Hybrid collaborative filtering recommendation algorithm based on user and item

Ping-Ping Lai, Sun Yat-sen University, Guangzhou, China

Email:1684676912@qq.com

***Abstract*—Collaborative filtering algorithm is one of the most widely used and successful recommendation technologies. However, the data sparsity problem and the accuracy problem of the traditional collaborative filtering algorithm seriously affect the accuracy of the recommendation. In response to these problems, this paper proposes a hybrid collaborative filtering algorithm based on user and item. The algorithm uses the user interest preference to reduce the sparsity of the data by predicting the rating of unrated item, and then calculates the prediction rating based on the user and item algorithm respectively. The user features and item categories are introduced when calculating the similarity between the user and the item so that the determination of the nearest neighbor is more accurate, then combine the results of the two by similarity weights, and finally generate recommendations based on the predicted ratings. Experiment with Movielens 100k dataset shows that the proposed algorithm is superior to the traditional user-based and item-based collaborative filtering method, which alleviates the data sparsity problem and improves the accuracy of prediction, thus improving the recommendation quality.**

***Key words*— collaborative filtering , user interest distribution, user characteristics, item attributes, similarity weights**

I. INTRODUCTION

The rapid development of the Internet has brought about rapid growth of information. It is more difficult to obtain valuable information in massive data. Personalized recommendation algorithms have become one of the most effective techniques to solve this problem[1]. At present, almost all large-scale business websites, such as Amazon, Ebay, Taobao, Dangdang and other websites use their own recommendation system [2][3].

Collaborative filtering is considered to be one of the most successful and widely used personalized recommendation technologies to date. Its core idea is to measure the similarity between users based on the user's rating of the item, and to find neighbor users with the most similar interests to the target users. Then, the neighbor user's rating of the item is used to predict the target user's preference for the project, thereby generating a recommendation [4][5]. The existing collaborative filtering recommendation algorithms are mainly divided into two categories: (1) a neighbor-based (memory-based) algorithm that generates recommendations based on similar neighbors; and (2) a model-based algorithm that generates a scoring model to generate recommend. It has been pointed out that the algorithm based on neighbors can obtain better recommendation accuracy, but it cannot solve the scalability problem caused by the increase of data volume; the model-based algorithm has better scalability, but the model cannot expresses the diversity of user hobbies, so it is not as good as the neighbor-based algorithm in terms of recommendation quality.

Sarwar et al.[6] differentiated content-based (neighbor-based) algorithms into user-based collaborative filtering algorithms [7] and item-based collaborative filtering algorithms [8] based on the correlation between things used in collaborative filtering[9]. The user-based algorithm generates recommendations based on the similarity of the users, and the item-based algorithm generates recommendations based on the similarity of the items. User- or item-based algorithms simply consider the unilateral impact of the user or item, ignoring the connection between the user and the item. On the other hand, traditional user-based or project-based algorithms only consider the impact of scoring data, while ignoring other factors such as user characteristics and item attributes. Whether it is user-based or item-based collaborative filtering, there is a problem of data sparsity. In response to these problems, this paper proposes a hybrid collaborative filtering algorithm based on users and items. The algorithm uses the user interest reference to reduce the sparsity of the data by the score matrix prediction, and then calculates the prediction score based on the user and item algorithm respectively. The user characteristics and item attributes are introduced when calculating the similarity between the user and the item, so that the determination of the nearest neighbor is more

accurate, then combine the results of the two by similarity weights, and finally generate recommendations based on the predicted scores. The experimental results show that the proposed algorithm is superior to the traditional user-based and item-based collaborative filtering method, which alleviates the data sparsity problem and improves the prediction accuracy, thus improving the recommendation quality.

II. THE RELATED CONCEPTS

A. User-item rating matrix

The user-item rating matrix is a matrix R of mxn, where m is the number of users, n is the number of items, and Ri,j represents the rating of user i on item j

Table 1 user-item rating matrix

	$Item_1$	…	$Item_j$	…	$Item_n$
$User_1$	$R_{1,1}$	…	$R_{1,j}$	…	$R_{1,n}$
…	…	…	…	…	…
$User_i$	$R_{i,1}$	…	$R_{i,j}$	…	$R_{i,n}$
…	…	…	…	…	…
$User_m$	$R_{m,1}$	…	$R_{m,j}$	…	$R_{m,n}$

B. Item-Category Matrix

The Item-Category matrix is a matrix G of nxt, where n is the number of items, t is the total category of all items, G_{ij}= 1 means that the item i belongs to category j. G_{ij}= 0 means that the item i don't belong to category j.

