Approach of Item-Based Collaborative Filtering Recommendation Using Energy Distance

Tu Cam Thi Tran [1], Lan Phuong Phan [2], and Hiep Xuan Huynh [2,*]

[1] Faculty of Information Technology, Vinh Long University of Technology Education (VLUTE), Vinh Long Province, Vietnam

[2] College of Information and Communication Technology, Can Tho University (CTU), Can Tho City, Vietnam

Email: tuttc@vlute.edu.vn (T.C.T.T.); pplan@ctu.edu.vn (L.P.P.); hxhiep@ctu.edu.vn (H.X.H.)

*Corresponding author

***Abstract*—The current collaborative filtering recommendation method using energy distance only focuses on the relationship between the user and the user, between the user group and the user group. This method has not yet considered the relationship between the item and the item. In this article, we mainly focus on proposing an item-based collaborative filtering model using the energy distance. The proposed model is evaluated on two popular datasets Jester5k and MovieLens100k. Besides, the proposed model is also compared with two item-based collaborative filtering models using the Cosine and Pearson measures. The experimental results have shown that the proposed model is better than two compared models.**

***Keywords*—item-based, energy distance, collaborative filtering, incompatibility matrix, recommendation system**

I. INTRODUCTION

A recommendation system [1–4] is a form of information filtering system, its purpose is to provide the most relevant items for a particular user. The recommendation system is especially useful when a user needs to choose a few items (i.e., products, songs, movies, etc.) from a large number of items. Energy metric [5] measures the distance between the distributions of random vectors. The dimensions of those vectors are not certainly equal. Energy distance is also widely applied in researches [6] for example: testing independence by distance covariance, goodness-of-fit, generalizations of clustering algorithms, change point analysis, feature selection, and more. The smaller the energy distance between two users/items is, the closer the relationship between the two users/items is. Therefore, the energy distance measure can be used to find users/items, which are closely related; then applying this to collaborative filtering recommendation.

Energy-based recommendation systems were proposed in [5, 7]. Tran and Phan *et al.* focus on the relationship between two users in [5], whereas focus on the relationship between two groups of users to give the recommendations in [7]. However, the relationship between the items has not yet been considered in all these studies. Moreover, for the systems to be used for many years, the number of users grows faster than the number of items, so it takes longer time to find similar users than similar items. Besides, for the new users, when the system does not have this user's transaction information, the relationship between the items is especially useful in giving the recommendation.

In this article, we propose a new collaborative filtering recommendation model that considers relationships among the items. This approach is made on the basis of determining the energy relationship between items in pairs. In addition, the accuracy-based evaluation method (Precision, Recall, and F-measures) and the error-based evaluation method (RMSE and MAE measures) are used for comparing the proposed model, and two models available in the "recommenderlab" and "rrecsys" packages.

The article is structured as follows. Section II presents background, including: the collaborative filtering recommendation system, the energy distance between the items. Section III shows the methods to be used for evaluating recommendation models. Section IV depicts the proposed model that uses the energy relationship among items. Section V shows the experiment results on the Jester5k and MovieLens100k datasets. Section VI is the conclusion.

II. BACKGROUND

A. Collaborative Filtering Recommendation System

The recommendation system [2, 3, 8] can be seen as a quadruple set:

$$S = < U, I, R, f >$$

where:
$U = \{u_1, u_2, \dots, u_\mathrm{n}\}$ is a finite set of n users.
$I = i_1, i_2, \dots, i_m\}$ is a finite set of m items.
$R = \{r_{ij}\}$, $i = 1, \dots, n$, $j = 1, \dots, m$ is the rating matrix, in which r_{ij} is the rating value (feedback) of the user u_i for the item i_j. $R_k = \{r_{k1}, r_{k2}, \dots, r_{km}\}$ is the ratings of the user u_k for the item i_j, $j = 1..m$. The values of r_{kj} can be binary (0/1), integer, real or Not Available (NA) if the users $\mathrm{u_k}$ have not yet rated the item i_j.

