2020 6th International Conference on Advanced Computing & Communication Systems (ICACCS)

Applying Matrix Factorization In Collaborative Filtering Recommender Systems

R.Barathy
Computer Science and Engineering
Thiagarajar College of Engineering
Madurai, India
barathyraman18@gmail.com

P.Chitra
Computer Science and Engineering
Thiagarajar College of Engineering
Madurai, India
pccse@tce.edu

***Abstract*— Collaborative filtering plays a vital part in advancing the recommendation environment by using the matrix factorization (MF) decomposition technology which is demonstrated to be most successful recommendation strategies. Despite being a successful method used in recommendation systems, SVD-based methods suffer from the data sparsity problem, which leads to inaccurate prediction of ratings. This paper proposes an incorporation-based recommendation method, to address the issue of sparsity in SVD-based strategies. Initially, similar users and items are found. Then, data is generated according to co-rated values. Finally, the data is incorporated into the SVD framework. We implemented our method on MovieLens 100k dataset. Experiment results represent that our method of prediction is better than the existing system.**

Keywords*— *Collaborative filtering (CF), Matrix Factorization (MF), Singular Value Decomposition (SVD), Recommendation System (RS) ,Sparsity.

I. INTRODUCTION

Recommender systems made a significant growth with the proposal of various content-based, collaborative, and hybrid methods and the development of several industrial-strength systems. The method of adding data to increase the quality of SVD based recommendation, alleviated the data sparsity and also provided an improved recommendation. Utilizing additional information for inadequate rating framework has been proven to improve the existing system. Much work done has been carried out both in the industry and the scholarly community on developing new approaches to recommender systems over the last decade.

The interest in this recommender framework still remains high because it constitutes a problem-rich research area and there are numerous practical applications that are loaded with information and provide recommendations which help the users to handle the information overload. Cases of such applications incorporate suggesting Books, CDs, and other items at Amazon.com, motion pictures by MoviesLens. In spite of all of these propels, the current era of recommender frameworks still requires further advancements to handle Sparsity and Cold Start issues. Better strategies are required to represent the user behavior.

A Hybrid recommender system is built by leveraging auxiliary information like textual description and images in collaborative filtering to handle the data sparsity problem and cold start problem which has proven to be an effective approach for online recommendation [1].The issue of cold start problem is addressed by using side information about the users and items which while used with matrix factorization and neural networks has shown significant improvements over existing approaches [2].

A three dimensional matrix factorization method is devised with each dimension for each of the three Semantic attitudes including sentiment, volume and objectivity extracted from user –generated content. Temporal alterations are also considered in the factorization model [3].

These days, the exponential progression of social systems is making unused application regions for RS. People-to-people RS point to exploit user's interface for recommending important individuals to take after. In any case, conventional recommenders don't consider that individuals may share comparable interface, but might have diverse sentiments or suppositions almost them. In this paper, a novel proposal motor which depends on the recognizable proof of semantic demeanors, that's, estimation, volume, and objectivity, extricated from user-generated substance. In arrange to do this at large-scale on conventional social systems, a three-dimensional MF, one for each state of mind. Potential worldly modifications of users' attitudes are moreover taken into thought within the factorization demonstrate. Broad offline tests on diverse genuine world datasets, uncover the benefits of the proposed approach compared with a few state-of-the-art procedures [4].

Dual Regularized Matrix Factorization (DRMF) with deep neural network is employed to effectively extract the description documents of users and items which aids in alleviating the problem of data sparsity [5].

Bayesian non-negative Matrix Factorization (BNMF) is used to handle the issue of data sparsity in CF technique [6]. Deep learning based collaborative filtering, called the Deep Matrix Factorization (DMF)is used to integrate all kinds of side information effectively thereby increasing the model training efficiency [7].

Calibrated recommendations can prevent lesser interests of users from getting crowded out by his main interests. The output of the recommender system is post processed using a re-ranking algorithm [8].

CF methods improve the accuracy of recommender systems. But it suffers from data sparsityissues which are handled by using Singular Value Decomposition (SVD) method [9]. To improve the quality of prediction in SVD based systems, imputed data is created and incorporated into the SVD framework [10].

In this paper, we addressed the issue of data sparsity which makes the quality of SVD based recommendation poor. Our study is summarized as follows.

- Initially the rating matrix is created using the ratings dataset in MovieLens dataset.
- Similar users and items are found by computing the similarity.
- After computing the similarity, effective neighbors are identified for both the users and the items.
- Data creation is done based on the co-rated values computed using effective neighbors.
- The data thus created is incorporated into the SVD framework which is then used to predict the rating.