Table 2 Item-Category Matrix

	$category_1$	…	$category_j$	…	$category_t$
$Item_1$	$G_{1,1}$	…	$G_{1,j}$	…	$G_{1,t}$
…	…	…	…	…	…
$Item_i$	$G_{i,1}$	…	$G_{i,j}$	…	$G_{i,t}$
…	…	…	…	…	…
$Item_n$	$G_{n,1}$	…	$G_{n,j}$	…	$G_{n,t}$

C. User feature matrix

The user feature matrix is a matrix B of mxq, where m is the number of users, q is the number of user features, and B_{ij} is a value representing the j feature of user i.

Table3 User feature matrix

	$Feature_1$	…	$Feature_j$	…	$Feature_q$
$User_1$	$B_{1,1}$	…	$B_{1,j}$	…	$B_{1,q}$
…	…	…	…	…	…
$User_i$	$B_{i,1}$	…	$B_{i,j}$	…	$B_{i,q}$
…	…	…	…	…	…
$User_m$	$B_{m,1}$	…	$B_{m,j}$	…	$B_{m,q}$

D. User Interest Preference Matrix (User-Category Matrix)

The user interest preference matrix is a matrix D of mxt, where m is the number of users and t is the number of item categories. D_{ij} represents the average of the rating of the items belonging to category j among the rating items of user i, and the user interest preference matrix is obtained by the user-item matrix and the item-category matrix.

Table 4 User Interest Preference Matrix

	$category_1$	…	$category_j$	…	$category_t$
$User_1$	$D_{1,1}$	…	$D_{1,j}$	…	$D_{1,t}$
…	…	…	…	…	…
$User_i$	$D_{i,1}$	…	$D_{i,j}$	…	$D_{i,t}$
…	…	…	…	…	…
$User_m$	$D_{m,1}$	…	$D_{m,j}$	…	$D_{m,t}$

E. Similarity measure

Pearson coefficient

$$\mathrm{sim}(i,j) = \frac{\sum_{u\in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u\in U}(R_{u,i} - \bar{R}_i)^2}\sqrt{\sum_{u\in U}(R_{u,j} - \bar{R}_j)^2}}$$

U is represented as a set of users who have a common rating for item i and j, and $\bar{R}_i$ and $\bar{R}_j$ are the average values of the ratings of the item i and j for the users who rating the items i, j respectively. $R_{u,i}$ is the user u's rating of item i, and $R_{u,j}$ is the user u's rating of item j.

III. THE PROPOSED ALGORITHM

. In this section, we propose a hybrid collaborative filtering algorithm based on users and items.

A. Predict fill vacancy values based on user interest preference matrix

The increase in the size of the user and the number of items makes the data sparsity of the rating matrix gradually increase, resulting in low accuracy of the predicted rating. The current solution is to pre-populate the user's unrated items. The most common and simple method is to fix the default value, which can effectively improve the system's recommendation accuracy. However, the default values in the rating matrix in real life may not be the same, so the method of populating the fixed default does not fundamentally solve the problem of data sparsity. In this paper, the user interest preference matrix is used to find similar users, and the vacancy value is predicted and filled, which reduces the sparseness of user rating data.

We divide the item into different categories, and then judge the user's preference for each category based on the user's rating of the item and the category to which the item belongs.

The calculation formula of the user interest preference matrix D is

$$D_{uj} = \frac{\sum_{i \in I} P_{ui}}{N} \quad (1)$$

where I represents the item belonging to category j among the items rated by user u, P_{ui} represents the rating of user u for item i, and N represents the number of items in I.

In conjunction with the user interest preference matrix, the Pearson coefficient is used to calculate the similarity between users.

$$\mathrm{sim}_D(\mathrm{u},\mathrm{v}) = \frac{\sum_{i \in I}(R_{u,i} - \bar{R}_u)(R_{v,i} - \bar{R}_v)}{\sqrt{\sum_{i \in I}(R_{u,i} - \bar{R}_u)^2}\sqrt{\sum_{i \in I}(R_{v,i} - \bar{R}_v)^2}} \quad (2)$$

where I is represented as a collection of categories representing the common rating of user u and v, and $\bar{R}_u$ and $\bar{R}_v$ are the average value of the ratings of the user u and user v for the categories in I respectively. $R_{u,i}$ is the user u's rating of category i, and $R_{u,j}$ is the u's rating of category j.