Manuscript received March 17, 2023; revised May 5, 2023; accepted July 14, 2023, published January 3, 2024.

$$R = \begin{bmatrix} R_1 \\ R_2 \\ R_3 \\ \dots \\ R_n \end{bmatrix} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & \dots & r_{1m} \\ r_{21} & r_{22} & r_{23} & \dots & r_{2m} \\ r_{31} & r_{32} & r_{33} & \dots & r_{3m} \\ \dots & \dots & \dots & \dots & \dots \\ r_{n1} & r_{n2} & r_{n3} & \dots & r_{nm} \end{bmatrix}$$

$f : U \, x \, I \to R$ is an information function. $f(u_k, i_j)$ is used to estimate the rating of user $u_k \in U$ for item $i_j \in I$.

Collaborative Filtering Recommendation System (CFRS) [9–11] is a technique to predict user's preference and suggest the items that a user might prefer on the information of other users with similar preferences. There are two types of CF [9, 12–14], item-based collaborative filtering and user-based collaborative filtering. CFRS uses the common metrics such as Cosine, Pearson, and several others to measure the similarity and dissimilarity of the users or the items.

B. *Energy Distance*

Energy distance [5, 15] is a powerful tool for multivariate analysis. It is used to test for multivariate and univariate inference, multivariate independence, multivariate normality, distance components for non-parametric analysis of structured data, and more. Energy distance is applied to random vectors, where these random vectors have an unlimited size. Let $I_1 = I_{11}, I_{12}, \dots, I_{1n}$ and $I_2 = I_{21}, I_{22}, \dots, I_{2m}$be independent random vectors in Euclidean space. The distance of I_1 and $I_2$is determined according to the distance between the random vectors.

$$D^2(I_1, I_2) = 2\sum_{i=1}^{n}\sum_{j=1}^{m}\|I_1 - I_2\| - \sum_{i=1}^{n}\sum_{j=1}^{m}\|I_1 - I_1'\| - \sum_{i=1}^{n}\sum_{j=1}^{m}\|I_2 - I_2'\| \geq 0 \quad (1)$$

In Eq. (1), a random variable I_1' (or I_2') represents a copy, which is independent and distributed like I_1(or I_2).

The potential energy (shortly, energy) of the independent random variables I_1 and I_2 is defined by distance function ε as the follow:

$$\varepsilon_{n,m}(I_1, I_2) = 2E[\delta(I_1, I_2)] - E[\delta(I_1, I_1')] - E[\delta(I_2, I_2')] \quad (2)$$

where:

$$\mathrm{E}[\delta(\mathrm{I}_1, \mathrm{I}_2)] = \frac{1}{\mathrm{nm}}\sum_{\mathrm{i}=1}^{\mathrm{n}}\sum_{\mathrm{j}=1}^{\mathrm{m}}\left\|\mathrm{I}_{1\mathrm{i}} - \mathrm{I}_{2\mathrm{j}}\right\| \quad (3)$$

$$E[\delta(I_1, I_1')] = \frac{1}{n^2}\sum_{i=1}^{n}\sum_{j=1}^{n}\left\|I_{1\mathrm{i}} - I_{1\mathrm{j}}\right\| \quad (4)$$

$$E[\delta(I_2, I_2')] = \frac{1}{m^2}\sum_{i=1}^{m}\sum_{j=1}^{m}\left\|I_{2\mathrm{i}} - I_{2\mathrm{j}}\right\| \quad (5)$$

One can prove that $\varepsilon_{n,m}(I_1, I_2) = D^2(I_1, I_2)$ is zero if and only if $I_1$and I_2 have the same distribution. It is also true that the statistic $\varepsilon_{n,m}$ is always non-negative.

C. *CFRS Using Energy Distance*

The larger the energy distance between the user/item A and the user/item B is, the further the relationship between A and B is. Therefore, the energy distance can be used for collaborative filtering recommendation to find the nearest neighbors (i.e., the nearest users or the nearest items) of the user/item A.

There are currently two types of CFRS using energy distance: (1). Collaborative filtering recommendation systems based on the users using energy distance [5]; (2). Collaborative filtering recommendation systems based on the user groups using energy distance [7]. The first one mainly focuses on applying the energy distance for user-based collaborative filtering recommendation system. The second one concentrates on applying the energy distance for user group-based recommendation system. However, both two recommendation systems are based on the relationship of users rather than the relationship of items.

