Energy-based Collaborative Filtering Recommendation

Tu Cam Thi Tran[1, [0000-0001-5811-6952]], Lan Phuong Phan[2], Hiep Xuan Huynh[3]*[, [0000-0002-9213-131X]]
Faculty of Information Technology, Vinh Long University of Technology Education (VLUTE), Vinh Long province, Vietnam[1]
College of Information and Communication Technology, Can Tho University (CTU), Can Tho city, Vietnam[2, 3]

***Abstract*—The core value of the recommendation model is the using of the measures to measure the difference between the jumps (e.g. pearson), some other studies based on the magnitude of the angle in space (e.g. cosine), or some other studies study the level of confusion (e.g. entropy) between users and users, between items and items. Recommendation model provides an important feature of suggesting the suitable items to user in common operations. However, the classical recommendation models are only concerned with linear problems, currently there is no research about nonlinear problems on the basis of potential/energy approach to apply for the recommendation model. In this work, we mainly focus on applying the energy distance measure according to the potential difference with the recommendation model to create a separate path for the recommendation problem. The theoretical properties of the energy distance and the incompatibility matrix are presented in this article. Two experiment scenarios are conducted on Jester5k, and Movielens datasets. The experiment result shows the feasibility of the energy distance measures/ the potential in the recommendation systems.**

Keywords—Energy distance; energy model; collaborative filtering; recommendation system; distance correlation; incompatibility

I. Introduction

Recommendation systems suggest the suitable items to a user based on his/her purchased items or his/her rated items [9]. There are many implementations of a recommendation system based on different factors and applied to different contexts, such as the recommendation systems determining the user's rating values according to the magnitude of the angle (e.g. cosine) [22], or the recommendation systems based on the difference of the users (e.g. pearson) [19][2], or the recommendation systems based on the confuse of one user with another (e.g. entropy) [4], some other recommendation systems based on the statistical implication [12][18].

Collaborative filtering recommendation model [6] mainly based on users, items. In particular, the Singular Value Decomposition algorithm - a classical method from linear algebra used as a technique to reduce size in machine learning - is combined with recommendation model, or Alternating Least Squares (ALS) - a matrix factorization algorithm – is used for the larger-scale collaborative filtering problems, or some techniques for selecting random or popular items are also integrated to recommendation systems. However, most of the recommendation models revolve around the problems of linear relations, not the problems of nonlinear relations.

In this article, we propose a new collaborative filtering recommendation model to consider nonlinear relations instead of focusing only on linear relations between users. This approach is performed on the basis of determining the relationship/distance between users in pairs, especially Newton's gravitational potential energy (known as potential energy, shortly energy) between two users. In this collaborative filtering recommendation model, the relationship between two users is determined through calculating the maximum mean discrepancy (MMD), or a lack of compatibility or similarity between two or more users.

The article is structured as follows. In Section II, we present collaborative filtering based on energy. Section III presents the learning model, data division methods and evaluation methods. In Section IV, we propose the new recommendation model based on energy. In Section V, we show the experiment on the Jester5k, and MovieLense datasets. Section VI is the conclusion of the article.

II. Collaborative Filtering

Collaborative filtering [3][15][22] is the process of filtering or evaluating items using the opinions of others. Collaborative filtering technology gathers the opinions of large interconnected communities on the webs, and supports filtering of substantial quantities of data. The recommendation system [1] uses a lot of information such as: the items, the users and the rating values to suggest the suitable items to user. However, the unwanted information has been removed by using the computerized methods before presenting the recommendation result to the user.

In collaborative filtering [1], the recommendation system searches for similar users to make predictions. The user's rating model is a useful feature for determining similarity. Normally, the collaborative filtering recommendation methods use ratings without additional information about the user or the item to recommend the suitable items.

The recommendation system [9][10][11] is an information system [17], includes a set of four:

$$S = < U, I, R, f > \quad (1)$$

Where

U - is the set of users (the closed universe), $U = \{u_1, u_2, \ldots, u_n\}$ with $u_k \in U, k = 1..n$. U is a finite set of n objects (a nonempty set)

*Corresponding Author.

