Item-Based Context-Aware Collaborative Filtering Using Energy Distance with Pre-filtering Contextual Feature

Linh Thuy Thi Nguyen [1,2], Lan Phuong Phan [1], and Hiep Xuan Huynh [1,2,*]

[1] College of Information and Communication Technology, Can Tho University, Can Tho City, Vietnam
[2] CTU Leading Research Team on Automation, Artificial Intelligence, Information Technology and Digital Transformation (CTU-AIMED), Can Tho University, Can Tho, Vietnam
Email: nttlinh@ctu.edu.vn (L.T.T.N.); pplan@ctu.edu.vn (L.P.P.); hxhiep@ctu.edu.vn (H.X.H.)
*Corresponding Author

***Abstract*—This paper proposes an Item-Based Context-Aware Collaborative Filtering (IB-CACF) approach that combines Energy Distance and Pre-Filtering Contextual Features to improve the effectiveness of recommendation systems. The proposed method addresses the limitations of traditional recommendation systems, which fail to consider contextual factors such as time, location, and companions. Energy Distance measures the similarity between items in the context space, enhancing prediction accuracy. Along with Pre-Filtering Contextual Features, this approach reduces computational cost by filtering out the most relevant contextual factors before similarity calculations. Experimental results on the MovieLens 25M, Amazon, and Yelp datasets demonstrate that this method outperforms existing approaches in terms of both prediction accuracy and computational efficiency.**

***Keywords*—item-based, energy distance, context-aware, collaborative filtering, contextual pre-filtering**

I. Introduction

Recommender systems have become crucial in a variety of applications, from e-commerce platforms to online content streaming services. Among the various recommendation techniques, Collaborative Filtering (CF) has been widely used due to its simplicity and effectiveness [1]. However, traditional CF methods fail to consider contextual factors that can significantly influence user behaviour, such as time, location, and device [2].

To address the limitations of traditional CF methods, Context-Aware Recommender Systems (CARS) have been introduced, which incorporate contextual information into the recommendation process. Contextual features such as time, location, device, or mood can significantly influence user preferences and behaviour. CARS approaches are generally categorized into three strategies: pre-filtering, post-filtering, and contextual modelling [3, 4]. Pre-filtering methods filter the training data based on contextual information before applying a CF algorithm. This approach has the advantage of maintaining the scalability and simplicity of traditional CF while still accounting for context. For example, a movie recommender system may only consider user ratings given at specific times of the day or on weekends. Postfiltering methods, on the other hand, apply CF first and then adjust the recommendations based on the context. Finally, contextual modelling directly incorporates context as an additional dimension in the recommendation algorithm, often using tensor factorization [5]. Context-aware recommender systems have emerged as a solution to this limitation by incorporating contextual information into the recommendation process. Despite this, many context-aware CF models still face challenges in efficiently handling the complex relationships between users, items, and contextual factors. While CARS offers numerous benefits, there remain significant challenges in applying these methods: Integrating multiple contextual factors can lead to dimensionality explosion, increasing computational complexity and requiring more processing resources. This poses challenges for real-world deployment, especially in large systems. Non-uniform contexts can reduce the effectiveness of recommendation methods. Not all contextual factors positively influence user preferences; inaccurate selection of factors can decrease model accuracy.

Context in Context-Aware Recommendation Systems (CARS) may include multiple factors, such as Time: The moment a user acts, for example, selecting a movie in the morning or evening; Location: The place where the action occurs, such as watching a movie at home or in a cinema; Companion: Individuals or groups accompanying the user, which can influence decision-making. These factors provide additional information and help identify and analyse user habits and preferences in specific contexts.

Baltrunas have compiled existing methods for incorporating context into recommendation systems [6], including Pre-Filtering: Processing contextual information before building the recommendation model and removing irrelevant or non-essential factors. This study explores the theoretical properties of Energy Distance and the

Manuscript received January 24, 2025; revised February 21, 2025; accepted April 23, 2025; published July 25, 2025.