III. Evaluation Method

A. *K-folds Cross Validation*

The k-folds cross validation method [6, 16, 17] is usually chosen to partition the datasets, which are used to evaluate the recommendation models. In this method, the datasets are divided into a training set and a testing set. For example, if k-folds equals to 5. The dataset is divided into 5 parts/folds with equal size: 80% (4 parts) of the dataset for training and 20% (1 part) for testing. The recommendation models are evaluated recursively 5 times, each time using a different testing part. The results are then averaged to create the final result. Therefore, the k-folds cross validation method ensures that all users and items are considered for both training and testing.

B. *Recommendation Systems Evaluation*

The accuracy-based evaluation method is used to evaluate the recommendation models. This method builds the confusion matrix 2×2 [9, 17, 18] (Table I) to calculate the Precision value and the Recall value. Besides, F-measure–the harmonic means of precision and recall is also used to evaluate for the recommendation models. The higher the value of the Precision, Recall and F-measure, the better the model is evaluated. This helps designers to select the suitable model before applying it in practice.

TABLE I. Confusion Matrix

		Predicted	
		negative	positive
Actual	negative	a	b
	positive	c	d

$$Precision = \frac{correctly\ recommended\ items}{total\ recommended\ items} = \frac{d}{b+d} \quad (6)$$

$$Recall = \frac{correctly\ recommended\ items}{total\ useful\ recommendations} = \frac{d}{c+d} \quad (7)$$

$$F - measure = \frac{2\times precision\times recall}{precision+recall} = \frac{2}{1/precision+1/recall} \quad (8)$$

where:
a is the number of items that were not recommended by the system, and were also not actually preferred by users.
b is the number of items that were recommended by the system, and were not actually preferred by users.

c is the number of items that were not recommended by the system, and were actually preferred by users.
d is the number of items that were recommended by the system, and were also actually preferred by users.

Besides, the error-based evaluation method (Root Mean Square Error, and Mean Absolute Error) is also used to evaluate errors of the proposed model.

Root Mean Square Error (RMSE) [17, 18] is calculated between the real values and the predicted values as Eq. (9).

$$RMSE = \sqrt{\frac{\sum_{(i,j)\in n}(p_{ij}-\hat{p}_{ij})^2}{|n|}} \tag{9}$$

in which, p_{ij} be the real rating of user i for item j; $\hat{p}_{ij}$ be the predicted rating of user i for item j and n is the set of all user-item pairings (i, j).

Mean Absolute Error (MAE) [17, 18] is calculated between the real values and the predicted values as the following formula**.**

$$MAE = \frac{1}{|n|}\sum_{(i,j)\in n}\left|p_{ij}-\hat{p}_{ij}\right| \tag{10}$$

Among the compared models, the smaller error value the model has, the better the model is.

IV. Item-Based Recommendation Model Using Energy

A. *Modelling*

The items-based recommendation model using energy is shown in Fig. 1. This proposed model has the input be a tri-set (*U*×*1*×*R*). In which, *I* is the set of items, *U* is the set of users and *R* is the rating matrix. The model will calculate the distances between the items by the energy function shown in Eq. (2), save the result in an incompatibility matrix. Next, the model: (1) finds the K-Nearest Neighbors (KNN) for each item i_j; (2) predicts the missing ratings of all users for the item i_i and then predicts missing ratings of all users for all items. The output of the model is a table of predicted ratings.

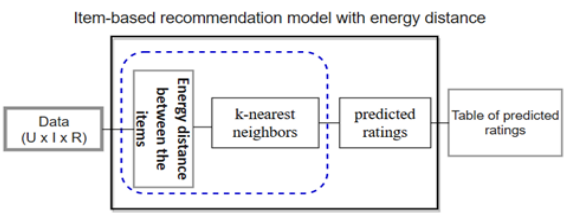

Fig. 1. The Item-based recommendation model using energy distance.

B. *Incompatibility Matrix of the Items*

The incompatibility matrix E represents the energy distances between the items. The incompatibility matrix $E = \{e_{ij}\}$, $i = 1, 2, \ldots, n$, $j = 1,2, \ldots, n$ and e_{ij} to be calculated by Eq. (2).