I - is the set of items $I = \{i_1, i_2, \ldots, i_m\}$ with $i_j \in I, j = 1..m$. I is a finite set of m attributes (a nonempty set).

$R = \{r_{ij}\}$, with $i = 1..n$, $j = 1..m$. R is a rating matrix, where r_{ij} is rating value of the user u_i to item i_j.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} & .. & r_{1m} \\ r_{21} & r_{22} & r_{23} & .. & r_{2m} \\ r_{31} & r_{32} & r_{33} & .. & r_{3m} \\ .. & .. & .. & .. & .. \\ r_{n1} & r_{n2} & r_{n3} & .. & r_{nm} \end{bmatrix}$$

For example, Table I is a rating matrix where the rating value ranges from 1 to 5 or not available.

$f : U \, x \, I \rightarrow$ R - is the total decision function called the information function such that $f(u, i) \in R_i$ for every $i \in I$, $u \in U$. Function $f(u_k, i_j)$ is used to measure the relevance (the rating value) of item i_j with user u_k. The rating value $f : U \, x \, I \rightarrow R$.

A. Energy Distance

The energy distance [7][8][13] is the distance between the probability distributions. Energy is defined as the similarity in the form of potential energy between objects in gravitational space. The potential energy is zero if and only if the positions (centers of gravity) of the two objects coincide, and the potential energy increases as the difference between the objects in space increases. The concept of potential energy can be applied to collaborative filtering. Let U_1 and U_2 be independent random vectors in U , where F and G are cumulative distribution functions, and they correspond to each other. Accordingly, $\|.\|$ represents the Euclidean normal of its argument, E represents the expected value, and a random variable U_1' represents a copy (iid), which is independent and distributed like U_1; that mean, U_1 and U_1' are iid. Similarly, U_2 and U_2' are iid. The squared energy distance [16][20][24] can be determined according to the expected distance between random vectors.

$$D^2(F, G) := 2E\|U_1 - U_2\| - E\|U_1 - U_1'\| - E\,\|U_2 - U_2'\| \geq 0 \tag{2}$$

Consider the null hypothesis that two random variables, U_1 and U_2, have the same cumulative distribution functions: F = G. For samples $u_{11}, \ldots, u_{1n}$ from U_1 and $u_{21}, \ldots, u_{2m}$ from U_2, respectively, the E-statistic for testing this null hypothesis is:

$$\varepsilon_{n,m}(U_1, U_2) := 2A - B - C \tag{3}$$

where A, B, and C are simply averages of pairwise distances:

$$A = \frac{1}{nm} \sum_{i=1}^{n} \sum_{j=1}^{m} \left\|u_{1i} - u_{2j}\right\|, \tag{4}$$

$$B = \frac{1}{n^2} \sum_{i=1}^{n} \sum_{j=1}^{n} \left\|u_{1i} - u_{1j}\right\|, \tag{5}$$

$$C = \frac{1}{m^2} \sum_{i=1}^{m} \sum_{j=1}^{m} \left\|u_{2i} - u_{2j}\right\|: \tag{6}$$

One can prove (3) that $\varepsilon(U_1, U_2) := D^2(F, G)$ is zero if and only if U_1 and U_2 have the same distribution ($F = G$). It is also true that the statistic $\varepsilon_{n,m}$ is always non-negative. When the null hypothesis of equal distributions is true, the test statistic.

$$T = \frac{nm}{n+m} \varepsilon_{n,m}(U_1, U_2) \tag{7}$$

B. Incompatibility Matrix

The incompatibility matrix E represents the energy distance between users. The incompatibility matrix $E = \{e_{ij}\}$, with $i = 1..n$, $j = 1..n$, e_{ij} is calculated by formula (2).

$$E = \begin{bmatrix} e_{11} & e_{12} & e_{13} & .. & e_{1n} \\ e_{21} & e_{22} & e_{23} & .. & e_{2n} \\ e_{31} & e_{32} & e_{33} & .. & e_{3n} \\ .. & .. & .. & .. & .. \\ e_{n1} & e_{n2} & e_{n3} & .. & e_{nn} \end{bmatrix}$$

For example, Table II shows the matrix representing the energy distance between users by using information of Table I.

