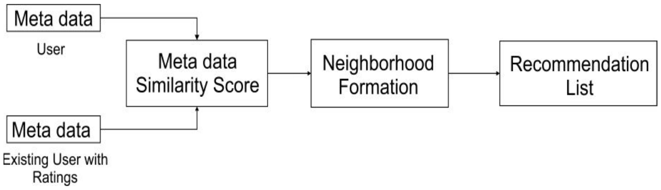


Fig. 2: Meta data usage to solve cold start

In this framework, we propose that using the metadata should not be limited to cold start-problems but further should be used in later stages as well. Meta data based filtering requires storing and usage of person's meta data such as gender, age, location etc. It assumes that people with same demographic attributes are likely to rate the items similarly. We redefine the neighborhood list based on the meta data so that a more refined and accurate recommendation list can be obtained. People with same meta data tend to have similar thinking. People in an area might be sports oriented hence lot of people might like sports movies. Students are young and might like more of action movies. Retired people might not be that interested in action movies as they want to watch light movies in general.

978-1-4799-3080-7/14/$31.00 ©2014 IEEE

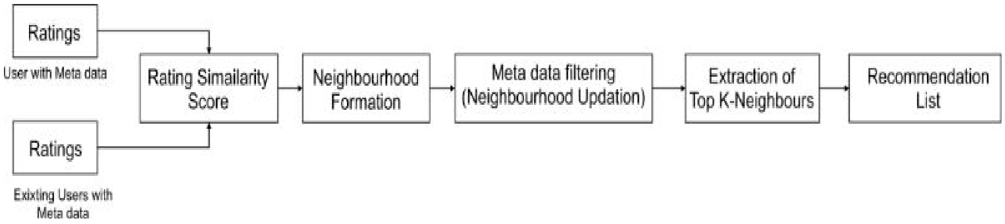

Ratings
User with Meta data
Ratings
Exixting Users with Meta data
Rating Simailarity Score
Neighbourhood Formation
Meta data filtering (Neighbourhood Updation)
Extraction of Top K-Neighbours
Recommendation List

Fig. 3: Proposed Framework

Ratings & meta data	M1	M2	M3	M4	M5	M6	M7	M8	Age	Gender	Maritual status	Have Kids
A	5	4	5		3			4	20	M	No	No
B		1	5	3	5			1	34	M	Yes	Yes
C	1		1		4		4	4	27	F	Yes	No
D	5		5	1	5	5	2		45	M	Yes	Yes

TABLE I: Users and Movie Ratings

II. Brief Review of Recommender Systems

Recommender systems want to predict items which have not yet been rated by them. Two approaches that are used to make a recommender systems are the Neighborhood based methods [11] and Latent Factor Models [16]. In the latent factor models it assumes that ratings are deeply influenced by set of parameters that are specific to domain in which the recommender system is made for. For example: director, genre, actor of the movie. In practice these parameters are hidden and figuring them out is difficult. Even if one is able to think about some of these hidden parameters, estimating the impact of these parameters is very difficult task. It is usually done using the mathematical models. Some of the approaches are decomposition of matrix into user feature and item feature matrix. It automatically rates the features for their impact but understanding them might them is still a difficult task. Another factor to take care of is that these matrices are sparse and have missing data which can not be assumed to be null so methods such as SVD and Lanczo algorithms can be used. The decomposition focuses on ratings which are known and tries to minimize the reconstruction error.

In the Neighborhood based models [13] [14] [15] the relations between users is based on their historical ratings of the products, which identifies users-product associations. It does not require extensive data collection in comparison to latent methods. These methods require no domain knowledge as well. It can be seen as missing value estimate. We first find the user-item matrix of scores which measures the interest between respective users and items. It is done by finding the similarity measure between different users based on the items they have already been rated. Different measures such as Pearson, spearman, cosine etc can be employed for this. An example of pearson formula is given below. Now we pick a user for whom recommendations have to be generated. Based on the similarity scores the users with the highest scores are like-minded with the given users which form the neighborhood. K nearest neighbors is employed for the same. The items which are rated by the neighborhood and not by users are given as recommendation to the users. The merits of using such an algorithm are intuitive, requires no training and the relationship can be easily explained. Estimating the neighborhood can vary depending on the parameters. Various weighting techniques such as variance weighting and weighted sum are used to associate weights with neighbors.