The user interest preference matrix is used as the input of the nearest neighbor, and the predicted rating is performed by the user-based collaborative filtering algorithm. The predicted rating for item i that is not rated in user u is

$$P_{D_{u,i}} = \frac{\sum_{v \in K(u)} \mathrm{sim}_D(\mathrm{u},\mathrm{v}) P_{D_{v,i}}}{\sum_{v \in K(u)} \mathrm{sim}_D(\mathrm{u},\mathrm{v})} \quad (3)$$

Where $\mathrm{sim}_D(\mathrm{u},\mathrm{v})$ is the similarity obtained by user u and user v through the user interest preference matrix, $P_{D_{v,i}}$ is the rating of similar user v for item i, and $K(u)$ is K neighbors with highest similarity to user u.

B. Predict rating based on user ratings and user's features

The traditional user-based collaborative filtering only considers the user's simple rating relationship when calculating the similarity between users, and does not consider the influence of user's features on user similarity, resulting in inaccurate similarity results, resulting in inaccurate nearest neighbors. Therefore, we propose a method for calculating user similarity in combination with user features, adding the similarity of user features in the user similarity based on user rating.

The traditional user-based collaborative filtering only considers the user's simple rating relationship when calculating the user similarity, and does not consider the influence of user features on user similarity, resulting in inaccurate similarity results and inaccurate nearest neighbors, which ultimately lead to inaccurate recommendations. People with the same user features may have similar hobbies. For example, female college students may prefer romantic romance, and boys prefer action movies and so on. Thus, combining the similarity of user features with the traditional user-based computational similarity method can improve the recommendation accuracy of the recommendation system.

$$\mathrm{sim}(\mathrm{u},\mathrm{v}) = a \cdot \frac{\sum_{i \in I}(R_{u,i} - \bar{R}_u)(R_{v,i} - \bar{R}_v)}{\sqrt{\sum_{i \in I}(R_{u,i} - \bar{R}_u)^2}\sqrt{\sum_{i \in I}(R_{v,i} - \bar{R}_v)^2}} + (1-a)\frac{\sum_{k=1}^{N}(u_k == v_k)}{N} \quad (4)$$

Where a represents the weight of the user's rating for the user's similarity, and 1-a represents the weight of the user's features for the user's similarity. N refers to the number of features of the user, and $u_k$and v_k represents the value of the kth user feature of the user u and user v respectively. I is represented as a collection of items representing the common rating of user u and v, and $\bar{R}_u$ and $\bar{R}_v$ are the average value of the ratings of the user u and user v for the items in I respectively. $R_{u,i}$ is the user u's rating of item i, and $R_{u,j}$ is the user u's rating of item j

The K nearest neighbor users are found according to the user similarity, and the predicted rating is performed by the user-based collaborative filtering algorithm. The predicted rating for item i that is not rated in user u is:

$$P_{u,i_{user}} = \frac{\sum_{v \in K(u)} sim(u,v) P_{v,i}}{\sum_{v \in K(u)} sim(u,v)} \quad (5)$$

Where $P_{v,i}$ is the rating of similar user v for item i, K(u) is the K neighbor with the highest similarity to user u.

C. Predict rating based on item ratings and item's categories

The traditional item-based collaborative filtering only considers the user-item scoring matrix data when calculating the similarity between items, and does not consider the influence of item's categories on item similarity, resulting in inaccurate similarity results and inaccurate nearest neighbors, which ultimately leads to inaccurate recommendation results. The categories of the target item's neighbor item should have certain similarities with the target item's categories. For example, if the target item is an action movie, then the neighbor item we generated is more likely to be an action movie. Therefore, combining the similarity of item categories with the traditional item-based computational similarity method can improve the recommendation accuracy of the recommendation system.

