IJournals: International Journal of Social Relevance & Concern (IJSRC)
ISSN-2347-9698
Volume 3 Issue 11, November 2015

A Refined Survey on Recommenders 2

Lebi Jean Marc Dali and Qin ZhiGuang

Abstract—Recommender systems are used to help in decision making. They suggest items that are of interest to a particular user, thus they make predictions about potential items a user may like. Hence recommenders are very important for online businesses and individuals. There exists many recommenders on the market and each year new recommenders are created. It is almost impossible for a common user to keep track of all these recommenders and know which one is required for his particular application. Hence in this paper, we present a general survey on important recommenders. The recommenders are chosen here vary widely in the problems they solve. Therefore all types of users irrespective of their peculiarities may find their share in this work. Also this paper is written in a simple language so that the common users (with no recommender background) may understand not feel alienated.

***Index Terms*— *Recommenders,Collaborative Filtering,model-based RS, Trust based algorithm, social network*.**

1. Introduction

Before taking a major decision in our lives (like in purchasing a new car), we rely upon recommendations from relatives, friends or people with good knowledge of the domain we are interested in, or we just consult articles, journals or books that deal with the object of interest. What we are after is a set of good recommendations that exactly fit our particular need. Recommendation systems are automated tools that do exactly that. Recommenders are roughly divided into two groups namely content based recommendation system and Collaborative Filtering recommendation system. Content based method makes use of information pertaining to the user like his interests, occupation, age and information pertaining to the item like title, topic etc to make predictions [1]. In this paper we won't deal with content-based recommendation system, since it is no more a hot topic in the research community. On the other hand, Collaborative Filtering (CF) introduced by D. Goldberg and friends in his paper "Tapestry" [2] in 1992 is the buzzword in the community. CF describes the collaboration among users in order to filter and identify the items of interest. CF is divided into memory based CF and model based CF. Memory based CF uses the entire dataset each time a prediction is needed. One approach of memory based CF also called nearest neighbor method uses rating matrix to compute the similarity value (which determine the level of closeness among users) and based on this value, it makes the prediction. Memory based CF is also called non-parametric model. On the other hand model-based CF or the parameterized model the database to build a model and use this model to make prediction[3]. Hence here once the model parameters are found, the dataset can be discarded altogether and only the parameters are used to make new predictions. We also have the novel trust based recommenders to solve the infamous Cold-Start problem where the prediction is done by incorporating the popular social networking elements. Trust based recommender is normally combined with either memory based or model based CF. We also have hybrid memory and model CF

Department of Information Technology, University of Electronics, Science and Technology of China
Email:lebijean@qq.com

Manuscript received ??????, 20??; revised ???????, 20??. This work was supported by the National Nature Science Foundation of China under Grant No. 88888888. **Please give the supported projects.**

J.-Z. Kang is with Editorial Office, Journal of Electronic Science and Technology, Chengdu, 610054, China (Corresponding author e-mail: zskang@intl-jest.com).
Y. Wang and X. Xuan are with Editorial Office, Journal of Electronic Science and Technology, Chengdu 610054, China (e-mail: wangyan@intl-jest.com and xiexuan@intl-jest.com).

Color versions of one or more of the figures in this paper are available online at http://www.intl-jest.com.

Digital Object Identifier: 10.3969/j.issn.1674-862X.20??.0?.0??

techniques. In this paper we will show examples of each category. In section 2, we describe memory based CF techniques. Model based techniques are addressed in section3. In section 4, we talk about Hybrid models. Section 5 contains a description of trust based models. Finally we conclude in section 6 by describing the problems still existing and provide guidance for further research in this area.

.

2. Memory based Collaborative Filtering

Memory based CF also called non-parametric method uses the entire (or part) of the dataset to compute a prediction for a new user (active user). The term non-parametric refers to the fact that the amount of data needed in order to compute a prediction grows with the size of the training set. Unlike model based where the form of the model is defined before-hand. Normally we first compute a similarity weight between our active user and all the users in the dataset. The users with a large weight are considered friends to our active user that is they share similar interests. Hence to predict a new value for our active user, we can just refer that of the friends and assign it as our prediction.