$$pearson(\mathbf{u},\mathbf{v}) = \frac{\sum_{Iu\cap Iv}(\mathbf{r_{u,i}}-\mathbf{\bar{r}_u})(\mathbf{r_{v,i}}-\mathbf{\bar{r}_v})}{\sqrt{\sum_{Iu\cap Iv}(\mathbf{r_{u,i}}-\mathbf{\bar{r}_u})^2}\sqrt{\sum_{Iu\cap Iv}(\mathbf{r_{v,i}}-\mathbf{\bar{r}_v})^2}} \tag{1}$$

III. Existing meta data Recommender systems

Various approaches which use meta data to get better recommendation list are there which give some weight-age to metadata. A hybrid work in recommender system [1] in which the K nearest neighbors algorithm is modified by adding another vector of length 21. Hence saving all the meta data information (in this case age, gender and occupation) then the both the meta data and score similarity is calculated combined as a single feature vector. New K nearest neighbors are chosen based on the new updated score.

Another research in this hybrid system of meta data and collaborative filtering which have proposed of using classification of domain of movie. This classification is based on the meta data of users available to them. The meta data that they have used are student, marital status, age, and gender [2].

In this paper we propose using the meta data not just to solve the cold start problem but also in running the recommender system in subsequent stages as well. Our framework uses both historical and meta data in 2 stage process to get better and more accurate results instead of using them together.

IV. Proposed Framework

Fig. 3 illustrates the steps involved in the proposed framework. Our framework is a hybrid recommender system of meta data and historical ratings. It can also be seen as a modification of K nearest neighbors where the neighbors are using both the information available to us. The idea behind the algorithm is to use the meta data along with the historical ratings to find the correct neighborhood. The algorithm is divided into 4 major parts which are explained in the following subsection:

1) *Historical Rating based neighborhood estimation*: Since the user is already in the system he has already rated various items. The system calculates the similarity between this user and all other users in the system using any similarity measure. Then based on the similarity score between the user, all other users are sorted such that the user with the most similar score is at the top while the user with least score is at the bottom. Cosine similarity measure is utilized:

$$\cos(\mathbf{u1}, \mathbf{u2}) = \frac{\mathbf{u1u2}}{\|\mathbf{u1}\|\|\mathbf{u2}\|} = \frac{\sum_{i=1}^{n} \mathbf{u1}_i \mathbf{u2}_i}{\sqrt{\sum_{i=1}^{n} (\mathbf{u1}_i)^2}\sqrt{\sum_{i=1}^{n} (\mathbf{u2}_i)^2}} \tag{2}$$

2) *Meta data filtering*: The similarity score of the user with all other other users in obtained in the previous step. We now use meta data such as gender, age, occupation, ethnicity, and demography to further update this list. In the framework we propose that users with same meta data should be added to the neighborhood list while the users with different meta data should be removed from the neighborhood list. Each system has variety of information in the metadata. Choosing the proper metadata is an essential part. The metadata chosen is such that it covers a large amount of users in the system such that lot the representation of that metadata is sufficient but care should be taken that metadata which covers entire population should also be not chosen. This way we are updating the list with only with users with the same meta data.

$$u_{neighbor} = neig \quad where\ u_{metada} = n_{metada} \tag{3}$$

3) *Picking Top K-Neighbors*: From the updated list that was obtained in the previous step, top K-Neighbors are chosen based on the similarity score from step-1. They form the neighborhood for the given user. The nearest neighbor parameter K is set. Top K people in the same occupation based on similarity have to be considered. If the threshold is not reached then we take the maximum available users from that metadata and not add any other user from different classes of the given metadata.

4) *Recommendation List*: Now the neighborhood is obtained for each user in the previous step. Each user is given equal weightage when calculating the rating of items which are not rated by the user. Now the top rated items are shown to the given user as shown in equation 4. Here k are the neighbors and j is the item. w_k is weight of each neighbor which is set to $\frac{1}{K}$

$$u_{i,j} = \sum_{k=1}^{K} w_k \times n_{k,j} \tag{4}$$

Movie	A	B	C	D
A	1.0000	0.6505	0.2933	0.7325
B	0.6505	1.0000	0.6505	0.6622
C	0.2933	0.6505	1.0000	0.4846
D	0.7325	0.6622	0.4846	1.0000

TABLE II: Similarity Score between all users (Cosine)

A hypothetical example of the framework is shown through Tables I and II. We have 4 users and 8 movies rated by them. We have the user as A and we have to find the ratings for items not yet rated by him/her. We also have the meta data age, gender, marital status and have kids for each of them. Table I gives the information of meta data and movie ratings available for each user. Then similarity score is calculated between them using the available movie ratings as given in Table II. On the basis of similarly we sort them in descending order. For user A the sorted order will be B,D,C. Now we use the meta data gender as filtering so user C is removed as the gender is female while the gender of user A is male. So we are left only with user B and D in its neighborhood. So the unrated items are calculated using both B and D.