C. Incompatibility Neighborhood

The neighborhood of the user u_a is defined by the energy distance between the users and u_a . The neighborhood is filtered with a certain number of the users, who has the lowest potential energy (i.e. k nearest neighbors - knn).

TABLE I. AN EXAMPLE OF THE RATING MATRIX R

	i_1	i_2	i_3	i_4	i_5	i_6	i_7	i_8
u_1	?	4.0	4.0	2.0	1.0	2.0	?	?
u_2	3.0	?	?	?	5.0	1.0	?	?
u_3	3.0	?	?	3.0	2.0	2.0	?	3.0
u_4	4.0	?	?	2.0	1.0	1.0	2.0	4.0
u_5	1.0	1.0	?	?	?	?	?	1.0
u_6	?	1.0	?	?	1.0	1.0	?	1.0
u_7	1	3.0	2.0	?	?	2.0	?	?
u_8	5	?	?	2.0	1.0	?	?	?
u_9	?	4.0	?	?	1.0	2.0	?	?

TABLE II. AN EXAMPLE OF THE MATRIX OF ENERGY DISTANCE FOR THE ACTIVE USERS

	u_1	u_2	u_3	u_4	u_5	u_6	u_7	u_8	u_9
u_1	0	0	0	0	0	0	0	0	0
u_2	7.874	0	0	0	0	0	0	0	0
u_3	7.280	5.196	0	0	0	0	0	0	0
u_4	8.306	6.403	2.828	0	0	0	0	0	0
u_5	6.000	5.656	4.795	5.385	0	0	0	0	0
u_6	5.567	5.196	4.898	5.830	1.732	0	0	0	0
u_7	3.316	6.557	6.324	6.928	3.605	3.464	0	0	0
u_8	7.810	5.000	4.000	4.690	4.795	5.656	6.164	0	0
u_9	4.472	6.480	6.708	7.549	4.000	3.316	2.645	7.000	0

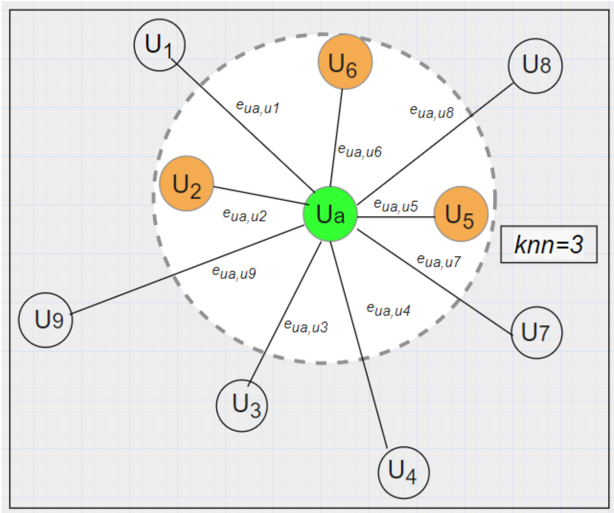

Fig. 1. The Neighborhood of u_a with knn = 3.

To find the k-nearest neighbors (knn) for u_a, the energy distance is used. Fig. 1 shows the 2D space of the incompatibility points with the active user u_a - the users with low energy will be displayed closer together. If knn equals to 3, u_2, u_5 and u_6 are selected to be three nearest neighbors of u_a.