2.1-**Pearson Correlation**

The Pearson correlation is a formula for the computation of the weight value and it is given in [4]. The Pearson Correlation formula between users **a** and **u** is given by:

$$w_{a,u} = \frac{\sum_{i \in I}(r_{a,i} - \bar{r}_a)(r_{u,i} - \bar{r}_u)}{\sqrt{\sum_{i \in I}(r_{a,i} - \bar{r}_a)^2}\sqrt{\sum_{i \in I}(r_{u,i} - \bar{r}_u)^2}} \qquad (1)$$

-where the summation over **i** are over the items for which both users **a** and **u** have rated.

Some variations of Pearson correlation can be computed. We have for instance constrained Pearson correlation where midpoint are used instead of mean rate, Spearman rank correlation where ranks are used instead of ratings and Kendall's tau correlation where relative ranks are used to calculate correlation.

2.2-**Vector Cosine-Based similarity**

Given two user vectors $\vec{a}=\{a_1, a_2\}$ and $\vec{u}=\{u_1, u_2\}$, the similarity weight between them can be computed using the cosine of the angle formed by them. The weight becomes [4]:

$$w_{a,u} = \cos(\vec{a}, \vec{u}) = \frac{\vec{a}.\vec{u}}{||\vec{a}|| \bullet ||\vec{u}||} = \frac{a_1.u_1 + a_2.u_2}{\sqrt{a_1^2 + u_1^2}\sqrt{a_2^2 + u_2^2}} \qquad (2)$$

In actual situation, adjusted cosine similarity is used by subtracting the corresponding user average from each co-rated pair to remove bias from different user scaling.

2.3-**Weigted Sum**

Having found the weight we can now compute the prediction for the active user **a** on a certain item **i** by taking a weighted average of all the ratings on that item as[4]:

$$P_{a,i} = \bar{r}_a + \frac{\sum_{u \in U}(r_{u,i} - \bar{r}_u)w_{a,u}}{\sum_{u \in U}|w_{a,u}|} \qquad (3)$$

Where $\bar{r}_a$ and $\bar{r}_u$ are the average ratings for the user **a** and user **u** respectively on all the rated items and $w_{a,u}$ is the weight between the user **a** and user **u**. The summations are over all the users **u** who have rated item **i** .

2.4- **Simple Weighted Average (SWA)**

We can also use the Simple Weighted Average (which made use of item-based CF) to predict the rating $P_{a,i}$ for active user **a** on item **i** [5] using:

$$P_{a,i} = \frac{\sum_{n \in N} r_{a,n} w_{i,n}}{\sum_{n \in N}|w_{i,n}|} \qquad (4)$$

where the summation are over all other rated items **n**for user **a**. $w_{i,n}$ is the weight between items **i** and **n** and $r_{a,n}$ is the rating for user **a** on item **n**.

3. Model based CF

Instead of maintaining continuously the entire dataset in memory, an alternative would be to learn the parameters for a model from the dataset and discard the dataset altogether. And prediction is done solely with the model parameters and the input variable. This technique is called model based CF.

3.1- **Bayesian Belief Net (BN) CF**

A Bayesian BN is a Directed, Acyclic Graph (DAG) with a triplet <N,A,θ> where n of N represents a random variable, each directed arc a of A is the probabilistic association between variables and $\theta$is a conditional probability table [6]. Bayesian BNs are often used for classification. Bayesian theorem:

$$P(z|\mathbf{x}) = \frac{P(z)P(x|z)}{P(x)} \qquad (5)$$

Compute this posterior probability for each class and select the class having the highest probability as the prediction. Also the Laplace smoothing can be used in order to avoid a conditional probability of zero (i.e. the element has not yet been observed):

$$P(class_z|\mathbf{x}) = \frac{P(class_z, x)+1}{P(x)+|class_z|} \qquad (6)$$

where $|class_z|$ is the size of the class set.
It is also called the Simple Bayesian CF when it uses the Naive Bayes (NB) to make prediction. The naive Bayes assumes the variable attributes are independent of each other. Hence the covariance matrix is simply a diagonal matrix. Because of the limitation of the Simple Bayesian CF, advanced BNs have been devised[7]. We have for instance the Extended Logistic Regression (ELR) which is basically a gradient ascent discriminative algorithm which maximizes the log conditional likelihood.