The following parts are organized as follows. We propose Rating matrix, Similarity computation, Neighbor selection and data creation in section II.

The SVD method is explained in section III. In section IV, we provide the experimental results carried out on the MovieLens 100k dataset. Conclusions are given in section V.

II. PROPOSED METHOD

A. *Proposed Approch*

CF recommender systems provide recommendations based on the user's interest over an item. The Rating provided by the user indicates his interests over the product. Ratings range from 1 to 5. Few items remain unrated which becomes difficult to handle in CF recommender systems. SVD is one of the most successful among MF techniques. But it still suffers from data sparsity.

To overcome sparsity, new data is created and imputed into the SVD framework. The fig (1) represents the workflow of our system

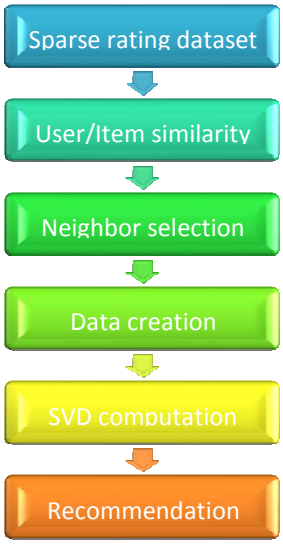

Fig.1. The flowchart of SVD

.

B. *Rating Matrix*

Creation of Rating matrix is done by using MovieLens 100k dataset which contains 671 users and 9066 movies. Ratings, users and items are taken from the dataset and converted into two dimensional Rating matrix which contains m users and n items and ratings given by each user on the items they prefer.

C. *Calculation of Similarity*

Vector Space Similarity (VSS) or Pearson Correlation Coefficient (PCC) typically measures the closeness between users. PCC is more powerful than VSS, because the

formertakes into consideration the rating fashion of each customer. Thus in this paper we use PCC to measure the user-to-user similarities:

$$Sim(a,i) = \frac{\sum_{j\in I(a)\cap I(i)}(r_{aj} - \bar{r}_a).(r_{ij} - \bar{r}_i)}{\sqrt{\sum_{j\in I(a)\cap I(i)}(r_{aj} - \bar{r}_a)^2}\sqrt{\sum_{j\in I(a)\cap I(i)}(r_{ij} - \bar{r}_i)^2}} \quad (1)$$

where similarity between user a and user i are denoted by Sim(a, i),The item sets rated by user a and user i are represented as I(a) and I(i), The average ratings rated by users a and i are represented as r_a and r_i. PCC is used to calculate similarity which ranges between [-1, 1], so that mapping function f (x) = (x + 1)/2 is used to merge the similarity range within [0, 1]. Even items also have similarity relation like users. The item j's neighbors are belong to the same class of item j and the probabilities of and the similarities represent the neighbors who belong to the same class. The similarity among items j and k can be determined as.

$$Sim(k,j) = \frac{\sum_{j\in U(k)\cap U(j)}(r_{ik} - \bar{r}_k).(r_{ij} - \bar{r}_j)}{\sqrt{\sum_{i\in U(k)\cap U(j)}(r_{ik} - \bar{r}_k)^2}\sqrt{\sum_{i\in U(k)\cap U(j)}(r_{ij} - \bar{r}_j)^2}} \quad (2)$$

The sets of users who gave ratings to items k and j represented as U(k) and U(j), The average rating on items k and j represented as $r_k$and r_j. Likely, a mapping function f (x) = (x + 1)/2 is used to merge the similarity range within [0,1].

D. Selection of Neighbors

After computing similarity for both users and items effective Neighbors are selected. The effective neighbor is selected by creating CO-RATED matrix which is created by following steps.

- Initially all users are compared with each other and if they have rated the same item , the ratings provided by each user for that item is obtained.
- After getting co-rated items value, the value is counted
- Finally all the co-rated count values are entered into a CSV file

E. New Data Creation

To reduce data sparsity in CF methods New data creation methods are developed for collaborative filtering. Which perform preprocessing to handle the missing values in rating matrix for the active user before estimating the ratings. In this paper the New data creation is done based on similar users and their co-rated values by taking sum of co-rated values and dividing it by the sum of similar users in the co-rated matrix. After getting the new value it is updated in the sparse rating matrix.

$$\hat{r}_u(i,j) = \frac{\sum_{a\in F(i)} r_{aj} \times Sim(a,i)}{\sum_{a\in F(i), r_{aj}\neq 0} Sim(a,i)} \quad (3)$$

For the missing rating $\hat{r}_{u,v}(i,j)$, The calculation based on user and item denoted as u and the New rating is the average value of the ratings on item j by the neighbors of user i. The number of i's neighbors is represented as |F(i)|.

