



**ARTICLE**

# Enhancing Multicriteria-Based Recommendations by Alleviating Scalability and Sparsity Issues Using Collaborative Denoising Autoencoder

S. Abinaya\* and K. Uttej Kumar

School of Computer Science & Engineering, Vellore Institute of Technology, Chennai, Tamilnadu, 600 127, India

\*Corresponding Author: S. Abinaya. Email: s.abinaya@vit.ac.in

Received: 27 October 2023 Accepted: 05 December 2023 Published: 27 February 2024

## ABSTRACT

A Recommender System (RS) is a crucial part of several firms, particularly those involved in e-commerce. In conventional RS, a user may only offer a single rating for an item—that is insufficient to perceive consumer preferences. Nowadays, businesses in industries like e-learning and tourism enable customers to rate a product using a variety of factors to comprehend customers' preferences. On the other hand, the collaborative filtering (CF) algorithm utilizing AutoEncoder (AE) is seen to be effective in identifying user-interested items. However, the cost of these computations increases nonlinearly as the number of items and users increases. To triumph over the issues, a novel expanded stacked autoencoder (ESAE) with Kernel Fuzzy C-Means Clustering (KFCM) technique is proposed with two phases. In the first phase of offline, the sparse multicriteria rating matrix is smoothened to a complete matrix by predicting the users' intact rating by the ESAE approach and users are clustered using the KFCM approach. In the next phase of online, the top-N recommendation prediction is made by the ESAE approach involving only the most similar user from multiple clusters. Hence the ESAE\_KFCM model upgrades the prediction accuracy of 98.2% in Top-N recommendation with a minimized recommendation generation time. An experimental check on the Yahoo! Movies (YM) movie dataset and TripAdvisor (TA) travel dataset confirmed that the ESAE\_KFCM model constantly outperforms conventional RS algorithms on a variety of assessment measures.

## KEYWORDS

Recommender systems; multicriteria rating; collaborative filtering; sparsity issue; scalability issue; stacked-autoencoder; Kernel Fuzzy C-Means Clustering

## 1 Introduction

Over recent decades, Recommender Systems (RS) have grown to be a pivotal solution in tackling the issues of the overwhelming amount of information, particularly within the dynamic landscape of e-commerce. These systems serve the essential function of providing personalized suggestions, encompassing a wide array of items and services, to individual users. This is achieved through the intricate analysis of users' diverse information sources, including their explicit ratings, detailed reviews, historical purchasing patterns, and even implicit behavioral cues. Recommendation systems often employ content-driven and cooperation-driven filtering approaches. In the realm of content-driven filtering, the technique revolves around delving into the preferences and attributes of the



present item, along with the historical choices of the active user [1]. This method essentially tailors' recommendations based on the intrinsic qualities of items and the user's demonstrated inclinations, creating a personalized tapestry of suggestions. On the other hand, the landscape of collaborative filtering (CF) has emerged as a potent force within the e-commerce industry, amassing notable achievements [2]. This method hinges on the power of collective wisdom, orchestrating recommendations by extrapolating from the user's historical rating interactions. By identifying patterns in user behavior, CF techniques adeptly uncover latent connections among users with analogous preferences. This enables them to proficiently recommend items that remain unexplored within a user's history but are anticipated to resonate with their tastes. Within the realm of collaborative filtering (CF) algorithms, we find a crucial categorization: Memory-focused CF and Model-centric CF. Memory-driven CF, in particular, operates by recognizing akin users or comparable items concerning the current user (or a designated target item). These proximate entities' inclinations then serve as valuable inputs in shaping the recommendations provided [3]. In contrast, Model-based CF methodologies, celebrated for their enhanced precision, delve deeper by comprehensively comprehending users' and items' inherent attributes. This adeptness is cultivated during the model development phase, facilitated by the deployment of sophisticated machine-learning techniques. This encompasses the strategic use of methodologies like matrix factorization [4], factorization machines [5], and deep neural networks [6–9], all synergistically working to grasp and internalize users' nuanced preferences.