D. Rating prediction

The predicted rating $\hat{r}_{aj}$ of user u_a for item i_j is calculated by (8),

$$\hat{r}_{aj} = \frac{1}{\sum_{i \in N(a)} e_{ai}} \sum_{i \in N(a)} e_{ai}\, r_{ij} \quad (8)$$

where: e_{ai} is the incompatibility between u_a and the user u_i in the neighborhood. $N(a)$ is knn of the user u_a. r_{ij} is the rating value of the user u_i to item i_j.

E. Top N Items Recommendation

To recommend the suitable items to the active user u, N items with the highest ranking are selected. The ranking of each item i is calculated by the ranking function. This function is reversible to map the predicted rating on the normalized scale back to the original rating scale. Normalization is used to remove individual rating bias by users who use lower or higher ratings than other users. A popular method is to center the rows of the rating matrix by formula:

$$h(r_{ui}) = \hat{r}_{ui} - \bar{r}_u \quad (9)$$

Where $\bar{r}_u$ is the average of all available ratings in row u (i.e. the available ratings of user u) of the rating matrix R; $\hat{r}_{ui}$ is the predicted rating of user u to item i.

III. RECOMMENDATION EVALUATION

A. K-folds Cross Evaluation

In order to evaluate the effectiveness of recommender models [5][25], k-folds cross evaluation method is performed. In this article, the dataset is divided into a training set and a testing set with k-folds = 5. The dataset is splitted into 5 subsets, all subsets of equal size, 80% (4 subsets) of the dataset is used for training and 20% (1 subset) of the dataset is used for testing. The model is evaluated recursively 5 times, each time using a different train/test split, which ensures that all users and items are considered for both training and testing. The results are then averaged to produce the final result.

B. Evaluation

To evaluate the recommendation model, three measures of error: 1. Mean Absolute Error (MAE); 2. Mean Squared Error (MSE); and 3. Root Mean Square Error (RMSE) is used. The evaluation of the error of the recommendation model is an important step in the design of the recommendation system. This helps the designer to select the model and they can check the error of the model before the designer applies this model in practice.

- Root Mean Square Error (RMSE) [14][23]. Root mean square error between real rating value r_{ij} and the predicted rating value $\hat{r}_{ij}$ is calculated by fomular (10).

$$RMSE = \sqrt{\frac{\sum_{(i,j)\in n}(r_{ij} - \hat{r}_{ij})^2}{|n|}} \quad (10)$$

- Mean Squared Error (MSE) [14][23]. The mean square error between real rating value r_{ij} and the predicted rating value $\hat{r}_{ij}$ is calculated by fomular (11).

$$MSE = \frac{\sum_{(i,j)\in n}(r_{ij} - \hat{r}_{ij})^2}{|n|} \quad (11)$$

- Mean Absolute Error (MAE) [14][23]. The mean absolute error between real rating value r_{ij} and the predicted rating value $\hat{r}_{ij}$ is calculated by fomular (12).

$$MAE = \frac{1}{|n|} \sum_{(i,j)\in n} |r_{ij} - \hat{r}_{ij}| \quad (12)$$

IV. ENERGY BASED RECOMMENDATION MODEL

A. Model

Fig. 2 presents the general overview of energy-based recommendation model including four components: the dataset (U x I x R), energy including incompatibility matrix which is calculated by the energy distance measure, predicted ratings performing the prediction, and the table of predicted rating to be used for recommending top n items to active user u_a.

The dataset consists of a set of m items (items), where $I = \{i_1, i_2, \dots, i_j, i_{j+1}, \dots, i_m\}$ and the set n users (users), where $U = \{u_1, u_2, u_k, u_{k+1} \dots, u_n\}$, the rating values $R = \{r_{11}, r_{12}, \dots, r_{1n}; r_{21}, r_{22}, \dots, r_{2n}; \dots; r_{m1}, r_{m2}, \dots, r_{mn}\}$, u_a is the active user to be recommended.