V. Experiment Evaluation

The given hybrid structure is evaluated on the publicly available Movie Lens data set provided by the GroupLens movie systems. [12]. It has 3 different sets of size 100k, 1M and 10M. Our experiment was conducted on 100k and 1M as both these datasets provide meta data while 10M does not provide any meta data. The meta data consist of age, gender, occupation, and zip code. Table III gives detailed information for the datasets used. We used 80% of total data available to us as training while testing was done on the remaining 20%. The partitioning is performed three times for cross validation and average accuracy is reported. The details of each dataset are mentioned in Table III. Ratings are given out of a maximum of 5 stars where only whole numbers exist. Two experiments are conducted for each dataset. In each experiment we compare with an existing algorithm, baseline, using occupation(top 12), occupation and gender as meta data. Occupation(12) means that only occupations those frequency are in top 12 (others also not there) while occupation has all 21 possible occupations.

$$MAE = \frac{1}{n}\sum_{t=1}^{n} |f_t - p_t| = \frac{1}{n}\sum_{t=1}^{n} |e_t|$$

In this experiment, occupation and gender are chosen as meta data. We chose different metadata to show that choosing the metadata is highly important. Using metadata which are not good representation can yield significantly less results. Both the datasets have 21 different types of occupation. We have compared our algorithm with an existing algorithm which shows improved performance over the existing algorithm and also with the baseline. Accuracy measurement MAE (Mean absolute error) is used to find the accuracy of predicted ratings which is compared to user's actual ratings which we are assuming unknown to us.

All experiments are conducted in MATLAB enviroment on a system of 64 bit 6 GB RAM and having a processor of 2.53 Ghz.

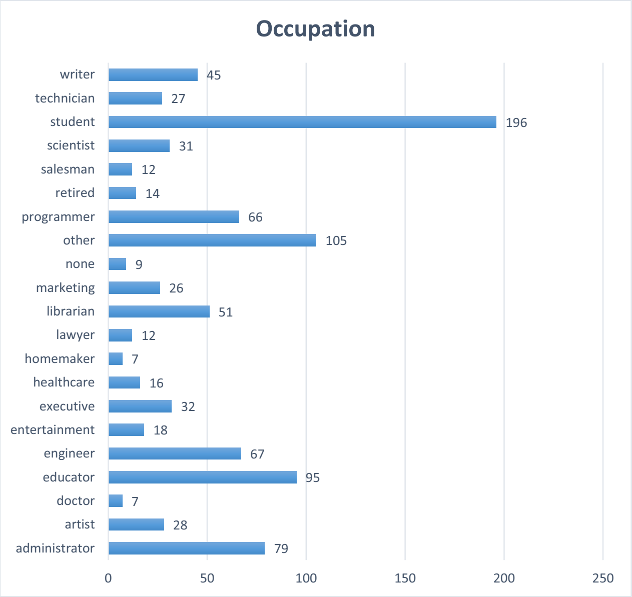


Fig. 4: Occupation for 100k Dataset

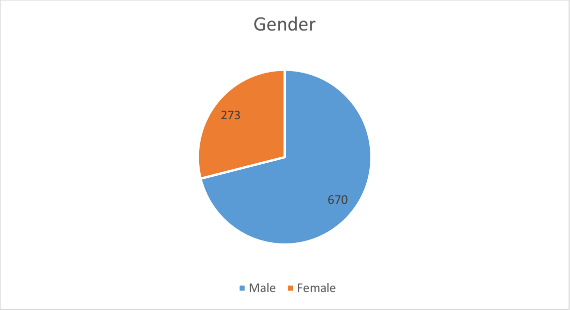


Fig. 5: Gender for 100k Dataset

A. 100 K dataset

The distribution of number of people in each occupation is given in Fig. 4 . The distribution of gender is presented in Fig. 5. The results of this experiment are summarized in Table IV. The nearest neighbor parameter K is set to 45. The coverage of the algorithm in 100 K dataset is 73%. Some of the key observations are given as follows:

