Recommender System : An Automated Movie Rating Prediction Using an Improved Timefly Algorithm (ITFA)

Arun Kumar R[1], Deepika R[2], Sathesh Kumar K[3], Dinesh P S[4]

[1]Assistant Professor, Bannari Amman Institute of Technology, Erode – 638 401, India,
itsark1987@gmail.com

[2]Assistant Professor, Bannari Amman Institute of Technology, Erode – 638 401, India,
deepikar@bitsathy.ac.in

[3]Assistant Professor, National Engineering College, Kovilpatti, India,
satheshkumar-cse@nec.edu.in

[4]Assistant Professor, Bannari Amman Institute of Technology, Erode – 638401, India,
dineshpudhu@gmail.com

ABSTRACT

A Recommender System refers to the system which predicts the user preferences in future based on the user's rating or review. The automated movie rating prediction helps the user to identify the rating of a movie based on the user preferences. This paper proposes a recommender system which applies an improved TimeFly Algorithm (iTFA) and works based on the changes in user behavior over the time and thus the proposed methodology resolves the problem in fluctuation of user's preferences with respect to the time. MovieLens Dataset 100K, 1M, 10M and 20M is used to implement the proposed model. Two similarity measures Cosine similarity measures and Jaccard similarity is used for classification and in order to measure the error rate RMSE and MAE are applied. The results obtained are compared with the classical model and shows that proposed work improves the accuracy in a larger way.

Keywords: Recommendation, Sparsity, Similarity, Time-variant

1. INTRODUCTION

Recommender systems play a major role in suggesting the products/ features to users based on the rating given by other users in the past. And nowadays it is most widely used in the world primary e-commerce websites like Alibaba, eBay, Walmart, Amazon etc. Classical methods such as content - based filtering [2], collaborative filtering [1], and hybrid filtering [3] are the keen techniques to be notable in the field of Recommender systems. Even though these methods handle the huge volume of information generated over the internet and helps the user in predicting the movie rating, these techniques have some assumptions on user behavior is static.

These methods work on the hypothesis that the user's behaviour in a particular item never changes as the time goes. Since we could see that this cannot be applicable in all cases, we are in need to address this issue.

In our proposal, the above issue is resolved by categorizing the items based on the various ranges of timestamp. We also tried to maximize the accuracy by increasing the number of cycles and by using two similarity measures instead of one metric that was adopted in TimeFly Algorithm [10].

The entire work is structured as below: Section 2 summarizes the similar works in the domain of recommender systems. The third section briefs the proposed model. The fourth and fifth section highlighted the experimental setup and final results respectively. Finally, we concluded our research work with a hint for future work in the last section.

2. RELATED WORK

PMMF [2] introduced a novel scheme known as maximum margin factorization scheme with the indent to resolve the over fitting problem for the discrete valued matrix. It also helps in minimizing the overall processing time by introducing the factorization approach.

In order to approximate the matrix, singular value decomposition (SVD) method [3] can be used. In addition to this, matrix factorization is fused [4] with SVD to bring out the dimension in the rating. The major merit of using SVD is that it eliminates the over fitting issue when we are dealing with sparse data. Another method defined as RMF [5] technique is also applied to address the over fitting.

We can also reduce the dimensionality of the dataset by combining similar items to the identical cluster and then diverting dissimilar items to the different clusters. This technique [6] is known as Self-constructing algorithm (SCC).
We have certain drawbacks in the above discussed methods while minimizing dimensionality with the motto of reducing time complexity. Some other systems [7,8 and 9] require to know the numbers of clusters prior to proceeding the classification.

The user behavior changes with respect to the time and this also needs to take utmost care while rating a movie based on the user's interest. TimeFly algorithm [10] categorizes the group of user items based on the different range of timestamp values and improves the accuracy.

We propose an improved TimeFly algorithm (iTFA) in order to address the above mentioned problems and also to improve the prediction accuracy in case of handling sparse dataset.

3. PROPOSED METHODOLOGY

In most of the recommendation techniques, there exists a lagging during the insufficient data and high computational operations. In order to address the above mentioned limitations, we proposed an improved TimeFly algorithm (iTFA) to develop a updated recommendation system.