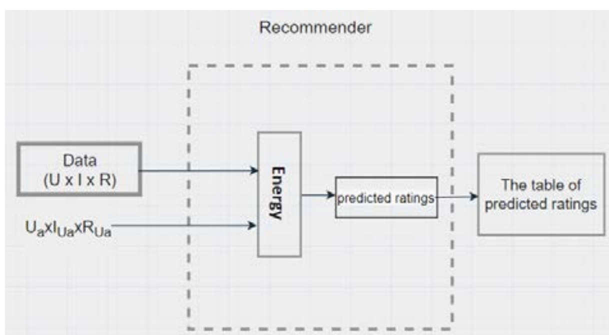

Fig. 2. Energy-based Recommendation Model.

B. Algorithm

The recommendation algorithm of the energy based model includes six steps as the folow:

Algorithm. Energy-based recommendation

Input: The data matrix (U x I x R); and the active user needs to be suggested u_a

Output: The rating prediction table to be used for recommendation to the active users u_a;

Begin

Step 1: Calculating the incompatibility matrix E by using the energy distance of u_a with all users

Step 2: Finding k nearest neighbors (int u, int i, int [][]R, int [][]E)

// u is the users, i is the items, rating is the rating matrix, E is the incompatibility matrix.

if (R[u][i] != 0 && E [u_a][u] != 0)

Step 3: Predicting the rating value of u_a for items based on k nearest neighbors.

<$R'[a][j] = (1/E(u_a, u))(E(u_a, u)\ R'[i][j])$>

Step 4: Calculating the ranking of each item <List_N[i]>;

Step 5: Sorting the list of predicted ratings in descending order < Sort (List_N)>;

Step 6: Recommeding the top *N* item with the highest ranking to the active u_a < Print (Top-N>);

End.

V. Experiment

A. Datasets

Experiment is performed on the Jester5k and MovieLense datasets. These two datasets are summarized in the Table III, and the distribution of ratings of them is displayed Fig. 3.

Jester5k[1] contains the ratings of 5000 anonymous users collected from the Jester Online Joke Recommendation System between April 1999 and May 2003. This data set contains 5000 users and 100 jokes with ratings ranging from -10.00 to +10.00. All selected users have rated 36 or more jokes.

MovieLense [2] (100k) were published in 1998 by GroupLense (https://grouplens.org). This dataset includes 100,000 (100k) ratings from 943 users for 1682 movies with ratings ranging from 1 to 5. Each user has rated at least 20 movies.

B. Tool

The "recommenderlab" package [21] is used in experiment of this article; specifically, user based collaborative filtering model using cosine measure (named UBCFCosine RS).

[1] https://rdrr.io/cran/recommenderlab/man/Jester5k.html, accessed on February 01, 2021.

[2] https://rdrr.io/cran/recommenderlab/man/MovieLense.html

TABLE III. THE TABLE TO DESCRIBE DATASETS: JESTER 5K AND MOVIELENSE

Names	Number of rows (users)	Number of col.s (items)	Number of ratings	Value domain of ratings
Jester5k	5000	100	362106	-10 - +10
MovieLense	943	1682	99392	1 - 5

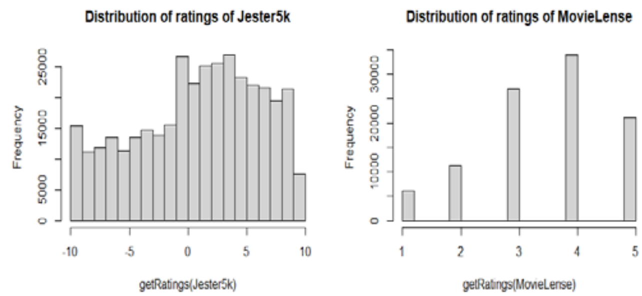

Fig. 3. Distribution of Ratings of Jester5k vad Movielense Datasets.

We have implemented the proposed energy based recommendation model (named UBCFEnergy RS) in R language. This model is integrated into the recommenderlab package. We have also written a function to compare the results of the proposed model UBCFEnergy RS and the selected model of recommenderlab package UBCFCosine RS.