Dataset	Number of Users	Number of Movies	Number of Movies Rated	% Ratings Known
100K	943	1682	100,000	93.6953
1M	6040	3952	1,000,209	95.5316

TABLE III: Details of the dataset

Base	Existing [1]	**Gender**			Occup	Occup(12)
		M	F	Comb		
0.79	0.83	0.78	0.88	0.81	0.79	0.74

TABLE IV: Results (MAE) of algorithms on 100k

- MAE is least when we have the occupation (top 12) followed by occupation. Occupation(12) is performing better than occupation as in occupation(12), instances with low frequency are removed. As the count is low, finding sufficient amount of people even with occupation is difficult. For lot of occupations, even the threshold K is not reached.
- In gender, males are performing better than females due to difference in quantities. As shown in Fig. 5 number of males is higher than number of females but overall still less than baseline. Since we first sort all of them based on ratings then take top K neighbors that are considered for males. It does not impact very significantly in females due to the limited quantity of data samples.
- Both the proposed algorithms are better than the existing methods which utilize meta data filtering. This is due to the fact that chances are higher that people with same meta data have more inclination towards common things. The retired do not like pure crime movies as they might not want to think about it.
- The results show that occupation does better than all other algorithms as it has large division. Due to this large division chances become very high that people with same mentality are in same group while gender is a highly generic attribute and fails to provide reliable segregation. This shows that choosing the proper metadata is an essential component. The metadata should cover sufficient large population from the system but too large. The results shows that when using occupation gives better results in comparison to gender.

B. 1M dataset

The distribution of number of people in each occupation is given in Fig. 6. The distribution of gender is given in 7. The results of this experiment are shown in Table V. The nearest neighbor parameter K is set to 250. The coverage of the algorithm in 1M dataset is 82%. Some of the key observations are as follows :

- One can observe that by using the proposed scheme,

Base	Existing [1]	**Gender**			Occup	Occup(12)
		M	F	Comb		
0.78	0.82	0.81	0.90	0.82	0.77	0.75

TABLE V: Results (MAE) of algorithms on 1M

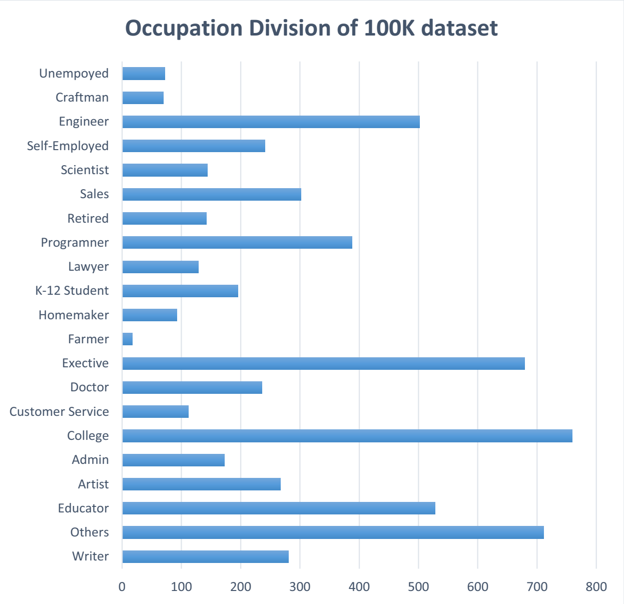


Fig. 6: Occupation for 1M Dataset

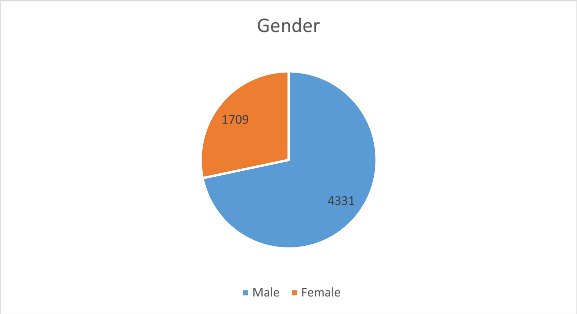


Fig. 7: Gender for 1M Dataset

prediction accuracy is improved by 1 to 1.5% on the average. To put the results in perspective one needs to be aware of the famous Netflix competition- where 1 million dollar award was given to a group which could decrease the prediction error by 8.43%.