3.1 Stages in proposed model:

There are five stages in the proposed model and consists of transforming, randomly selecting users, group generation to train the data, prediction data and determining the model accuracy at last.

3.1.1 Data Transformation:

Here, we generated a sequence of data which comprises movie ratings, timestamp, and also genre information.

3.1.2   User selection:

We randomly selected users on which we perform training and testing our model. The movie dataset (MovieLens [11]) like 10K, 1M, 10M and 20M is used for both training and testing.

3.1.3   Group generation:

For instance, consider an user 'u' which is randomly selected during step 2. We arrange all the items which are being rated by the user 'u' with respect to the various timestamp value. Then we classify the arranged timestamp range of values into five cycles namely CYCLE I, II, III , IV and V and the items lies in these five individual range of timestamp will assist us in determining the group category.

3.1.4 Predicting test data:

The actual prediction is done at this step and we use two different types of similarity measures. First one is cosine similarity measure which can be calculated as follows,

$$\frac{\sum_{i=1}^{n} A_i \, X \, B_i}{\sqrt{\sum_{i=1}^{n} A_i^2} \; X \; \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{1}$$

Secondly, Jaccard similarity measure is used and defined as follows:

$$J(A,B) = \frac{|A \cap B|}{|A \cup B|} = \frac{|A \cap B|}{|A| + |B| - |A \cap B|} \tag{2}$$

Consider a test data k (belongs to the user 'u') whose values goes upto the total number of users present in the cycle, we compute the term c_i, i= 1 to 10 as follows:

c_1, c_3, c_5, c_7 and c_9 are the cosine similarity measure of item k with CYCLE I, CYCLE II, CYCLE III, CYCLE IV and CYCLE V respectively. c_2, c_4, c_6, c_8 and c_{10} are the Jaccard similarity measure of item k with CYCLE I, CYCLE II, CYCLE III, CYCLE IV and CYCLE V respectively.

After the computation of above similarity measures, we take intersection between the sets $(c_1, c_3, c_5, c_7, c_9)$ and $(c_2, c_4, c_6, c_8, c_{10})$ to identify the group category. The above process is repeated for all the remaining test data items for the u particular user 'u' and averaged them to predict the final rating.

3.2 Algorithm:

Input : User – item raing matrix along with the timestamp values and genre details.

   N – number of users ' u'

   n – number of training dataset

   m – number of test data

Output : List of Recommendation

#Process of selecting users (N)

Choose 'N' number of users randomly from the training data.

#Categorizing the group and trend analysis

for u = 1 : N

  Classify the matrix (movie_id, rating) into training data (90%) and test data (10%) for user 'u'.

  for i = 1 : n

    Arrange the training data set based on the timestamp values

    Classify the arranged itemset into five different CYCLES (I, II, III, IV and V)

    for k = 1: m

      c_1 = Cosine similarity measure for the item k w.r.to CYCLE I

      c_2 = Jaccard similarity measure for the item k w.r.to CYCLE I

      c_3 = Cosine similarity measure for the item k w.r.to CYCLE II

      c_4 = Jaccard similarity measure for the item k w.r.to CYCLE II

      c_5 = Cosine similarity measure for the item k w.r.to CYCLE III

      c_6 = Jaccard similarity measure for the item k w.r.to CYCLE III

      c_7 = Cosine similarity measure for the item k w.r.to CYCLE IV

      c_8 = Jaccard similarity measure for the item k w.r.to CYCLE IV

      c_9 = Cosine similarity measure for the item k w.r.to CYCLE V

      c_{10} = Jaccard similarity measure for the item k w.r.to CYCLE V

group_category(k) = max(c1,c3,c5,c7,c9) intersect max(c2,c4,c6,c8,c10)
rating(k) = avg_trend(group_category(k))
#Error Evaluation of an user 'u'
   Calculate RMSE for u
   Calculate MAE for u
#Performance Evaluation for all users on the test data
Calculate average Root Mean Square Error for 'N' users.
Calculate average Mean Absolute Error for 'N' users