III. SINGULAR VALUE DECOMPOSITION

SVD is utilized as a collaborative filtering (CF) algorithm within the setting of recommendation frameworks. Most CF calculations are based on user-item rating matrix where each row represents a user, each column represents an item. These matrix sections are ratings which are given to objects by users.

SVD is a method of matrix factorization that is commonly used to decrease the number of data set features by decreasing space dimensions from N to K where K < N is used. The matrix factorization element which keeps the same dimensionality for the purposes of the recommendation systems. The factorization of the matrix is performed on the basis of the user-item scores.

From a high level, one can think of matrix factorization as finding 2 matrices whose product is the original matrix.

$$R \approx (U) * (V_T) \quad (4)$$

Where R is the original rating matrix which is decomposes into two low dimensional matrices such as U and V.Their product is approximately equal to the original rating matrix.

Algorithm 1 Generating New ratings based on user similarity

Input: Rating matrix R

Output: New data rating matrix $\hat{R}$

1: Calculating the similarity between users and items according to Eq.(1,2).

2: Choosing neighbors for each user according to co-rated values.

3: Generating New ratings for missing ratings according to Eq.(3)

4: Return new data rating matrix $\hat{R}$.

The above algorithm represents the generation of new data which is to be filled with sparse rating matrix. Once the filling process is done the SVD is performed.

IV. EXPERIMENTS AND DISCUSSION

In this section, various experiments are done in order to prove our method is effective. Every experimentis done using Python.

A. Dataset

To test the impact of our process, experiments are implemented on MovieLens 100 K datasets, MovieLens contains 0.5-5.0 range of ratings rated by users in dataset of different sizes. The ranking period is from 1995 to 2016. This offers separate scores on movies ranging from 0.5 to 5.0 for 671 users on 9066 objects for the MovieLens 100 K dataset.

TABLE I
DATASET INFORMATION

Characteristic	User	Item	Rating
MovieLens 100k	671	9066	100004

B. Evaluation of Accuracy

To measure the prediction accuracy in CF methods are widely use Mean Absolute Error (MAE) and the Root Mean Square Error (RMSE). They are stated as formula (5) and formula (6)

$$MAE = \frac{\sum_{i,j \in T} |r_{ij} - \hat{r}_{ij}|}{|T|} \quad (5)$$

$$RMSE = \sqrt{\frac{\sum_{i,j \in T} (r_{ij} - \hat{r}_{ij})^2}{|T|}} \quad (6)$$

Where r_{ui} is the actual user rating u for item i, r_{ui} is the expected user rating u for item i and T is the number of predicted values.

The average difference between the item's true rating in the test set and the expected rating is known as MAE. The smaller the size, the greater will be the accuracy of the prediction.The square root of the ratio of the square of the deviation of the predicted value from the original value to the number of observations is known as RMSE. For large parts of the prediction RMSE needs high requirements on the stable of the experimental method.

C. ACCURACY ANALYSIS

For accuracy analysis our dataset is split into 5 fold of ratings. Which contains user base ratings and user test ratings that are taken as input for RMSE and MAE evaluation. Experimental results of Table II show that our method of adding new data with SVD significantly outperforms the existing SVD method which has no new data.

TABLE II

DATA SPARSENESS EXPERIMENTAL RESULTS

METHOD	RMSE	MAE
SVD without New data(Existing System)	Fold 1:1.1355 Fold 2:1.1257 Fold 3:1.0116 Fold 4:0.9646 Fold 5:0.9518	0.9490 0.9447 0.8093 0.7572 0.7466
SVD with New data	Fold 1: 0.9406 Fold 2: 0.9338 Fold 3: 0.9399 Fold 4: 0.9419 Fold 5: 0.9395	0.7408 0.7378 0.7397 0.7426 0.7417

From Table II it is clear that without adding additional data into SVD framework performs low when compared to our method of adding new data with SVD framework is proved.

The below graph represents the RMSE and MAE for our method whereas x axis has values 10 to 50 represents the number of iterations and y axis represents the range of predicted accuracy for our method. The blue line represents the RMSE value which was obtained while evaluation and red line represents the MAE for our method.