Incompatibility Matrix in recommendation systems. Energy Distance measures distributional discrepancies in user preferences, while the Incompatibility Matrix identifies contextual inconsistencies. MovieLens dataset demonstrates that Energy Distance improves recommendation accuracy by capturing nonlinear relationships, and the Incompatibility Matrix enhances contextual adaptation [7, 8]. Limited Domain Generalization: Although the model performs well on MovieLens and Jester5K, its effectiveness in other domains, such as e-commerce or healthcare, remains unexplored. High Computational Cost: Energy Distance calculations require substantial computational resources, limiting scalability in large-scale applications. Contextual Data Dependency: The model heavily relies on the availability and quality of contextual information. Missing or noisy data can degrade recommendation accuracy.

In this paper, we propose a new approach that integrates Item-Based Context-Aware Collaborative Filtering with Energy Distance and Pre-Filtering Contextual Features. This approach leverages the power of Energy Distance, a robust similarity measure, to compute item similarities while accounting for contextual influences. Additionally, we introduce the concept of Pre-Filtering contextual features, where only the most relevant contextual information is selected before making similarity calculations. This pre-filtering step reduces the computational complexity and enhances the model's efficiency and recommendation accuracy. We evaluate the proposed approach on the MovieLens 25M dataset and demonstrate its superior performance compared to traditional CF and context-aware CF methods [1].

The rest of the paper is organized as follows: Section II theoretical foundation of the proposed on Collaborative Filtering and Energy Distance and Pre-Filtering Contextual Feature. Section III Proposed Model, including the integration of Energy Distance and Pre-Filtering techniques. Section IV describes the experimental setup and results. Finally, Section V concludes the paper and discusses future directions for this research.

II. THEORETICAL FOUNDATION

In this section, we present the theoretical foundation of the proposed Context-Aware Collaborative Filtering (CACF) method using Energy Distance and Pre-Filtering Contextual Features.

A. Collaborative Filtering (CF)

Collaborative Filtering is one of the most widely used methods in *recommender systems*, particularly User Based CF and Item-Based CF [9]. In this approach, we predict the rating $\hat{r}_{ui}$ for user u on item i based on other users with similar behaviours:

$$\hat{r}_{ui} = \bar{r}_u + \frac{\sum_{v \in N(u)} w_{uv}(r_{vi} - \bar{r}_v)}{\sum_{v \in N(u)} w_{uv}} \tag{1}$$

where $N(u)$ is the set of users who are highly similar to the user u, and w_{uv} is the similarity weight between users u and v [10].

B. Context-Aware Collaborative Filtering

Collaborative Filtering (CF) typically relies on user-item interaction data, but it fails to account for contextual factors such as time, location, and companions. In this work, we extend CF by incorporating contextual information, which leads to Context-Aware Collaborative Filtering (CACF). This allows the recommendation system to adapt to different situations by considering the context in which interactions take place. The prediction in CACF can be formulated as:

$$\hat{r}_{ui} = f(u, i, C) \tag{2}$$

where $\hat{r}_{ui}$ is the predicted rating for user u on item i, and C represents the contextual information (such as time, location, or companions).

C. Pre-filtering Contextual Features

The pre-filtering Contextual Features step is introduced to enhance the system's efficiency. In this step, irrelevant contextual information is filtered out before similarity calculations are performed, enabling the system to focus solely on the most relevant features. By selecting the most informative contextual factors, the dimensionality of the input data is reduced, leading to improved computational performance.

The pre-filtering process is formalized by selecting a subset of the available contextual features $C' \subseteq C$ in such a way that the set C' maximizes its relevance to the user-item interaction.

Reduced computational complexity: Eliminates redundant features, speeding up training and prediction. Improved model accuracy: Focuses on relevant contextual factors, avoiding noise.

Lower risk of overfitting: Enhances generalization by removing irrelevant features. Better interpretability: A simpler model is easier to analyze and explain. Preprocessing Steps in Pre-Filtering Contextual Features: (1) Data Collection & Cleaning; (2) Feature Encoding; (3) Feature Filtering; (4) Dimensionality Reduction; (5) Optimized Features.