- K is set as 250 when choosing the nearest neibhours. Since both the datasets have same division in terms of reasoning. Number of males being higher than females and same division in occupation. The same trend of results can be seen for 1M dataset as well where males outperform females giving less MAE.
- Just like in case of 100k Dataset occupation (12) is doing better among all algorithms giving a minimum MAE of 0.75 .

VI. Conclusion and Future Work

In this paper we presented a novel algorithm using historical ratings along with meta data filtering in recommender systems to recommend items to users which are existing in the system. The proposed framework was tested on two sets of movie lens dataset by taking two different meta data viz occupation and gender. Substantial difference in MAE shows that metadata which properly covers the user database should be chosen. MAE shows that the proposed framework performs better than baseline. This shows that using meta data along with historical data gives a better recommender system.

For future work we would like to test our framework on datasets having more meta data. Correlation between different attributes of a meta data can be used to exploit to further increase the accuracies. This framework can also be applied and tested on different domains for robustness.

References

[1] M. Vozalis and K. G. Margaritis, Collaborative filtering enhanced by demographic correlation, in Proc. AIAI Symposium on Professional Practice in AI, of the 18th World Computer Congress, 2004.

[2] D. Almazro, G. Shahatah, L. Albdulkarim, M. Kherees, R. Martinez, and W. Nzoukou. A Survey Paper on Recommender Systems, arXiv preprint arXiv:1006.5278, Dec. 2010.

[3] Robin Burke. Hybrid recommender systems: Survey and experiments. User Modeling and User-Adapted Interaction, 12:331370, 2002.

[4] M. Balabanovic and Y. Shoham. Fab: Content-based, collaborative recommen- dation. Communications of the ACM, 40, 1997.

[5] Bardul M. Sarwar, George Karypis, Joseph A. Konstan, and John T. Riedl. Item-based collaborative filtering recommendation algorithms. In 10th Interna- tional World Wide Web Conference (WWW10), Hong Kong, 2001.

[6] John S. Breese, David Heckerman, and Carl Kadie. Empirical analysis of pre- dictive algorithms for collaborative filtering. In Fourteenth Conference on Un- certainty in Artificial Intelligence, Madison, WI, 1998.

[7] Schein, Andrew I., et al. "Methods and metrics for cold-start recommendations. " Proceedings of the 25th annual international ACM SIGIR conference on Research and development in information retrieval. ACM, 2002.

[8] Zhang, Zi-Ke, et al. "Solving the cold-start problem in recommender systems with social tags." EPL (Europhysics Letters) 92.2 (2010): 28002.

[9] Ahn, Hyung Jun. "A new similarity measure for collaborative filtering to alleviate the new user cold-starting problem." Information Sciences 178.1 (2008): 37-51.

[10] Drachsler, Hendrik, Hans GK Hummel, and Rob Koper. "Personal recommender systems for learners in lifelong learning networks: the requirements, techniques and model." International Journal of Learning Technology 3.4 (2008): 404-423.

[11] D. Goldberg, D. Nichols, B. M. Oki and D. Terry, Using Collaborative Filtering to Weave an Information Tapestry", Communications of the ACM 35 (1992), 6170.

[12] MovieLens. available at http://www.grouplens.org/node/73, 2006.

[13] J. L. Herlocker, J. A. Konstan, A. Borchers and John Riedl, An Algorithmic Framework for Performing Collaborative Filtering", Proc. 22nd ACM SIGIR Conference on Information Retrieval, pp. 230237, 1999.

[14] J. Konstan, B. Miller, D. Maltz, J. Herlocker, L. Gordon and J. Riedl, GroupLens: Applying Collaborative Filtering to Usenet News", Communications of the ACM 40 (1997), 7787, www.grouplens.org.

[15] G. Linden, B. Smith and J. York, Amazon.com Recommendations: Item-to-item Collaborative Filtering", IEEE Internet Computing 7 (2003), 7680.

[16] R. Bell, Y. Koren, and C. Volinsky. Modeling relationships at multiple scales to improve accuracy of large recommender systems. Procs of the 13th ACM SIGKDD, 2007.

[17] A. Ansari, S. Essegaier, and R. Kohli. Internet Recommendation Systems. J. of Marketing Research, 37(3), 2000

[18] Tsunoda, Tomohiro, and Masaaki Hoshino. "Automatic metadata expansion and indirect collaborative filtering for TV program recommendation system." Multimedia Tools and applications 36.1-2 (2008): 37-54.