C. Scenario 1: Recommendation on Jesterk5k

This scenario evaluates the errors (MAE, RMSE, MSE) of two recommendation models UBCFEnergy RS and UBCFCosine RS.

The comparison results of errors (MAE, MSE, RMSE) of the two models are shown in Fig. 4 for each known ratings (given = 2, 16, 36) on all k nearest neighbors knn = 10, 20, 30, 40. The results show that the error of the UBCFEnergy RS model is always smaller than that of UBCFCosine RS model.

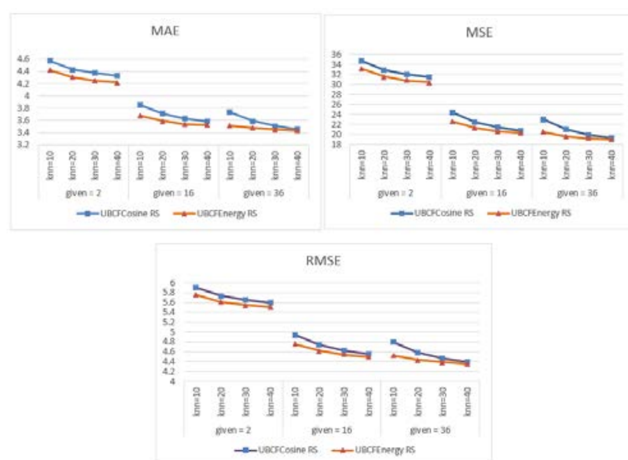

Fig. 4. Errors for each given (2, 16, 36) on all knn = 10, 20, 30, 40.

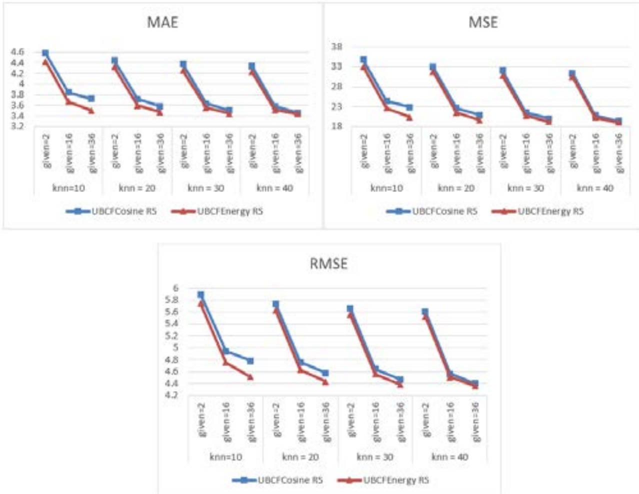

Fig. 5. Errors with each knn (10, 20, 30, 40) for all given = 2, 16, 36.

Fig. 5 presents the errors of UBCFEnergy RS and UBCFCosine RS for each k nearest neighbors knn = 10, 20, 30, 40 on all known ratings (given = 2, 16, 36). The experiment result also show that the proposed model is better than UBCFCosine RS model.

D. Scenario 2: Recommendation on MovieLense

This scenario presents the experement result of two models UBCFEnergy RS and UBCFCosine RS on MovieLense dataset.

Fig. 6 shows the comparison results of errors (MAE, MSE, RMSE) of the two models for each known ratings (given = 4, 10, 20) on all k nearest neighbors knn = 10, 20, 30, 40. Fig. 7 presents the errors of two models for each k nearest neighbors knn = 10, 20, 30, 40 on all known ratings (given = 4, 10, 20). Both results indicates that the errors of UBCFEnergy RS model are smaller than those of UBCFCosine RS model.