The previously mentioned methodologies have conventionally been applied to solitary grading schemes. But depending only on these prove inadequate in capturing the diverse spectrum of user feedback, particularly in multifaceted service sectors like restaurants, hotels, and movies. In such contexts, users' experiences are multidimensional and cannot be fully encapsulated through a single rating. In multi-criteria recommender systems, a more comprehensive approach is adopted. These systems enable users to provide feedback across various criteria, offering a nuanced evaluation of an item. For instance, within the realm of restaurants, a user might furnish ratings for distinct attributes such as taste, hygiene, ambiance, hospitality, and price, supplementing an overall rating. This augmentation of feedback affords multi-criteria recommender systems [10] a richer pool of information to draw upon. Consequently, these systems excel in suggesting items that align more accurately with users' multifaceted preferences in comparison to their single-rating counterparts.

Several researchers have ventured into incorporating multi-criteria rating information within their recommendation frameworks [11,12]. However, the potential of applying diverse machine learning techniques to multi-criteria recommender systems remains largely untapped. This study introduces an innovative utilization of a deep learning method known as Stacked Autoencoder (SAE) [13]. In response to the intricacies of multi-criteria rating systems, we present an extended version of SAE that is tailored to address their unique demands. To adapt this technique effectively, modifications are implemented to both the interface component of the traditional network and the objective function. These adjustments are strategically introduced to simplify the comprehension of complex associations between multi-criteria evaluations and their respective aggregate ratings.

As RS is made to aid users in navigating through an enormous number of items, one of its main objectives is to scale up to actual datasets [14]. Conventional collaborative filtering techniques may encounter significant scalability challenges as the volume of items and users grows, leading to computational demands that surpass practical or acceptable thresholds [15]. User clustering via affinities in their user profiles is a popular technique for making recommender systems more scalable and reducing their time complexity. The intricacy of suggestions is only dependent on the size of the cluster when made by cluster representatives for the remaining cluster members [16,17]. Several model-based strategies [18] have been presented before addressing scalability and sparsity challenges

in recommender systems. To realize the full potential of recommender system research, it is necessary to comprehend the most common methodologies applied to directly construct recommender algorithms or to pre-process recommendation datasets, as well as their advantages and disadvantages. As a result, our proposed methodology surpasses the performance benchmarks set by both contrasting single-score recommendation platforms with cutting-edge scalable recommendation systems utilizing diverse criteria.

In brief, this research offers the following major contributions:

- \_ The novel ESAE\_KFCM has 2 offline stage, the sparse user-item multicriteria rating matrix undergoes a smoothing process to make the matrix complete without sparsity by estimating an intact rating of the item using an expanded stacked autoencoder that is not possible in a conventional autoencoder.
- \_ To attempt the problem of scalability and to generate the most excellent recommendations, a Kernel Fuzzy C-Means Clustering (KFCM) is utilized to cluster highly correlated users. Thereby, highly relevant users from multiple clusters alone are utilized in the online phase prediction of intact rating using the ESAE\_KFCM approach that minimizes the time taken for recommendation generation and increases the prediction accuracy.
- \_ Extensive experiments on real-world multicriteria data sets from the Yahoo! Movies (YM) movie dataset and TripAdvisor (TA) travel dataset confirm that ESAE\_KFCM outperforms the traditional recommender systems.

The structure of the work is as follows: The related work of the multicriteria-based recommender system is reviewed in [Section 2](#). In [Section 3](#), the proposed ESAE\_KFCM approach is presented. The specifics of the experimental evaluation and results in terms of prediction accuracy and computation time are discussed in [Section 4](#). Finally, the conclusion and future work of the paper is discussed in [Section 5](#).