$$\mathrm{sim}(\mathrm{i},\mathrm{j}) = b \cdot \frac{\sum_{u \in U}(R_{u,i} - \bar{R}_i)(R_{u,j} - \bar{R}_j)}{\sqrt{\sum_{u \in U}(R_{u,i} - \bar{R}_i)^2}\sqrt{\sum_{u \in U}(R_{u,j} - \bar{R}_j)^2}} + (1-b)\frac{\sum_{k=1}^{N}(i_k == j_k)}{N} \quad (6)$$

Where b represents the weight of the item's rating for the user's similarity, and 1-b represents the weight of the item's categories for the item similarity. N refers to the number of categories of the item, and $i_k$and j_k represents the value of the kth item category of the item i and item j respectively. U is represented as a set of users who have a common rating for item i and j, and $\bar{R}_i$ and $\bar{R}_j$ are the average values of the ratings of the item i and j for the users who rating the items i, j respectively. $R_{u,i}$ is the user u's rating of item i, and $R_{u,j}$ is the user u's rating of item j.

The K nearest neighbor items are found according to the item similarity, and the predicted rating is performed by the item-based collaborative filtering algorithm. The predicted

rating for item i that is not rated in user u is:

$$P_{u,i_{item}} = \frac{\sum_{j \in K(i)} sim(i,j) P_{u,j}}{\sum_{j \in K(i)} sim(i,j)} \quad (7)$$

Where $P_{u,j}$ is the rating of user u for similar item j, and K(i) is the K neighbor with the highest similarity to item i.

D. Combine user-based and item-based collaborative filtering algorithms with similarity weights

The user-based recommendation algorithm generates recommendations based on the user similarity, and the item-based recommendation algorithm generates recommendations based on the item similarity. Both algorithms only consider the unilateral impact of the user or item, ignoring the relationship between the user and the item. If the two algorithms can be combined to achieve the purpose of improving the prediction accuracy, then a better recommendation quality can be obtained.

In order to combine the respective favorable factors of user-based and item-based algorithms, we introduce weighting factors λ_u and λ_i, which λ_u is the weighting influence of the user-based algorithm on the predicted rating, and λ_i is the weighting influence of the item-based algorithm on the predicted rating.

$$\lambda_u = \frac{\sum_{v \in K(u)} sim(u,v)^2}{\sum_{v \in K(u)} sim(u,v)^2 + \sum_{j \in K(i)} sim(i,j)^2} \quad (8)$$

$$\lambda_i = \frac{\sum_{j \in K(i)} sim(i,j)^2}{\sum_{v \in K(u)} sim(u,v)^2 + \sum_{j \in K(i)} sim(i,j)^2} \quad (9)$$

where $\lambda_u + \lambda_i = 1$.

The weighted predicted rating is

$$P_{u,i} = \lambda_u \cdot P_{u,i_{user}} + \lambda_i \cdot P_{u,i_{item}} \quad (10)$$

E. Recommend based on top-N algorithm

After calculating the user's predicted rating for the unrated item, the N items with the best predicted rating are recommended to the user.

IV. EXPERIMENT

A. Dataset

The experimental data is based on the MovieLens dataset provided by the US GroupLens project team, and the size is Movielens 100k[10]. MovieLens is a web-based research recommendation system that receives user ratings for movies and provides a list of recommendations. The data set contains 100 000 ratings for 1,682 movies from 943 users, with ratings ranging from 1 to 5, and each user rated at least 20 movies. The sparse rating of the rating data is 1-100000/943X1682=0.9369. And in this data set we can also get information about the items, as well as demographic information about the user. In this experiment we used 80% of them as a training set and 20% as a test set.

B. Recommend based on top-N algorithm

In this experiment, the mean absolute error MAE is used to evaluate the recommended accuracy of the algorithm[11]. The accuracy of the prediction is measured by calculating the deviation between the predicted user rating and the actual rating. The smaller the MSE, the more accurate the prediction and the higher the recommended quality.