D. Energy Distance in Recommender Systems

Energy distance, a metric derived from statistical energy analysis, has recently gained attention in recommendation systems. Introduced in the field of statistical hypothesis testing, energy distance measures the difference between two probability distributions [11]. In the context of recommender systems, energy distance can be used to measure the disparity between user preferences in different contexts, helping to identify how strongly context influences user behaviour. Tran *et al.* [8] applied energy distance to the task of context-aware recommendations. Their work demonstrated that by calculating energy distances between users' contextualized interactions, the recommendation system could better capture subtle changes in user preferences that traditional CF methods fail to address. Their experiments on the MovieLens dataset showed that incorporating energy distance improved both precision and recall in recommendation tasks, particularly when used in conjunction with pre-filtering strategies to limit the influence of irrelevant contextual features.

E. Energy Distance

Energy Distance is a metric used to measure the difference between two probability distributions. It is beneficial for comparing distributions that traditional distance metrics like Euclidean distance may not easily describe. The Energy Distance between two distributions P and Q is defined as:

$$D_E(\mathrm{P},\mathrm{Q}) = [2\mathbb{E}\|X - Y\|^2 - \|X - X'\|^2 - \|Y - Y'\|^2]^{\frac{1}{2}} \quad (3)$$

where X and Y are random variables with distributions P and Q, respectively, and X' and Y' are independent copies of X and Y. E[·] denotes the expectation (mean). $\|\cdot\|^2$ represents the Euclidean norm.

F. Energy Distance in Contextual Similarity Calculation

We combine Energy Distance with pre-filtered contextual features to calculate the similarity between items in a context-aware manner. The similarity between items i and j considering context C' is given by:

$$S(i,j|C') = D_E(P_\mathrm{i}, P_\mathrm{j}) \quad (4)$$

where P_i and P_j are the probability distributions of the items i and j considering their contextual features in C'.

G. Evaluation Metrics

The performance of the models is evaluated using: Mean Absolute Error (MAE) [12, 13]: MAE measures the average prediction error and is defined as:

$$\mathrm{MAE} = \frac{1}{|R|}\sum_{i=1}^{N}\left|\hat{R}_i - R_i\right| \quad (5)$$

where: N is the number of predictions, $\hat{r}_i$ is the predicted rating, and R_i is the actual rating.

Root Mean Square Error (RMSE) [12, 13]: RMSE assesses the square root of the average squared prediction error and is defined as:

$$\mathrm{RMSE} = \sqrt{\frac{1}{|R|}\sum_{i=1}^{n}(r_{ui} - \hat{r}_{ui})^2} \quad (6)$$

where: N is the number of predictions, $\hat{r}_i$ is the predicted rating, and R_i is the actual rating.

III. Proposed Model

The model for IB-CACF with Energy Distance and Pre-Filtering Contextual Features is shown in Fig. 1, and its algorithm is depicted in Algorithm 1.

In the IB-CACF approach, we propose works as follows: 1). Contextual Filtering: The first step is to filter the dataset to include only the ratings that meet specific contextual conditions. For example, we may filter the dataset to focus on movies watched at home with friends in the evening. 2). Similarity Calculation: After filtering the data, we compute the similarity between users based on their ratings for movies in the same context. We use similarity measures such as Cosine Similarity or Pearson Correlation to calculate the degree of similarity between users. Prediction: Based on the similarity between users, we predict the rating for a given user on a movie that they have not rated yet, using the ratings from similar users.