## 2 Related Work

Among the pioneering endeavors in the domain of recommendation systems employing multiple aspects is the work by Adomavicius et al. [10]. In their approach, they harnessed statistical methodologies to formulate an aggregation function that bridges the gap between diverse evaluative factors and the comprehensive overall assessment. This entails predicting multi-criteria ratings through conventional techniques [19,20] and subsequently utilizing the aggregation function to derive the overall rating, a process integral to the recommendation process. However, it is worth noting that this approach adheres to an established predilection for parameters consistently among all individuals. Lu et al. [21] presented a hybrid approach that amalgamates content-driven filtering and item-centric collaborative recommendation methods. This hybrid approach was devised to address the challenges posed by the cold-start problem and sparsity within the context of recommender systems employing multiple criteria. Nilashi has made significant contributions to the realm of multi-dimensional recommendation systems through a series of works, as evidenced in [22] and [23]. In [22], Nilashi et al. introduced algorithms rooted in fuzzy logic to enhance the precision of multi-dimensional recommendation systems. Continuing their exploration in [23], Nilashi et al. introduced a blended approach that utilizes the Ant K-means clustering technique. In this structure, essential elements are derived from each group of users through Principal Component Analysis. This information is then employed to facilitate the learning of the sequence linking global assessments and the harvested elements using Support Vector Regression (SVR). Zheng [12] introduced a set of three distinct approaches known as the ‘criteria chain.’ In the first approach, the prediction process commences with the initial criteria

rating, which then serves as contextual information while forecasting the subsequent criteria rating, and so forth. Ultimately, SVR is employed to anticipate the complete assessment based on the sequence of criteria evaluations in addition to Preference learning methodology [24]. The alternative method involves employing Context-Aware Matrix Factorization (CAMF) [25] for predicting the eventual rating by incorporating the ratings of criteria as contextual hints. In the sequential approach, each evaluation factor is autonomously projected, and these anticipated assessments subsequently contribute as background information when predicting the overall assessment. However, relying on sequentially predicted criteria ratings could potentially lead to a cumulative loss in predictive accuracy when estimating the final overall rating [26–31]. Table 1 provides related works, offering insights into various algorithms and methodologies.

**Table 1:** Summary of additional related works and algorithms

Work	Algorithms used	Inference
Zhang et al. [19]	Pair of Probabilistic Latent Semantic Analysis (PLSA) models	PLSA models tend to converge towards local minima, limiting optimization efficiency.
Liu et al. [20]	T-test on users' ratings data to identify significant criteria	Assumes a Gaussian distribution, impractical for sparse rating datasets.
Jannach et al. [11]	Support Vector Regression (SVR)	Comprehends the complex interaction between global assessments and individual evaluation scores, aligning with the framework introduced by Adomavicius and Kwon.
Shu et al. [32]	Specific Class Center Guided Deep Hashing (SCCGDH)	Proposes SCCGDH to learn specific class centers from neural networks and guide hashing learning for multimedia data. Uses networks to reduce intraclass variation and achieve inter-modal invariance, outperforming other hashing approaches.
Shu et al. [33–35]	RSMFH (Matrix Factorization), ROHLSE (Online Hashing with Label Semantic Enhancement)	Introduces RSMFH for multimodal data, maintaining shared and specific properties via matrix factorization, enhancing discriminative ability. Proposes ROHLSE address label noise in online hashing, utilizing low-rank and sparse constraints, achieving superior results in cross-modal retrieval tasks.
Ramesh et al. [34]	Skill Level Navigation Patterns-Mining, CF, Content-Based Filtering	Addresses information overload in programming online judges (POJs) and proposes RS based on learners' ability level navigation patterns. Outperforms other approaches in practice problem recommender systems.