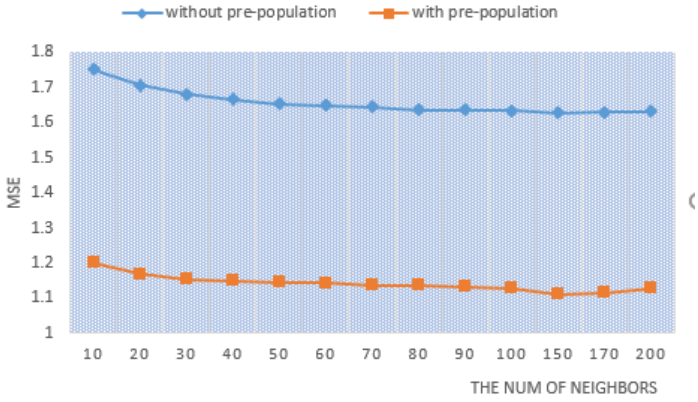
MSE is defined as follows

$$\text{MAE} = \frac{\sum_{i=1}^{N} |R_{u,i} - \hat{R}_{u,i}|}{N} \quad (11)$$

where $R_{u,i}$ is the actual rating of user u on item i, $\hat{R}_{u,i}$ is the predicted rating of user u on item i, N is the number of items for which the rating is predicted.

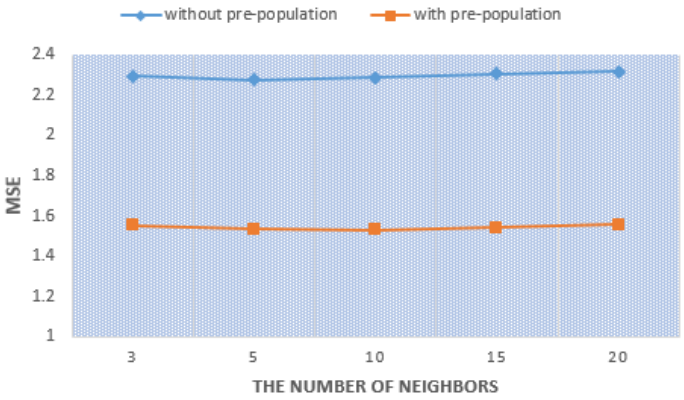
C. Experimental process and experimental results

（1）we firstly pre-populate the user-item rating matrix according to the user interest preference matrix, and then compare the experimental results of the pre-populated user-based collaborative filtering algorithm and the user-based collaborative filtering algorithm without pre-populating.



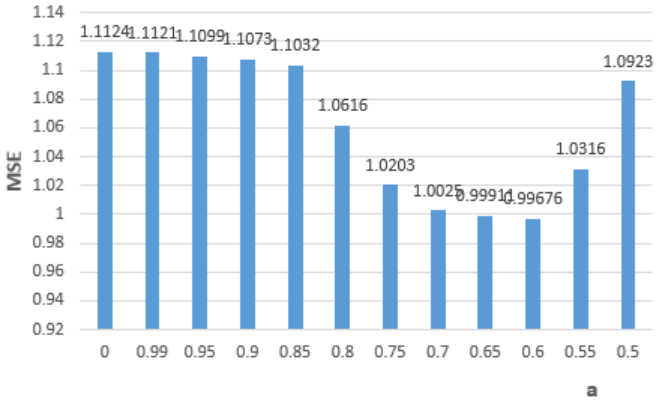
According to the experimental result, we can know that the method with pre-population is superior to the traditional user-based.

（2）we firstly pre-populate the user-item rating matrix according to the user interest preference matrix, and then compare the experimental results of the pre-populated item-based collaborative filtering algorithm and the item-based collaborative filtering algorithm without pre-populating.



According to the experimental result, we can know that the method with pre-population is superior to the traditional item-based.

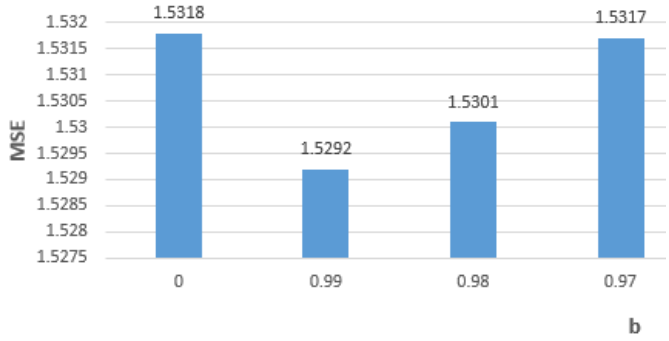
（3）we do this experiment based on the user-based collaborative filtering algorithm with the improved similarity calculation formula (4) and then compare the experimental result with the user-based collaborative filtering algorithm using the original similarity calculation formula.



a=1means using the original similarity.