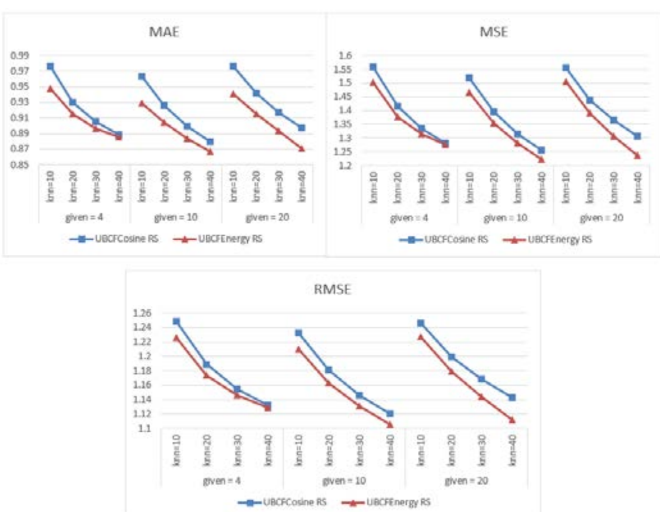

Fig. 6. Errors for each given (4, 10, 20) on all knn = 10, 20, 30, 40.

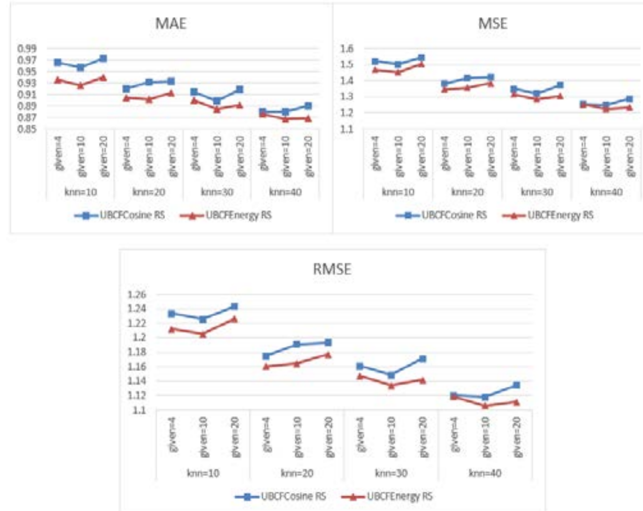

Fig. 7. Errors with each knn (10, 20, 30, 40) for all given = 4, 10, 20.

VI. Conclusion

We have built a new recommendation model based on energy UBCFEnergy RS. The errors (MAE, MSE, RMSE) of this model are compared with the error of user based collaborative filtering model using cosine measure of recommenderlab" package UBCFCosine RS on Jester5k and MovieLense datasets, two datasets commonly used in evaluating the effectiveness of recommendation models. The experimental results of the proposed recommendation model have the lower errors than the compared model on both Jester5k and Movielense. Therefore, the energy-based recommendation model shows the feasibility of applying the energy distance to build the recommendation systems.

References

[1] A.B. Suhaim, and J. Berri, “Context-Aware Recommender Systems for Social Networks: Review, Challenges and Opportunities,” in IEEE Access, vol 9, pp. 57440-57463, 2021.

[2] ABMK. Hossain, Z. Tasnim, S. Hoque, S. Hoque, and M.A. Rahman, “A Recommender System for Adaptive Examination Preparation using Pearson Correlation Collaborative Filtering,” Int J Auto AI Mach Learn, vol 21, pp. 30-43, 2021.

[3] B. Hong, and M. Yu, “A collaborative filtering algorithm based on correlation coefficient. Neural Computing and Applications, vol 31, pp. 8317–8326, 2019.

[4] C. Hemalatha, and B. Bharat, “Personalized recommender system using entropy based collaborative filtering technique,” Journal of Electronic Commerce Research, vol 12, 2011.

[5] C.C. Aggarwal, Recommender Systems, vol. 1. Springer, Heidelberg 2016.

[6] D. Jannach, P. Pu, F. Ricci, and M. Zanker, “Recommender Systems: Past, Present, Future,” AI Magazine, vol 42, 2021. pp. 3–6.

[7] D. Edelmann, T F. Móri, and G. J. Székely, On relationships between the Pearson and the distance correlation coefficients, Statistics & Probability Letters, Vol 169, 108960, ISSN 0167-7152, (2021).