3.2-**Clustering CF**

A cluster is a collection of data objects that are similar to one another within the sane cluster and are dissimilar to the objects in other clusters[8]. The measurement of the similarity between objects is determined using metrics such as Minkowski distance and Pearson Correlation. For two data objects **x**={x_1,...., x_D} and **y**={ y_1,....., y_D}, the Minkowski distance is:

$$d(\mathbf{x}, \mathbf{y}) = \sqrt{\sum_{i=1}^{D} |x_i - y_i|^q} \qquad (7)$$

When q=1, d is called the Manhattan distance and when q=2, d is called the Euclidean distance.
Clustering methods can be classified into 3 categories: partitioning methods, density-based methods and hierarchical methods .

3.3-**Regression-Based CF**

A regression method uses an approximation of the ratings to make prediction based on a regression model given by:

$$\mathbf{Y} = \mathbf{WX} + \mathbf{E} \qquad (8)$$

where **Y** is an N$\times$D matrix with $y_{n,d}$ representing the rating of user **n**on item **d, W** is the K$\times$D matrix, **E**={e_1,......,e_D} represents the noise in user choices, **X** is a N$\times$K matrix with each row as an estimate of the value of the random variable **x** for one user.
The matrix Y is very sparse. To remedy this, Canny [9] proposed a sparse factor analysis which replaces missing elements with default values and uses the regression model as the initialization for EM (Expectation Maximization) iteration.

3.4-**MDP (Markov Decision Process)**

An MDP is a model for optimization of sequential stochastic decision problem where an agent influences its surrounding environment through actions. An MDP can be defined as a four-tuple: <S,A,R,Pr>, where S is the set of states, A is the set of actions, R is the reward function for each state/action pair and Pr is the transition probability between every pair of states given each action. The objective of the MDP is to maximize the function of its reward stream. By starting with an initial policy π_0 (s)= $\arg\max_{a \in A} R(s,a)$, computing the reward value function Vi(s) based on the previous policy and updating the policy with the new value function at each step, the iteration will converge to an optimal policy [10].

3.5-**Latent Semantic CF**

A latent semantic CF relies on a statistical modeling technique that introduces latent class variables in a mixture model setting to discover user communities and prototypical interest profiles. Conceptually, it decomposes user preferences using overlapping user communities. The aspect model [11] is a probabilistic latent-space model, which models individual ratings as a convex combination of rating factors. A latent class variable is a associated with each observed pair of {user,item} with the assumption that users and items are independent from each other given the latent class variable.

4. Hybrid CF

Each type of recommender system seen so far has its share of obstacles[6]. For instance memory based CF methods are dependent on human ratings so vulnerable to shilling attacks, rating performance decrease when the data are sparse (sparsity problem), cannot recommend for new users and items (Cold Start problem) and have limited scalability. When model based CF methods while addressing some of the memory based problems have their own share of problems too like for instance the Cold Start problem which they find it difficult to address. Hence in order to solve all the obstacles, hybrid recommenders have been designed. Hybrid recommenders may combine CF methods with content based methods like in Content Boosted CF algorithms[12] or they may just combine memory based CF with model based CF like in Personality Diagnosis [13]. In the next section, we describe trust based CF which itself is also another hybrid CF method.

5. Trust-Based CF

The latest technique in this area is the trust based system which combines the rating matrix and the social network database for better prediction. This technique address successfully most of the problems encountered previously specially the slow start up commonly called Cold Start problem. It is easily understood. For though the new user has no history with us, through his social network information we can find the trust weight between him and his friends. Thus deriving his preferences. We have various implementation of the trust based methods shown below.