According to the experimental result, we can know that the method with formula (4) is superior to the used-based algorithm with the original similarity.

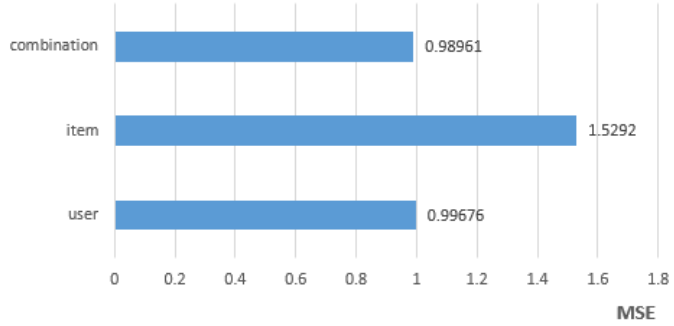
（4）we do this experiment based on the item-based collaborative filtering algorithm with the improved similarity calculation formula (6) and then compare the experimental result with the item-based collaborative filtering algorithm using the original similarity calculation formula.



b=1means using the original similarity.

According to the experimental result, we can know that the method with formula (6) is superior to the used-based algorithm with the original similarity.

(7) the experiment uses similarity weight to combine experiment (3) and (4), and then compare the experiment result of this experiment with the result of (3) and (4).



The experimental results show that the proposed algorithm is superior to the traditional user-based and project-based collaborative filtering method, which alleviates the data sparsity problem and improves the prediction accuracy.

V. CONCLUSION

Aiming at the accuracy problem and sparsity problem of traditional collaborative filtering method, this paper proposes a hybrid collaborative filtering recommendation method based on user and item. The algorithm uses the user interest preference to reduce the sparsity of the data by predicting the rating of unrated item. The predictive rating of the user and item algorithms introduces user features and item categories when calculating user and item similarity so that the determination of the nearest neighbors is more accurate, then combines the results of the two by similarity weights, and finally generates recommendations based on the predicted rating. The experimental results show that the proposed algorithm is superior to the traditional user-based and project-based collaborative filtering method, which alleviates the data sparsity problem and improves the prediction accuracy.

REFERENCES

[1] Francisco Ritchie FRANCESCORICCI. Recommended Systems: Technology, Evaluation and Efficient Algorithms [M]. Mechanical Industry Press, 2015

[2] SCHAFER JB,KONSTAN JA，RIED J.E-commerce recommendation applications［J］. Data Mining and Knowledge Discovery，2001，5 (1 /2) : 115 － 153

[3] SARWAR B，KARYPIS G，KONSTAN J，et al. Analysis of recommendation algorithms for e-commerce［C］/ / ACM Conference on Electronic Commerce. New York: ACM，2000: 158 － 167.

[4] HERLOCKER L J，KONSTAN A J，RIEDL T J. Empirical analysis of design choices in neighborhood-based collaborative filtering algorithms［J］. Information Retrieval，2002，5(4) : 287 － 310

[5] BREESE J，HECHERMAN D，KADIE C. Empirical analysis of predictive algorithms for collaborative filtering［EBOL］. ［2010 － 10 －05 ］.

[6] Sarwar,B.M.,Karypis,G.,Konstan,J.A.,and Ricdl,J.,Item-based Collaborative filtering Recommendation Algoriths [C],Proceedings of the 10th International World Wide Web Conference,pp.285-295,ACM Press,2001.

[7] Sarwar B,Konstan J,Borchers A,et a1.Using filtering agents to improve prediction quality in the groupLens research collaborative filtering system [C].In:Proc. ACM Conf. Computer Supported Cooperative Work(CSCw). New York:ACM Press,1998,345～354.

[8] Linden G, Smith B, York J, Amazon.com recommendations:Item-to-item collaborative filtering[J]. IEEE Internet Computing,2003,7(1):76-80
[9] Deshpande M, Karypis G, Item-based top-n recommendation algorithms[J]. ACM Transactions on Information Systems,2004.22(1):143-177
[10] https://grouplens.org/datasets/movielens/
[11] Lai S, Xiang L, Diao R, et al. Hybrid Recommendation Models for Binary User Preference Prediction Problem[J]. Journal of Machine Learning Research-Proceedings Track, 2012.13(5):137-151.