[8] E. Martínez-Gómez, M.T. Richards, and DStP. Richards, “Distance correlation method for discovering associations in large astrophysical databases,” the Astrophysical Journal, American Astronomical Society, vol 781, 2014.

[9] G. Adomavicius, and A. Tuzhilin, "Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions," IEEE Transactions on Knowledge and Data Engineering, vol. 17, pp. 734–749, 2005.

[10] G. Adomavicius, R. Sankaranarayanan, S. Sen, and A. Tuzhilin, "Incorporating contextual information in recommender systems using a multidimensional approach," ACM Trans. Inf. Syst, Vol 23, pp. 103–145, 2005.

[11] G. Adomavicius, N. Manouselis, and Y. Kwon, Multi-criteria recommender systems, Recommender Systems Handbook, pp.769-803 2011.

[12] H.X. Huynh, Q.N. Phan, T.N. Duong, and T.T.H. Nguyen, "Collaborative Filtering Recommendation Based on Statistical Implicative Analysis," International Conference on Computational Collective Intelligence, Springer, Cham, pp. 224-235, 2020.

[13] H. Zhang, Y. Jian, and P. Zhou, "Collaborative Filtering Recommendation Algorithm Based on Class Correlation Distance," Recent Advances in Computer Science and Communications, vol 14, pp. 887–894, 2021.

[14] J.L. Herlocker, J.A. Konstan, L.G. Terveen, and J.T. Riedl, "Evaluating collaborative filtering recommender systems," ACM Trans. Inf. Syst, vol 22, pp. 5–53, 2004.

[15] J. Chen, C. Zhao, Uliji, and L. Chen, "Collaborative filtering recommendation algorithm based on user correlation and evolutionary clustering. Complex & Intelligent Systems. 6(1), pp. 147–156 (2020).

[16] J.S. Gábor, L.R. Maria, and K.B. Nail, "Measuring and testing dependence by correlation of distances," The Annals of Statistics. Institute of Mathematical Statistics, vol 35, pp. 2769-2794, 2007.

[17] K.J. Cios, W. Pedrycz, and RW. Swiniarski, "Data mining: knowledge discovery methods," Hardback, Book. English. Published New York; London: Springer, pp. 29-37, 2007.

[18] L.P Phan, H.H. Huynh, and H.X. Huynh, "Hybrid recommendation based on implicative rating measures," Proceedings of the 2nd International Conference on Machine Learning and Soft Computing, (ICMLSC 2018), ACM, pp. 50-56, 2018.

[19] L. Sheugh, and S.H. Alizadeh, "A note on pearson correlation coefficient as a metric of similarity in recommender system," 2015 AI & Robotics (IRANOPEN), pp. 1-6, 2015.

[20] M. Rizzo, and G. Székely, "Energy distance. Wiley Interdisciplinary Reviews," Computational Statistics, vol 8, pp. 27-38, 2016.

[21] M. Hahsler, recommenderlab: "A Framework for Developing and Testing Recommendation Algorithm", 2015.

[22] S. Ramni, M. Sargam, T. Tanisha, N. Tushar, and S. Gaurav, "Movie Recommendation System using Cosine Similarity and KNN," International Journal of Engineering and Advanced Technology, Vol 9, pp. 2249-8958, 2020.

[23] S. Ben, J. Ben, F. Dan, Dan, Herlocker, Jon, Shilad, and S. Shilad, "Collaborative Filtering Recommender Systems," The Adaptive Web, Lecture Notes in Computer Science, Springer-Verlag, Berlin, Heidelberg, vol 4321, pp. 291-32, 2007.

[24] T. Park, X. Shao, and S. Yao, "Partial martingale difference correlation," Electronic Journal of Statistics, vol 9, pp. 1492-1517, 2015.

[25] Y. Koren, and R. Bell, Advances in Collaborative Filtering, Recommender Systems Handbook, Springer, pp. 145-186, 2011.