(Continued)

**Table 1 (continued)**

Work	Algorithms used	Inference
Sreepada et al. [24]	Preference learning methodology	Addressed individual preferences and common criteria associated with each element. Incorporates user-centered and product-focused collaborative filtration (CF).
Sinha et al. [26]	Social spider optimization, matrix factorization, neural networks	Integrates various techniques for recommendation systems but faces challenges in prediction accuracy and scalability.
Shu et al. [36]	Discrete Asymmetric Zero-Shot Hashing (DAZSH)	Integrates pairwise similarity, class attributes, and semantic labels for zero-shot hashing learning. Efficient discrete optimization strategy. Achieves promising results in cross-modal retrieval tasks.
Kannimuthu et al. [37,38]	Stacked CNN, BiLSTM, Conditional GAN	Both studies leverage advanced techniques – one for social media author profiling and the other for ASD prediction using neuroimaging, showcasing the authors' expertise in diverse applications of deep learning.
Wu et al. [39] & Gao et al. [40]	Graph Neural Networks (GNN)	A comprehensive survey on GNNs in recommender systems, categorizing models, addressing challenges, and providing insights for future development.
Gao et al. [41]	Large Language Models (LLMs)	Proposes Chat-Rec, an innovative approach to conversational recommender systems, leveraging LLMs for improved interactivity, explainability, and cross-domain recommendations.
Yang et al. [42]	Sequence-to-sequence paradigm, Prompt-based learning strategies	Proposes UniMIND, a unified framework for MG-CRS that addresses Goal Planning, Topic Prediction, Item Recommendation, and Response Generation.

### 3 Proposed Method

The architectural view of the proposed ESAE\_KFCM approach is depicted in Fig. 1. As specified in Fig. 1, the smoothening approach in the phase-I offline task utilizes a novel expanded stacked autoencoder that combines users' multicriteria-based ratings given for the specific item to predict overall or intact rating value. Thus, the sparse matrix of multicriteria ratings is made into a filled matrix of intact ratings that yields an efficient way to predict the most relevant time and identify the highly correlated user. To perform user modeling, the Kernel Fuzzy C-Means Clustering (KFCM) approach is utilized to cluster the correlated users. In the online phase, the active user is recommended with the top-N most relevant item by the based rating prediction approach. The overall process of the proposed ESAE\_KFCM approach is given in Algorithm 1.