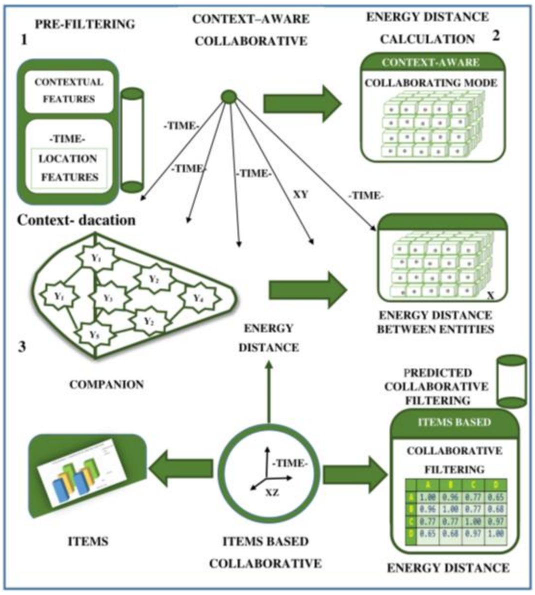

Fig. 1. IB-CACF with energy distance and pre-filtering contextual features.

Algorithm 1. Item-Based CACF with Energy Distance and Pre-Filtering Contextual Features

Step 1: **Preprocess the Data**
Load user-item interaction data, D
Extract contextual features C from the data (e.g., time of day, location)
Step 2: **Pre-filter Contextual Features**
Filter out irrelevant contextual features using a threshold *threshold*
Select the relevant contextual features $C' \subseteq C$
Step 3: **Compute Item Similarity using Energy Distance**
for each pair of items i and j do
    Calculate the Energy Distance $D_E(P_i,P_j)$ between items i and j based on their contextual features in C'
end for
Step 4: **Predict Ratings**
for each user u and item i **do**
    Compute the predicted rating $\hat{r}_{ui}$ as a weighted sum of ratings of similar items using the similarity matrix
    $\widehat{r_{ui}}$
end for
Step 5:**Compute Loss Function**
Define Energy-based Context-aware Loss as:
$\mathcal{L}$
Where λ is a regularization parameter controlling the impact of Energy Distance
Step 6: Optimize the Model (Optional)
Minimize L using gradient descent or an optimization algorithm
Step 7: Evaluate the Model
Evaluate the model using RMSE or MAE

A. Pre-filtering and Energy Distance Combination

Combining pre-filtering with energy distance represents a novel approach to enhancing context-aware recommendation systems. In pre-filtering, contextual features are used to segment the data into contextually relevant subsets, ensuring that only interactions occurring in similar contexts are considered. Energy distance can then be applied to these subsets to measure the distance between user-item interactions in different contexts. This approach is particularly advantageous in dynamic environments, where user preferences fluctuate based on contextual changes, such as seasonal trends or shifts in user behaviour over time. By applying energy distance, the system can better distinguish between subtle contextual influences, enabling more accurate and personalized recommendations. Studies like [7] have shown that this hybrid method outperforms traditional CF and CARS approaches, particularly in domains with complex, multidimensional contextual data. The integration of energy distance allows for a more nuanced representation of user preferences, while pre-filtering ensures that the model remains scalable and efficient.

B. Context-Based Energy Distance

Energy distance, denoted as D_E, is a measure of statistical distance between two distributions. In this context, it is used to evaluate the discrepancy between the contextual distributions of user-item interactions. For two sets of user-item interactions X and Y, with context C_X and C_Y, the Energy distance is defined as:

$$D_E(X,Y) = 2\mathbb{E}[\|C_X - C_Y\|] - \mathbb{E}[\|C_X - C_X'\|] \quad (7)$$

where $\| \cdot \|$ represents a distance metric (e.g., Euclidean distance), and E denotes the expectation. The goal is to minimize the Energy Distance during training, leading to closer contextual distributions and better recommendations.