5.1- Tidal Trust recommendation system

The Tidal Trust recommendation system [14] performs a modified breadth-first search in the system. It computes the trust value based on all the raters at the shortest distance from the target user. The trust between users $\mathbf{u}$ and $\mathbf{v}$ is:

$$t_{\mathbf{u},\mathbf{v}} = \frac{\sum_{\mathbf{w}\in\mathbf{N}} (t_{\mathbf{u},\mathbf{w}} * t_{\mathbf{w},\mathbf{v}})}{\sum_{\mathbf{w}\in\mathbf{N}} t_{\mathbf{u},\mathbf{w}}} \qquad (9)$$

Where $\mathbf{N}$ denotes the set of the neighbors of $\mathbf{u}$.
The trust depends on all the connecting paths. The predicted rating is computed as:

$$r_{\mathbf{u},\mathbf{i}} = \frac{\sum_{\mathbf{v}\in raters} (t_{\mathbf{u},\mathbf{v}} * r_{\mathbf{v},\mathbf{i}})}{\sum_{\mathbf{v}\in raters} t_{\mathbf{u},\mathbf{v}}} \qquad (10)$$

Where $r_{\mathbf{v},\mathbf{i}}$ denotes rating of user $\mathbf{v}$ for item $\mathbf{i}$.

5.2- MoleTrust recommendation system

We also have the MoleTrust recommendation system [15]. It is very similar in its operation to the previous technique (TidalTrust). The only difference is that instead of focusing on users at the shortest distance, it rather considers raters up to a maximum-depth d. Tuning appropriately d gives a very good result. Hence the MoleTrust gives a better recommendation the TidalTrust.

5.3- TrustWalker recommendation system

The TrustWalker method[16] too is similar in its proceedings to the MoleTrust but instead of the far friends who have rated the target item, we use the near neighbors who have rated similar items. Here we add the item-based CF to the normal trust based recommender to have a much more powerful technique. The main question here is how do we define similarity between two items. The similarity is given by:

$$sim(\mathbf{i},\mathbf{j}) = \frac{corr(\mathbf{i},\mathbf{j})}{1+e^{-\frac{|corr(\mathbf{i},\mathbf{j})|}{2}}} \tag{11}$$

Having already computed the trust value among users, we just look for similar items that were rated by the trusted members and assign the value to our active user.

5.4- **LebiD1 Trust-based recommender**

LebiD1 [17] is also a trust based technique. It combines the memory-based technique with the social network information to solve the cold start problem.

When applied to movielens-QQ and Linkedln, it outperforms the previous trust based systems in recommending items (movies) to users. The formula used in LebiD1 is:

$$r_{\mathbf{a},\mathbf{i}} = \frac{\sum_{\mathbf{u}\in\mathbf{F}} r_{\mathbf{u},\mathbf{i}}}{card(\mathbf{F})} \quad \mathbf{i} \in \mathbf{I} \tag{12}$$

Where $\mathbf{F}$ represents the closest friends of active new user $\mathbf{a}$, $\mathbf{I}$ is the set of all the items rated by all the users $\mathbf{u}$ of $\mathbf{F}$, denotes the predicted rating of novel user **a** to item **i**, **F** denotes the set of social network friendshaving rated item **i**, denotes the rating of friend user **u** to item **i** and finally *card* (**F**) denotes the number of such friends.

5.4- **LebiD2 Trust-based recommender**

Last but not least is the LebiD2 trust-based recommender[18]. We have already seen the problems associated with the memory-based recommender [4]. So instead of the memory based paradigm of LebiD1, LebiD2 uses rather the model based approach to trust based recommender. Indeed LebiD2 is an enhancement of the LebiD1. The results are amazing on Movielens-QQ and Linkedln. The model based formula used in LebiD2 is:

$$\mathbf{W} = (\mathbf{X}^T\mathbf{X})^{-1}\mathbf{X}^T.t \tag{13}$$

Where $\mathbf{W}$ represents the model parameters needed and it is solved using normal maximum likelihood (to avoid overfitting, a regularizer parameter can be applied). $\mathbf{X}$ represents the user dataset and $\mathbf{t}$ represents the given movie ratings .

6. Conclusions

In this paper, we have covered some of the recommendation systems in vogue in the recommendation system community. We have specially discussed the novel trust based recommenders which incorporate social networks information to the CF technique thus solving the cold start problem. For future research, we will encourage researches to focus on the CF applied in a mobile application context and be prudent about the noisy data that can be found in such an environment. Also we encourage researchers to use dataset from live environment instead of artificial data.

Acknowledgment

This research was supported by the NSFC under grant No. 61133016, 61202445, 61103206

References

[1] S. Chen, B. Mulgrew, and P. M. Grant, "A clustering technique for digital communications channel equalization using radial basis function networks," *IEEE Trans. on Neural Networks*, vol. 4, pp. 570–578, July 1993.