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

C. *K-Nearest Neighbors of an Item*

The energy distances in the incompatibility matrix are used for finding the k-nearest neighbors of the item i_j.

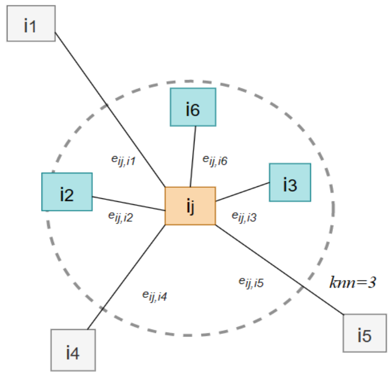

Fig. 2. Three nearest neighbors of the item i_j (KNN = 3).

Fig. 2 presents the 2D space of the energy distance between the item i_j and other items. The items with low energy will be displayed closer to i_j. If KNN equals to 3, the item i_2, i_3, and i_6 are selected to be three nearest neighbors of i_j.

D. *Algorithm*

The algorithm of proposed model is the follow:

Algorithm 1. Item-based recommendation algorithm using energy

Input: The tri-set (U × I × R); the number of nearest neighbors *knn*.
$U = \{u_1, u_2, \ldots, u_n\}$; $I = \{i_1, i_2, \ldots, i_m\}$; $R = \{r_{ij}\}$, $i = 1, \ldots, n$, $j = 1, \ldots, m$
Output: The table of predicted ratings to be used for recommendation R′.
Begin

// Building the incompatibility matrix Energy
1: for *i* = 1 to m do
2: calculate the energy distance of each item i_j with all other items;
3: end for
// Finding *k* nearest items of each item i_k,
// then creating the matrix K of size m *x* knn
4: for *k* = 1 to m do
5: // finding knn (*k* nearest items of the item i_k by using the rating matrix R and the incompatibility matrix Energy)
for *q* = 1 to knn
K[i_k][*q*] = i_t where E[i_k][i_t] has the q^{th} smallest energy distance
6: end for
7: end for
// Predicting the missing rating values based on
// k nearest neighbors.
8: for *i* = 1 to *n* do
9: for *j* = 1 to m do
10: if (R[u_i][i_j] == NA && Energy[i_j][i_k] != 0 && $i_k \in$ K[i_j, 1:knn])

11: R'[u_i][i_j] = (1/Energy[i_j][i_k]) x (Energy[i_j][i_k] x R[u_i][i_j])
12: end for
13: end for
14. return the table of predicted ratings R′.
End.

V. Experiment

A. Datasets

Experiments in this article are performed on two datasets Jester5k [8] and MovieLens100k [19]. These two datasets are commonly used in the recommendation systems. The information of these datasets is summarized in Table II.

Jester5k contains the ratings of 5000 anonymous users collected from the Jester Online Joke Recommendation System between April 1999 and May 2003. All selected users have rated 36 or more jokes.

MovieLense100k includes 100,000 ratings published in 1998 by GroupLens. Each user has rated at least 20 movies.

TABLE II. The Table to Describe Datasets: Jester5k and MovieLens100k

Names	Number of rows (users)	Number of cols (items)	Number of ratings	Value domain of ratings
Jester5k	5,000	100	362,106	−10~+10
MovieLens100k	943	1682	99,392	1–5

B. Tool

In the experiment of the article, R language is used to build the proposed model (named IBCFEnergy RS-Items Based Collaborative Filtering Recommendation System using Energy distance).

The IBCFEnergy RS model is compared with two models including: IBCFCosine RS (Items Based Collaborative Filtering Recommendation System using Cosine measure) and IBCFPearson RS (Items Based Collaborative Filtering Recommendation System using Pearson measure). These two models are in the "recommenderlab" [9] "rrecsys" [20] packages. Two models use Cosine and Pearson measures, respectively to find *k*-nearest neighbors of the items.