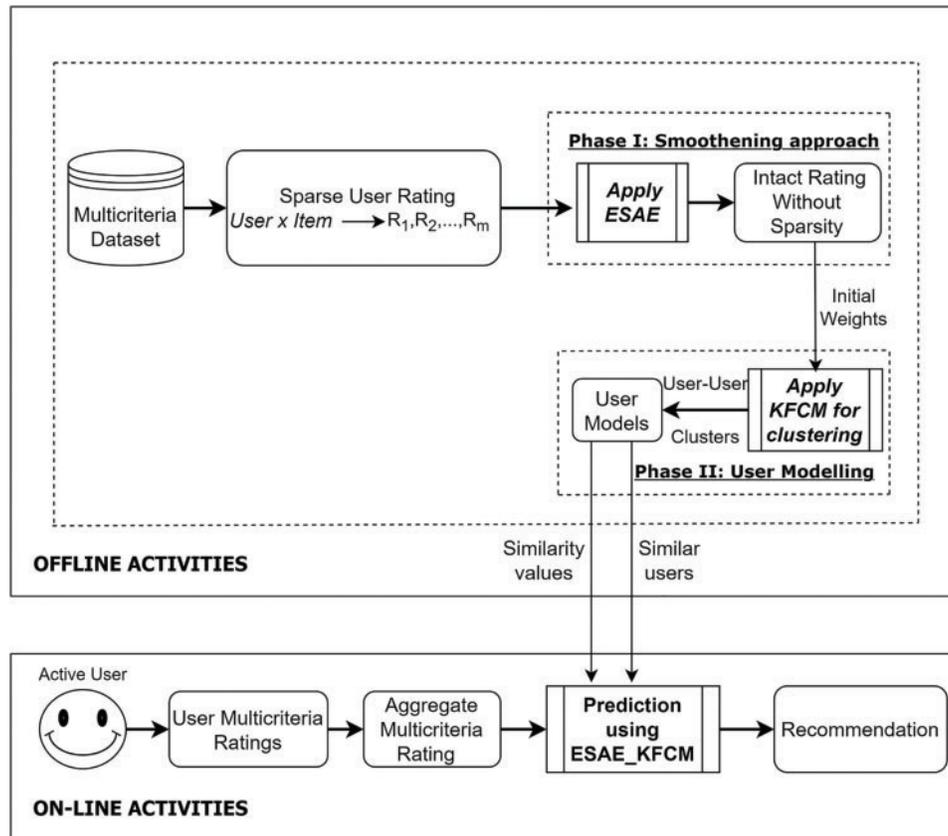


Figure 1: Proposed approach using ESAE\_KFCM

---

**Algorithm 1:** Proposed system using ESAE\_KFCM

---

**Input:** Sparse matrix  $R_t$  of size  $[I \times K]$  with multicriteria ratings  $\langle r_{nm}^1, r_{nm}^2, \dots, r_{nm}^q \rangle$  that contains a collection of  $I$  users  $\{U_n | n = 1, 2, \dots, I\}$  and collection of  $K$  diverse items  $\{I_m | m = 1, 2, \dots, K\}$ , Active user  $RV_{au}$

**Output:** Top- $N$  Items recommendation for  $RV_{au}$ .

---

**Method:**

1. The offline phase of ESAE\_KFCM has the following steps,
    - 1.1. Smoothen sparse  $R_t$  multicriteria matrix into completely intact  $\hat{R}_t$  rating matrix by ESAE approach as given in Section 3.1.1.
    - 1.2. Perform user modeling using the KFCM clustering approach as mentioned in Algorithm (2).
  2. The online phase of the ESAE\_KFCM for each  $RV_{au}$  has the following steps,
    - 2.1. Perform the Recommendation process as mentioned in Algorithm (3).
- 

The following data has been provided in a transaction database:

1. A collection of  $I$  users  $\{U_n | n = 1, 2, \dots, I\}$ .
2. A collection of  $K$  diverse items  $\{I_m | m = 1, 2, \dots, K\}$ .

3. A table of rating  $R_t$  of size  $[I \times K]$  that contains data on the users' history of multicriteria ratings of items  $o = \{n, m, r_{nm}^1, r_{nm}^2, r_{nm}^3, \dots, r_{nm}^q\}$  where  $r_{nm}^1$  implies the rating value for item  $m$  by user  $n$  under criteria  $1$  and  $q$  represents the total number of criteria. A zero is used to indicate the rating value of unavailable items.

### 3.1 Off-Line Task of the ESAE\_KFCM

Two phases of the task are carried out offline, specifically, Phase I performs Smoothing, and User modeling is done in Phase II.

#### 3.1.1 Smoothing-Expanded Stacked Denoising Autoencoder (ESAE) (Phase-I)

Here, the stacked denoising autoencoder is expanded to include a multicriteria of ratings given by the user for an item in the network. Consequently, the improved loss function is incorporated accordingly [27]. The user-item explicit preferences in the matrix of sparse multicriteria ratings are computed to a dense matrix of overall ratings using ESAE.

The count of neurons that comprise the first layer of a typical autoencoder is similar to the count of parameters in the input. The fact from each attribute is supplied to the appropriate neurons within the layer of input. But in the case of a multi-criteria context, any product will have multiple quality ratings that must be fed inside the network. As a consequence, the autoencoder was expanded by having an add-on layer that encompasses multi-criteria ratings. Particularly, denoising AE aims to prevent the latent layer from just learning the identification function and to compel it to find more robust features. Therefore, the expanded add-on layer serves as a first input layer that is corrupted with the addition of Gaussian noise and is subsequently connected to the intermediary layer containing item nodes. It is indeed connected to the successive  $T$  encoding layers to uncover the items' hidden representation.

The final layer of encoding is allied to the  $T$  layers of decoding, which are employed to interpret the latent features learned from corresponding encoders. The items' exact specific ratings are predicted as an output in the final decoding layer.