[1] S.Debnath, N. Ganguly, and P. Mitra, "Feature weighting in content based recommendation system using social network analysis. In Proceedings of the 17th international conference on World Wide Web, pages 1041-1042, 2008.

[2] D. Goldberg, D. Nichols, B. M. Oki, and D. Terry, "Using collaborative filtering to weave an information tapestry," Communications of ACM, vol. 35, no. 12, pp. 61-70, 1992.

[3] T. Hofmann, "Latent semantic models for collaborative filtering," ACM Transactions on Information Systems, vol. 22, no. 1, pp.89-115, 2004

[4] X. Su and T.M. Khoshgoftaar, "A Survey of Collaborative Filtering Techniques", Advances in Artificial Intelligence, 2009:4:2-4:2, January 2009.

[5] B.M. Sarwar, G. Karypis, J. Konstan and J. Riedl, "Item-based collaborative filtering recommendation algorithms", in Proceedings of the 10th International Conference on World Wide Web (WWW'01), pp.285-295, May 2001.

[6] J. Pearl,"Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference", in Morgan Kaufmann, San Fransisco, California, USA 1998.

[7] X. Su, T.M. Khoshgoftaar, " Collaborative filtering for multi-class data using belief nets algorithms", in Proceedings of the International

Conference on Tools with Artificial Intelligence (ICTAI'06), pp.497-504, 2006.

[8] L. H. Ungar and D.P. Foster, "Clustering methods for collaborative filtering", in Proceedings of the Workshop on Recommendation Systems, AAAI Press, 1998.

[9] J. Canny, "Collaborative filtering with privacy via factor analysis", in Proceedings of the 25th Annual International ACM SIGIR Conference on Research and Development in Information Retrieval, pp. 238-245, Tampere, Finland, August 2002.

[10] R. A. Howard, "Dynamic Programming and Markov Processes", MIT Press, Cambridge, Mass, USA, 1960.

[11] T. Hofmann and J. Puzicha, "Latent class models for collaborative filtering", in Proceedings of the 16th International Joint Conference on Artificial Intelligence (IJCAI '99), pp. 688-693, 1999.

[12] P. Melville, R. J. Mooney, and R. Nagarajan, “Content-boosted collaborative filtering in dynamic, sparse, high-dimensional domains”, in Proceedings of the 18th National Conference on Artificial Intelligence (AAAI ’02), pp. 187-192, Edmonton, Canada, 2002.

[13] D.M. Pennock, E. Horvitz, S. Lawrence and C.L. Giles, "Collaborative filtering by personality diagnosis: a hybrid memory and model based approach", in Proceedings of the 16th Conference on Uncertainty in Artificial Intelligence (UAI '00), pp.473-480, 2000.

[14] J. Golbeck, "Computing and Applying Trust in Web-based Social Networks", PhD thesis, University of Maryland College Park, 2005.

[15] P. Massa, P. Avesani, "Trust-aware recommender systems", RecSys 2007

[16] M. Jamal, M. Ester, "TrustWalker: A random walk model for combining trust-based and item-based recommendation", KDD 2009.

[17] J.M Dali and Q. ZhiGuang, "Cold Start Mastered: LebiD1", International Conferences on Computational Science and Engineering (CSE2014), Chengdu China, 2014.

[18] J.M Dali and Q. ZhiGuang, "Solving the Cold-Start with LebiD2", International Conference on Control Engineering and Electronics Engineering (CCEEE 2014), Chengdu China, 2014

Lebi Jean-Marc Dali was born in Abidjan, Ivory Coast, in 1987. He received the B.C.A. (Bachelor Computer Application) degree from Bangalore University (India) in 2010 and the M.C.A (Master Computer Application) degree from the same University, in 2013. He is currently pursuing the Ph.D. degree with the Department of Information Technology UESTC (University of Electronic, Science and Technology of China). His research interests include machine learning, recommendation systems and information security.

ProfQin ZhiGuang received his PhD from UESTC (University of Electronic, Science and Technology of China) in 1996. He is currently the Dean of the Information Technology Department. His research interests include machine learning, recommendation system and information security