In the first two scenarios, the accuracy (Precision, Recall, F1) of the proposed model is compared with the two IBCFCosine RS and IBCFPearson RS models of recommenderlab package. In recommenderlab, the dataset is partitioned into a training set and a testing set. The testing set is again divided into two parts: The query set and the target set (= testing set – query set) have the same size. Suppose, a user u in the test set has rated l items (l ratings), then the "*given*" ratings are selected at random ("*given*" ratings are also called the *known* ratings) in the query set and the remaining ratings of user u - (l-*given*) - are in the target set. The training set is used to build the recommendation model whereas the query set is used to predict the list of recommendations based on the built model; and the target set is used to evaluate the prediction results through the accuracy measures. In the third scenario, the errors (RMSE, MAE) of the proposed model are compared those of IBCFPearson RS model of the rrecsys package. In rrecsys, the dataset is partitioned into a training set and a testing set. The training set is used to learn the model whereas the testing set is used to evaluate the model.

C. Scenario 1: Accuracy-Based Evaluation on Jester5k

This scenario evaluates the F1, Precision and Recall values of three recommendation models IBCFCosine RS, IBCFPearson RS, and IBCFEnergy RS on the Jester5k dataset with the known ratings of each user in the testing set (*given*) to be 10, 18, 22, 30, 34; the number of items to recommend (n) to be 1, 3, 6, 9, 12 and k nearest items (knn) to be 30.

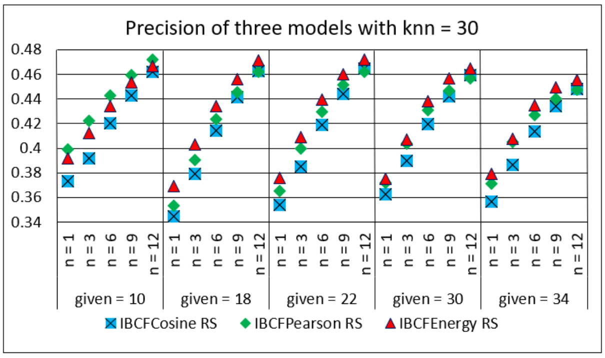

Fig. 3. Precision chart of three models with KNN = 30 on Jester5k dataset.

The experimental results in Fig. 3 show that the Precision values of IBCFEnerergy RS are always higher than the Precision values of IBCFCosine RS and IBCFPearson RS. However, when $given$ equals to 10 (the number of known ratings is small), the Precision value of the proposed model is smaller than those of IBCFPearson RS.

Because the Recall values of IBCFCosine RS are the lowest, the IBCFEnergy RS is compared to the IBCFPearson RS by using the Recall value differences between each of these two models and IBCFCosine RS as Fig. 4. The result shows that the difference values of the proposed model is higher than the model using Pearson measure; especially when the *given* value increases. However, when the *given* VALUE is less than or equal to 10, the Recall value of the proposed model is smaller than those of IBCFPearson RS.

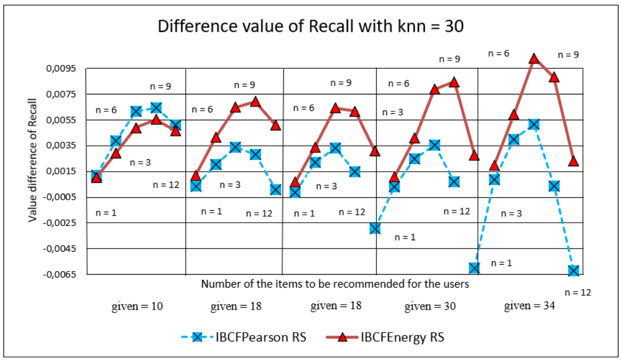

Fig. 4. Difference value of recall for two models IBCFPearson RS and IBCFEnergy RS compared to IBCFCosine RS on Jester5k dataset.

Besides, the experimental result shows that the F1 values of IBCFCosine RS are the lowest, therefore Fig. 5 only presents the F1 value differences between each of two models (IBCFEnergy RS and IBCFPearson RS) and IBCFCosine RS. It means that F1 values of IBCFEnergy RS is higher than the IBCFPearson RS, especially when the $given$ value increases. However, when $given$ is less than or equal to 10, the F1 value for the proposed model is smaller than those of IBCFPearson RS.