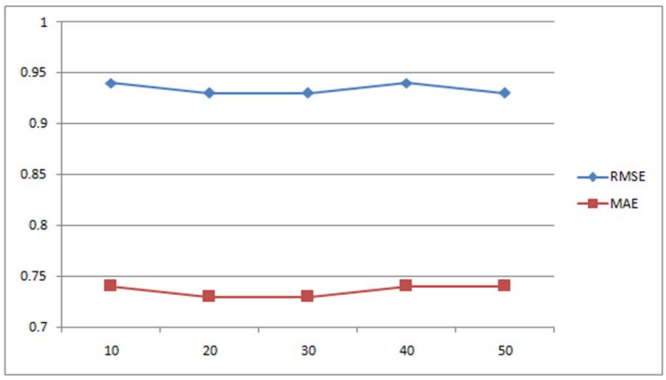

Fig.2.Performance analysis using RMSE and MAE

D. Prediction

Finally, we are interested in the expected rating for a certain combination of user-movie, so we can know if the user will like this movie or not. We'll first train the entire data set without splitting the data to get the best possible results.

```
t = data.build_full_trainset()
algo.train(t)
```

The above code is used to retrieve the trainset. Afterwards to predict a rating we give the user ID, item ID and the actual rating as follows.

```
uid = str(196)
id = str(302)
actual_rating = 4print algo.predict(uid, 302, 4)
```

The above code represents the user id 196 and the item id 302 and the rating given to that item by user 196 is 4.From the results we can see that the expected rating is 4.18 compared to the actual rating of 4.

```
user: 196        item: 302        r_ui = 4.00   est = 4.18
{u'was_impossible': False}
```

Therefore, the output shows that the user 196 will be rate for item 302 as 4.18. As the actual rating given to item302 by user 196 is 4. Thus predicting the rating for an item is approximately equal to the actual rating is proved.

V. CONCLUSION

Over the last decade, recommender systems made significant growth in numerous content-based, collaborative, and hybrid methods which have been proposed and developed. The method of adding new data to increase the quality of SVD based recommendation alleviate the data sparsity and also improves the recommendation. Thus this method is incorporated and the improvement of the existingsystem is proved. In future imputing data methods can be extended to several other matrix factorization techniques and also more knowledge can be explored for thecreation of new data and developing more accuratetechniques for measuring similarity among users or items.

VI. REFERENCE

[1] Al-Ghossein, M., Murena, P. A., Abdessalem, T., Barré, A., &Cornuéjols, A. (2018). Adaptive collaborative topic modeling for online recommendation. In *Proceedings of the 12th ACM Conference on Recommender Systems* pp. 338-346.ACM.

[2] Zhao, Q., Chen, J., Chen, M., Jain, S., Beutel, A., Belletti, F., & Chi, E. H. (2018). Categorical- attributes-based item classification for recommender systems. In *Proceedings of the 12th ACM Conference on Recommender Systems* pp. 320-328.ACM.

[3] Steck, H. (2018). Calibrated recommendations. In *Proceedings of the 12th ACM Conference on Recommender Systems* pp. 154-162.ACM.

[4] Gurini, D. F., Gasparetti, F., Micarelli, A., &Sansonetti, G. (2018). Temporal people-to- people recommendation on social networks with sentiment-based matrix factorization. *Future Generation Computer Systems*,Vol(*78)*, pp.430- 439

[5] Li,H.,Li,K.,An,J.,&Li,K.(2018).MSGD:a novel matrix factorization approach for large- scale collaborative filtering recommender systems on GPUs. *IEEE Transactions on Parallel and Distributed Systems*, *29*(7), pp.1530-1544.

[6] Wu, H., Zhang, Z., Yue, K., Zhang, B., He, J., & Sun, L. (2018). Dual-regularized matrix factorization with deep neural networks for recommender systems. *Knowledge-Based Systems*,Vol(*145)*,pp.46-58

[7] Bobadilla, J., Bojorque, R., Esteban, A. H., &Hurtado, R. (2018). Recommender systems clustering using Bayesian non negative matrix factorization. *IEEEAccess*,Vol(*6)*,pp.3549-3564.

[8] Cui, L., Huang, W., Yan, Q., Yu, F.R., Wen, Z. and Lu, N., 2018. A novel context-aware recommendation algorithm with two-level SVD in social networks. *Future Generation Computer Systems*, Vol(*86)*, pp.1459-1470

[9] Yi, B., Shen, X., Liu, H., Zhang, Z., Zhang, W., Liu, S., &Xiong, N. (2019). Deep Matrix Factorization with Implicit Feedback Embedding for Recommendation System. *IEEE Transactions on Industrial Informatics*.

[10] Yuan, X., Han, L., Qian, S., Xu, G., & Yan, H. (2019). Singular value decomposition based recommendation using imputed data. *Knowledge-BasedSystems*, Vol(*163)*,pp.485-494.