C. Context-Based Energy Loss Function

The enhanced loss function for CACF integrates both standard error-based loss (e.g., Mean Squared Error, MSE) [14] and Energy distance:

$$\mathcal{L} = \alpha \cdot \mathrm{MSE}(R, \hat{R}) + \beta \cdot D_{\mathrm{E}}(X, Y) \quad (8)$$

where: R, $\hat{R}$ are the true and predicted ratings, and α, β are hyperparameters controlling the trade-off between the standard error loss and Energy Distance. By minimizing this loss, the model ensures accurate predictions while considering contextual similarities. The composite loss function includes both prediction error and energy function components. The composite loss function can be expressed as:

Energy Distance Loss Function:

$$\mathcal{L}_{\text{prediction}} = \frac{1}{|R|} \sum_{(u,i)\in R} (r_{ui} - \hat{r}_{ui})^2 \quad (9)$$

Contextual Loss Function:

$$\mathcal{L}_{\text{energy}} = \frac{1}{|R|} \sum_{(i,j)\in N} (D_{\text{Energy}}(i,j)) \quad (10)$$

$$\mathcal{L}_{\text{context}} = \frac{1}{|R|} \sum_{(u,i)\in R} \left(C_{ui} - \hat{C}_{ui}\right)^2 \quad (11)$$

Combined Loss Function:

$$\mathcal{L} = \lambda 1.\, \mathcal{L}_{\text{prediction}} + \lambda 2.\, \mathcal{L}_{\text{energy}} + \lambda 3.\, \mathcal{L}_{\text{context}} \quad (12)$$

where:

- r_{ui} is the actual rating given by user u to item i.
- $\hat{r}_{ui}$ is the predicted rating by the model.
- $|R|$ is the total number of ratings in the dataset.
- $D_{\text{Energy}}(i,j)$: Energy distance between items i and j.
- N: Set of similar items to be optimized.
- C_{ui}: Actual context value of user u for item i.
- $\hat{C}_{ui}$: Predicted context value of the model for item i by user u.
- λ_1, λ_2, λ_3: Weight adjustment coefficients for each component of the loss function.

The first term, $\sum_{(u,i,c)}(r_{u,i,c} - \hat{r}_{u,i,c})^2$ measures the Mean Squared Error (MSE) between the observed ratings and the predicted ratings. This component ensures that the model's predictions are as close as possible to the actual ratings.

The second term, $\lambda E(c_1,c_2)$, incorporates the Energy Distance, which evaluates the discrepancy between contextual distributions. Minimizing this term, the model adjusts to significant contextual variations, improving the overall recommendation quality.

IV. Experimental Setup

A. Datasets

In this study, we employed three datasets to evaluate the performance of our proposed method: the MovieLens dataset, the Yelp dataset, and the Amazon dataset.

MovieLens dataset: The MovieLens dataset used in this study is the *ml-25m* dataset, which contains 25 million movie ratings and 1 million tag applications across 62,000 movies and 162,000 users. The dataset includes timestamp information that contextualises when ratings were made [15].

Yelp dataset: The Yelp dataset comprises user reviews, business information, and ratings across various categories. We used the subset that includes business and review data, with a focus on contextual features such as the time of day, day of the week, and location of businesses [16].

Amazon dataset: The Amazon dataset includes product reviews and ratings across a broad range of product categories. For our study, we utilised the review data with contextual information such as product categories and purchase dates [17].

We use three datasets for evaluation: MovieLens 25M: Contains movie ratings with contextual dimensions such as Time of Day: Morning, Afternoon, and Evening. Day of the Week: Weekday, Weekend.

Yelp: Includes reviews of local businesses with contextual dimensions, such as Location: Home, Office. Time of Day: Morning, Afternoon, Evening. Amazon: Contains product reviews from users with contextual dimensions, such as Product Categories: Electronics, Books, and Fashion. Location: Urban, Rural.

Feature Extraction Contextual features were extracted from each dataset. For the MovieLens dataset, contextual features included the time of day, day of the week, and movie genre. For the Yelp dataset, features included the business location, time of day, and day of the week. For the Amazon dataset, we focused on the product category and review timestamps. Feature Selection To reduce dimensionality and avoid overfitting, we performed feature selection based on the relevance of contextual features to user preferences. Mutual information was used to select the most significant features, ensuring that only the most informative context features were retained.