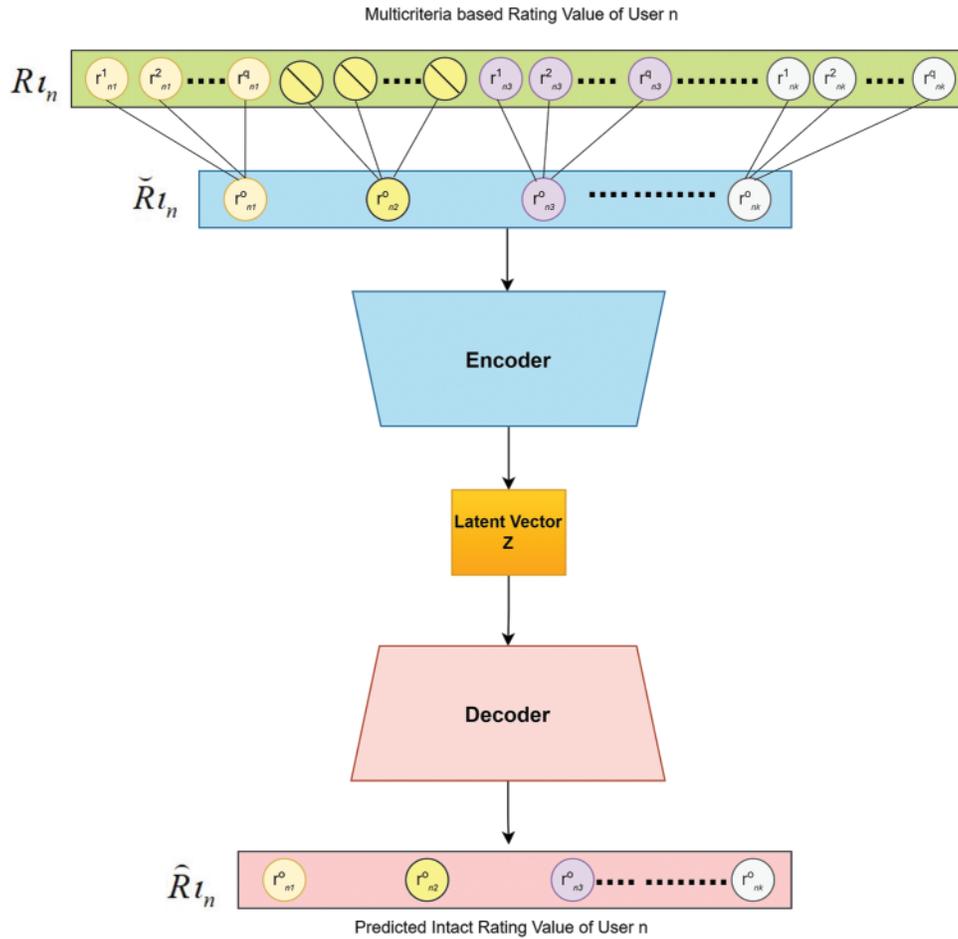
Let  $r_{nm}^o$  be the optimum rating for an item  $m$  given by user  $n$  which is attained from  $r_{nm}^1, r_{nm}^2, r_{nm}^3, \dots, r_{nm}^q$  respectively as mentioned in Eq. (1).

$$r_{nm}^o = r_{nm}^1 \times w_m^1 + r_{nm}^2 \times w_m^2 + r_{nm}^3 \times w_m^3 + \dots + r_{nm}^q \times w_m^q \quad (1)$$

where  $w_m^1, w_m^2, w_m^3, \dots, w_m^q$  are the multicriteria weights of the  $m^{\text{th}}$  item. These unified optimum values of rating are given to the subsequent layers of the ESAE model. Finally, a justifiable and intact preference value  $\hat{R}_t$  is predicted as in Eq. (2) by a completely connected network as depicted in Fig. 2.

$$\hat{R}_t = \rho (G' (\rho (G \{r_m^o\} + s)) + s') \quad (2)$$

where  $\hat{R}_t \in \mathbb{R}^K$  is the final predicted layer of intact rating value,  $G \in \mathbb{R}^{H \times K}$  and  $G' \in \mathbb{R}^{1 \times H}$  are the weight matrices,  $r_m^o \in \mathbb{R}^K$  is the intact rating value of item  $m$  in the sparse  $\hat{R}_t$ ,  $s \in \mathbb{R}^H$  and  $s' \in \mathbb{R}^K$  is the bias vectors and function  $\rho$  is the hyperbolic tangent.