3.3 Model Accuracy Evaluation:

We use two accuracy parameters to evaluate the model namely RMSE - Root Mean Square Error and MAE - Mean Absolute Error. These two values RMSE and MAE can be calculated using the following formula respectively:

$$RMSE = \sqrt{\frac{\sum_{i=1}^{n}(Predicted_i - Actual_i)^2}{N}} \tag{3}$$

$$MAE = \frac{1}{n}\sum_{j=1}^{n}\left|Predicted_j - Actual_j\right| \tag{4}$$

4. EXPERIMENTAL DISCUSSION

The proposed model is developed using the Python language and open source notebook Jupyter platform is used. The model tested over the datasets which are in varying sizes and sparsity levels. The dataset used is MovieLens [11] which has four categories namely 100K, 1M, 10M and 20M.

5. RESULTS

In order to measure the error rate of the proposed algorithm, two metrics namely Root Mean Square Error (RME) and Mean Absolute Error (MAE) values are used. The results of the improved TimeFly algorithm (iTFA) and the classical TimeFly algorithm(TFA) are compared and described below:

Table. 1: Comparison of TFA and ITFA algorithm with respect to the RMSE and MAE values

Dataset	Classical TimeFly algorithm (TFA)		Improved TimeFly algorithm (iTFA)	
	RMSE	*MAE*	*RMSE*	*MAE*
100K	0.712	0.594	0.654	0.517
1M	0.739	0.606	0.643	0.508
10M	0.711	0.473	0.589	0.327
20M	0.701	0.508	0.541	0.384

5. CONCLUSION AND FUTURE WORK

The proposed work of improved TimeFly algorithm (iTFA) improves the accuracy in a greater manner and which can be seen in the above table (Table. 1). It improves the prediction rate by concerning user behavior changes with respect to the time. The algorithm employs two similarity measure – Cosine similarity and Jaccard similarity to improve the classification. In future, same algorithm and model can be deployed to recommend the items in e-commerce websites and also we can use other similarity measure metrics to improve the classification and hence it improves the prediction.

REFERENCES

1. Behzad Soleimani Neysiani, Nasim Soltani, Reza Mofidi and Mohammad Hossein Nadimi-Shahraki. **Improve performance of association rule-based collaborative filtering recommendation systems using genetic algorithm**, *I.J. Information Technology and Computer Science*, Vol. 2, pp. 48-55, 2019.
2. Vikas Kumar, Arun K.Pujari, Sandeep Kumar Sahu, Venkateswara Rao Kagita and Vineet Padmanabhan. **Proximal maximum margin matrix factorization for collaborative filtering**, *Pattern Recognition Letters*, Vol. 86, January 2017.
3. Sarwar B, Karypis G, Konstan J and Riedl J. **Incremental singular value decomposition algorithms for highly scalable recommender systems**, *Fifth International Conference On Computer And Information Science*, 2002
4. Jiang S, Ding Z and Fu Y (2019). **Heterogeneous recommendation via deep low-rank sparse collective factorization** , *IEEE Trans Pattern Anal Mach Intell*, 2019.
5. Koren Y, Bell R and Volinsky C. **Matrix factorization techniques for recommender systems** , *Computer*, 2009.
6. Liao CL and Lee SJ. **A clustering based approach to improving the efficiency of collaborative filtering recommendation**, *Electron Commer Res Appl*, 2016
7. Sarwar BM, Karypis G, Konstan J and Riedl J. **Recommender systems for large-scale e-commerce: Scalable neighborhood formation using clustering**, *5th International Conference On Computer And Information Technology*, 2002.
8. Xue G-R, Lin , Yang Q, Xi W, Zeng H-J, Yu Y and Chen Z. **Scalable collaborative filtering using cluster-based smoothing**, *ACM SIGIR conference*, 2005
9. Cai Y, Leung H, Li Q, Min H, Tang J and Li J. **Typicality-based collaborative filtering recommendation**, *IEEE Trans Knowl Data Eng*, 2014.
10. Bam Bahadur Sinha1, R. Dhanalakshmi and Ramchandra Regmi. **TimeFly algorithm: a novel behavior‑ inspired movie recommendation paradigm** , *Pattern Analysis and Applications*, April, 2020.
11. Dataset: https://grouplens.org/datasets/movielens/