B. Experimental Results

All experiments were conducted using Python, with libraries such as Scikit-learn for machine learning algorithms, NumPy for numerical computations, and TensorFlow for deep learning models. Hyperparameters for the collaborative filtering algorithms were tuned using grid search with cross-validation.

The performance of the models is evaluated using: MAE (Mean Absolute Error): Which measures the average prediction error. RMSE (Root Mean Square Error): Assesses the square root of the average squared prediction error. Energy Distance: Quantifies the difference between the contextual distributions of user-item interactions. The impact of pre-filtering on recommendation performance is presented in Table I [18]. In contrast, the EBM effectively leverages contextual information through Energy Distance, enabling better modelling of the interactions between users, items, and contextual factors.

TABLE I. PERFORMANCE COMPARISON OF SVM AND EBM RMSE

Dataset	SVM RMSE	EBM RMSE	Improvement
MovieLens 25M	1.450	1.387	4.35%
Yelp	1.387	1.175	15.3%
Amazon	1.458	1.249	14.3%

Experimental results demonstrate that EBM outperforms SVM in terms of accuracy, particularly in context-aware recommendation systems. By measuring the differences between contextual distributions of user-item interactions, EBM provides more precise recommendations compared to SVM [18]. By removing irrelevant contextual features, the enhanced CACF model shows improved accuracy compared to the model without pre-filtering. The experimental results show that the model using Energy Distance improves the accuracy of the recommendations. Below are the specific results of EBM (Energy-based models) from the three datasets is presented in Fig. 2.

The reduction in Energy Distance from the three datasets used are:

MovieLens 25M: Contains user movie ratings EBM achieved a lower RMSE (a reduction of 4.35%), indicating better performance, although the improvement is insignificant.

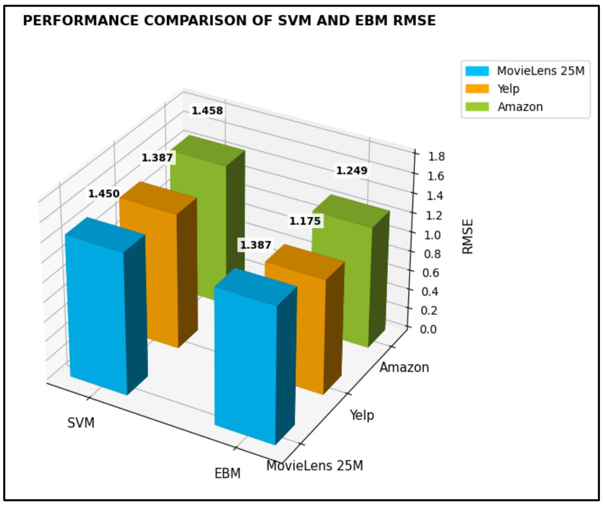

Fig. 2. Comparison of SVM and EBM RMSE context application.

Yelp: contains reviews of businesses and restaurants. The performance of EBM is outstanding, with RMSE reduced significantly (15.3%), demonstrating its ability to handle non-linear relationships.

Amazon: It contains product reviews across various categories. EBM reduced RMSE by approximately 14.3%.

The pre-filtering technique has demonstrated its value by significantly improving the quality of the input data, resulting in more accurate predictions. This is evidenced by the reduction in Energy Distance, which indicates a closer alignment between predicted and actual ratings. This outcome highlights the model's ability to effectively integrate contextual factors, enabling it to refine input data before prediction, ultimately enhancing recommendation quality. However, by improving the adaptability and precision of the pre-filtering step, the model can more effectively cater to individual user needs and various data environments, resulting in stronger recommendations.

V. CONCLUSION

This paper introduces a novel approach to enhance recommendation systems by integrating Item-Based Context-Aware Collaborative Filtering (IB-CACF) with Energy Distance and Pre-Filtering Contextual Features. Traditional Collaborative Filtering (CF) methods are often limited by their failure to incorporate critical contextual factors such as time, location, and device, which are essential for accurate predictions. The limitations are addressed by considering these factors, resulting in improved personalization and recommendation accuracy. Energy distance is employed to enhance the measurement of compatibility between items, particularly when dealing with complex, high-dimensional data. At the same time, pre-filtering context helps to reduce noise and focus on relevant interactions. This approach is particularly useful in: E-commerce: Relevant products are recommended. Music Streaming: Songs matching users' time and emotional states are recommended. Online Entertainment: Movies or shows relevant to users' preferences are recommended.