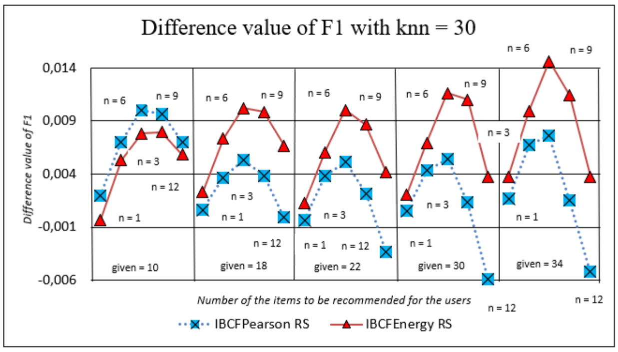

Fig. 5. Difference value of F1 for two models IBCFPearson RS and IBCFEnergy RS compared to IBCFCosine RS on Jester5k dataset.

D. Scenario 2: Accuracy-Based Evaluation on Movielense100k

Because F1 is the harmonic means of precision and recall values, the Scenario 2 focus on comparing the F1 values of three models IBCFEnergy RS, IBCFCosine RS and IBCFCosine RS on the MovieLense100k dataset. The $given$ of each user in the testing set to be 2, 8, 14, 17; the number of items to recommend (n) to be 1, 3, 6, 9, 12 and K Nearest item (KNN) to be 30, 40.

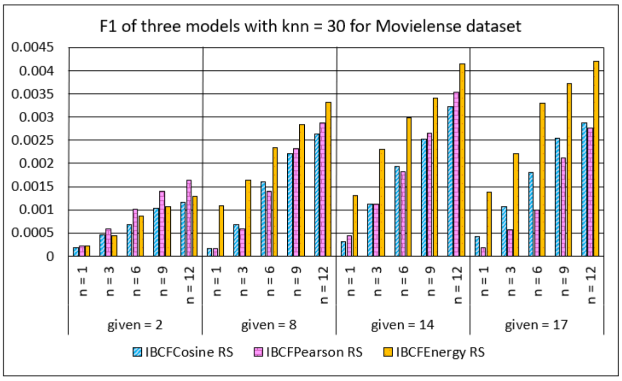

Fig. 6. F1 chart of three models with KNN = 30 on Movielense100k dataset.

Fig. 6 presents the results of comparing the F1 values of three models (IBCFEnerergy RS, IBCFCosine RS and IBCFPearson RS) when KNN equals to 30. It shows that the F1 values of IBCFCosine RS is smallest. Fig. 7 presents the F1 difference value of IBCFEnerergy RS and IBCFPearson RS respectively compared with IBCFCosine RS. The experimental results presented in Figs. 6 and 7 show that the $given$ value increases, the F1 value of the IBCFEnerergy RS is higher than the F1 value of the IBCFCosine RS and IBCFPearson RS. However, when $given$ to be 2, the F1 value of the IBCFEnergy RS is smaller than that of IBCFPearson RS.

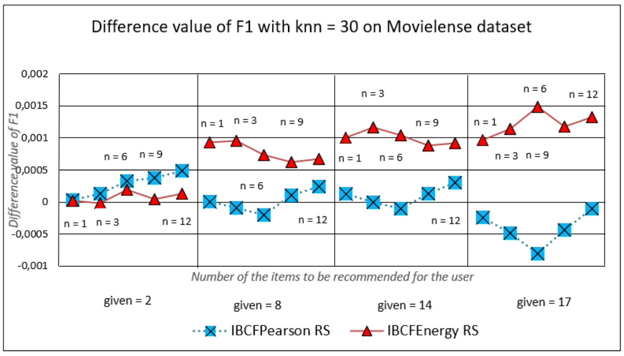

Fig. 7. Difference value of F1 for two models IBCFPearson RS and IBCFEnergy RS compared to IBCFCosine RS on MovieLens100k dataset.

When KNN equals to 40 and $given$ equals to 2, 4, 14, 17, the experimental results are displayed in Figs. 8 and 9. These figures shows that the conclusion for KNN = 30 as mentioned in above paragraph is right for KNN = 40.