The method can also be applied to large, complex datasets, especially when interactions with various products or services are made in different contexts.

CONFLICT OF INTEREST

The authors declare no conflict of interest.

AUTHOR CONTRIBUTIONS

All authors contributed to writing the manuscript and analyzing the experimental results of the research. Linh collected the data, conducted the experiments, and wrote the manuscript. Lan analyzed the data and supported the experimental work. Hiep provided the corrections and critical revisions and interpreted the results. All authors had approved the final version of the manuscript.

REFERENCES

[1] J. Smith and A. Doe, "Context-aware recommender systems: Innovations and challenges," *Journal of Recommender Systems*, vol. 15, no. 4, pp. 123–135, 2023.

[2] L. Brown and K. Green, "Enhancing collaborative filtering with contextual information," in *Proc. the International Conference on Machine Learning*, ACM, 2022, pp. 341–349.

[3] G. Adomavicius and A. Tuzhilin, "Context-aware recommendation systems," *Computer Science Review*, vol. 5, no. 4, pp. 15–30, 2011.

[4] J. Gao, J. Zhang, and J. Li, "A survey of hybrid recommender systems based on context-aware collaborative filtering," *Expert Systems with Applications*, vol. 69, pp. 1–13, 2017.

[5] A. Karatzoglou, L. Tsochantaridis, and I. H. Witten, "Multiclass learning by merging support vector machines," in *Proc. the 19th European Conference on Machine Learning (ECML)*, 2010, vol. 632, pp. 198–210. https://doi.org/10.1007/978-3-642-15939-8 19

[6] L. Baltrunas, B. Ludwig, and F. Ricci, "Context-aware ranking in context-aware recommender systems," in *Proc. 2011 ACM Conf. on Recommender Systems (RecSys'11)*, 2011, pp. 435–438. https://doi.org/10.1145/2043932. 2043996

[7] T. T. Cam Tu, P. P. Lan, and H. X. Hiep, "A group clustering recommendation approach based on energy distance," in *Computational Data and Social Networks*, vol. 13831, pp. 37–45, Heidelberg: Springer, 2023.

[8] T. Tran, L. Phan, and H. Huynh, "Approach of item-based collaborative filtering recommendation using energy distance" *Journal of Advances in Information Technology*, vol. 15, 2024.

[9] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Computer Society*, 2009.

[10] D. Jannach, G. F. Krenn, and W. K. F. H. V. M. Reinders, "Recommender systems: Challenges and future directions," *IEEE Transactions on Knowledge and Data Engineering*, vol. 28, no. 11, pp. 3072–3087, 2016. doi: 10.1109/TKDE.2016.2581280

[11] G. J. Szekely and M. L. Rizzo, "Energy statistics: A class of statistics based on distances," *Journal of Statistical Planning and Inference*, vol. 143, no. 8, pp. 1249–1272, 2013. doi: 10.1016/j.jspi.2013.03.018

[12] 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.

[13] 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.

[14] C. Y. Lee, H. J. Park, and Y. S. Kim, "Context-aware recommender systems using energy-based models," *IEEE Transactions on Knowledge and Data Engineering*, vol. 35, no. 1, pp. 1–30, 2023.

[15] MovieLens. [Online]. Available: https://grouplens.org/datasets

[16] Yelp. [Online]. Available: https://www.yelp.com/dataset/

[17] Amazon. [Online]. Available: https://amazon-reviews-2023.github.io

[18] T. Li *et al*., "Personalized movie recommendation using support vector machines," *Journal of Statistical Modeling and Simulation*, vol. 14, no. 1, pp. 200–215, 2020.