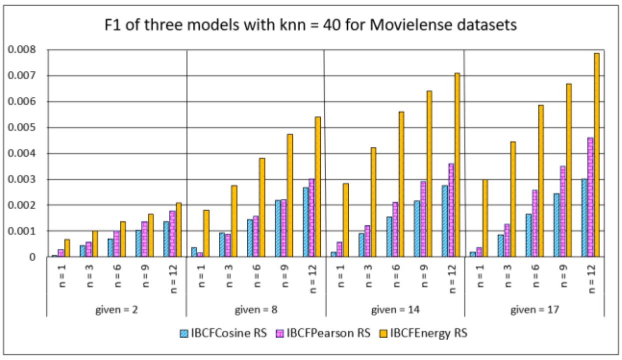

Fig. 8. F1 chart of three models with KNN = 40 on MovieLens100k dataset.

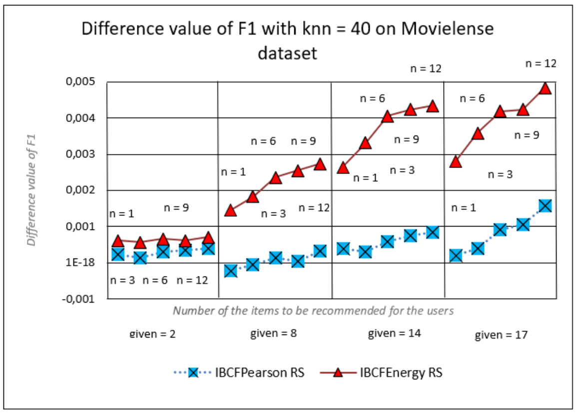

Fig. 9. Difference value of F1 for two models IBCFPearson RS and IBCFEnergy RS compared to IBCFCosine RS on Movielense100k dataset.

E. Scenario 3: Error-Based Evaluation on both Jester5k and Movielense100k

Scenario I and II shows that in three compared models, IBCFCosine RS always has the smallest F1, Precision, Recall values. Therefore, Scenario 3 will concentrate on comparing the errors (RMSE and MAE) of two models

IBCFEnergy RS, IBCFPearson RS on both Jester5k and MovieLense100k.

RMSE and MAE error values of both IBCFPearson RS and IBCFEnergy RS on Jester5k and MovieLense100k are shown in Figs. 10 and 11, respectively. The MAE and RMSE values of IBCFEnerergy RS are always lower than the MAE and RMSE values of IBCFPearson RS, when KNN is changed from 5, 10, 20, 30, 40 for Jester5k, and from 10, 20, 30, 40, 50, 60 for Movielense100k.

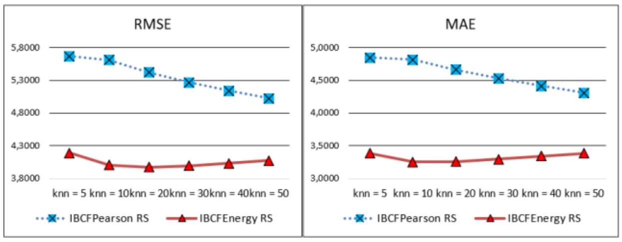

Fig. 10. MAE and RMSE for two models IBCFPearson RS and IBCFEnergy RS on Jester5k dataset.

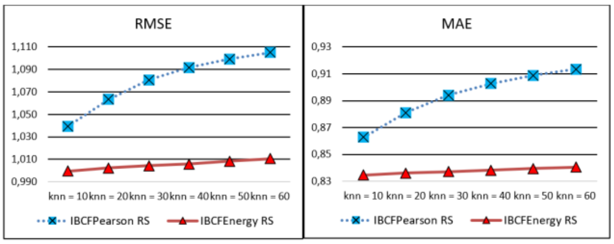

Fig. 11. MAE and RMSE for two models IBCFPearson RS and IBCFEnergy RS on Movielense100k dataset.

VI. Conclusion

We proposed the item-based collaborative filtering model with energy distance to recommend the suitable items to users, then enrich for the CF methods. The new recommendation model has been experimented on the Jester5k and MovieLens100k datasets, and compared with the item-based collaborative filtering recommendation models using Pearson and Cosine measures of the "recommenderlab" and "rrecsys" packages. The performance of the proposed model is evaluated by the accuracy and error metrics such as Precision, Recall, F1, RMSE and MAE. The experimental results have shown that the proposed recommendation model has a higher Precision, Recall and F1 values compared to those of two compared models on the both Jester5k dataset and Movielens100k dataset. However, when the number of known ratings ($given$) is small, the Precision, Recall and F1 values are not high. In addition, the RMSE and MAE errors of the proposed model are lower than the RMSE and MAE errors of the compared model. Generally, the experimental results of the proposed model show the applicability of the energy distance to the item-based collaborative filtering recommendation system. This contributes to more diversity for recommendation approaches. In the future, the authors will test the proposed model on a larger data range to fully evaluate it as well as develope collaborative filtering recommendation model using energy distance on the groups of items.

Conflict of Interest

The authors declare no conflict of interest.

Author Contributions

All authors wrote the manuscript and analyzed the experimental results of the research. Tu collected the data and wrote the manuscript. Lan analyzed the data and ran the experimental results. Hiep provided the corrections and interpreted the results. All authors had approved the final version of the manuscript.

References

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

[2] 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, no. 1, pp. 103–145, 2005.

[3] G. Adomavicius, N. Manouselis, and Y. Kwon, "Multi-criteria recommender systems," in *Recommender Systems Handbook*, 2011, pp. 769–803.

[4] 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, no. 6, pp. 734–749, 2005.

[5] T. C. T. Tran, L. P. Phan, and H. X. Huynh, "Energy-based collaborative filtering recommendation," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 13, no. 7, pp. 557–562, 2022.

[6] M. Rizzo and G. Székely, "Energy distance," *Wiley Interdisciplinary Reviews: Computational Statistics*, vol. 8, no. 1, pp. 27–38, 2016.

[7] T. C. T. Tran, L. P. Phan, and H. X. Huynh, "A group clustering recommendation approach based on energy distance," in *Proc. Computational Data and Social Networks, CSoNet 2022. Lecture Notes in Computer Science*, Springer, Cham, 2023, vol. 13831, pp. 93–104.

[8] Jester5k: Jester dataset (5k sample). In recommenderlab: Lab for developing and testing recommender algorithms. [Online]. Available: https://rdrr.io/cran/recommenderlab/man/Jester5k.html

[9] M. Hahsler. (2015). Recommenderlab, an R framework for developing and testing recommendation algorithms. [Online]. Available: https://cran.r-project.org/web/packages/recommenderlab/vignettes/recommenderlab.pdf

[10] 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.

[11] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, "Collaborative filtering recommender systems," *Foundations and Trends in Human–Computer Interaction*, vol. 4, no. 2, pp. 81–173, 2010.

[12] B. M. Sarwar, G. Karypis, J. A. Konstan, and J. Riedl, "Item-based collaborative filtering recommendation algorithms," in *Proc. the 10th international conference on World Wide Web (WWW '01). Association for Computing Machinery*, New York, NY, USA, 2001, pp. 285–295.

[13] C. Li and L. Ma, "Item-based collaborative filtering algorithm based on group weighted rating," in *Proc. the 2020 13th International Symposium on Computational Intelligence and Design (ISCID)*, Hangzhou, China, 2020, pp. 114–117.

[14] G. Linden, B. Smith, and J. York, "Amazon.com recommendations: Item-to-item collaborative filtering," *IEEE Internet Comput.*, vol. 7, pp. 76–80, 2003.

[15] 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, 2021.

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

[17] 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.
[18] M. D. Ekstrand, J. T. Riedl, and J. A. Konstan, “Collaborative filtering recommender systems,” *Foundations and Trends in Human–Computer Interaction*, vol. 4, no. 2, pp. 81–173, 2010.
[19] MovieLense: MovieLense Dataset (100k). In recommenderlab: Lab for developing and testing recommender algorithms. [Online]. Available: https://rdrr.io/cran/recommenderlab/man/MovieLense.html
[20] L. Çoba, M. Zanker, and P. Symeonidis. rrecsys: Environment for evaluating recommender systems. [Online]. Available: https://rdrr.io/cran/rrecsys/