**Figure 2:** Structure of expanded stacked autoencoder (ESAE)

The ESAE model is trained using the updated loss function below in Eq. (3) to gain a concise representation, since predicting the loss rate for the zero values in the rating vector  $R_{t_n}$  results in obsolete. To prevent overfitting of the proposed ESAE model, the objective function combines regularization terms and is given away as in Eq. (3).

$$L(\hat{R}_t, \check{R}_t) = \sum_{j \in R_{t'}} \left( ESAE(\hat{R}_t)_j - \check{R}_{t_j} \right)^2 + \beta (\|G\|_F^2 + \|G'\|_F^2 + \|w\|_F^2 + \|s\|_F^2 + \|s'\|_F^2) \quad (3)$$

where  $\hat{R}_t$  indicates an absolute array of intact ratings where  $\hat{R}_t = R_{t''} \cup R_{t'''}$ ,  $R_{t''}$  comprises of known intact rating instances,  $R_{t'''}$  comprises of the unknown instances,  $\|G\|$ ,  $\|G'\|$  are the vectors of weight of the ESAE model,  $\|w\|$  indicates the vectors of the weight of multicriteria rating of respective item, vectors of the weight of bias are  $\|s\|$ ,  $\|s'\|$  and  $\beta$  is the regularization hyperparameter that controls how much learning has an impact on how generalizable the model is, so it is important to determine the right value, and  $\beta$  is defined by the outcomes of the experiment. The parameters are updated using the ADAM optimizer, a version of stochastic gradient descent. In an attempt to overwhelm the identity network, regularized dropouts via a probability  $p$  are additionally introduced at all layers. The

forecasted instance  $j$ 's intact rating is  $ESAE (R'_j)$  and  $\check{R}_{l_j}$ , the real overall rating of instance  $j$ . From Eq. (3), it could be seen that the loss is computed using only the available rating instance set ( $R'$ ).

Now the rating table  $\hat{R}_l$  is a completely smoothed matrix of users' intact rating values for all the items that have been predicted from multicriteria ratings given for the item by the user. By this ESAE approach the accuracy of the intact rating prediction is improved which enhances Top-N recommendation.

### 3.1.2 User Modeling-Kernel Fuzzy C-Means Clustering (KFCM) (Phase-II)

Towards dealing with the scalability issue, the smoothed matrix  $\hat{R}_l$  using the ESAE approach as explained in Section 3.1.1 is clustered before the task of recommendation. This task is altered for ESAE\_KFCM. The total amount of ratings given by the active user during the phase of recommendation is incredibly low. Prevalent clustering techniques like expectation maximization, similarity-based clustering, and k-means project the data over a scale of rating space that may be significantly larger. These clustering methods have the limitation that their prototypes are located in a highly-dimensional rating space, consequently, the descriptions are inadequate and informative (sparse user-item matrix). As a result, Zhang et al. [28] introduced a kernel fuzzy approach for clustering that is utilized in user modeling in ESAE\_KFCM. The investigation demonstrates that fuzzy clustering is more resistant to outliers and noise than other approaches and can accept unequally sized clusters. In the phase of online, the most relevant users' multicriteria rating vector alone is retrieved from the multiple clusters and the accurate unknown ratings are predicted using the ESAE approach with minimal recommendation time. The procedure of clustering is elucidated in Algorithm 2.

---

#### Algorithm 2: Clustering by KFCM approach

---

*Input:*  $\hat{R}_l$  is the smoothed user  $\times$  item matrix,  $\hat{R}_{l_n} \in \mathbb{R}^k$  indicates the intact rating vector of user  $n$ , and  $f$  is the number of clusters.

*Output:* Center of the cluster  $C_b$ , membership degree  $md_{nb}$  for cluster  $b$  and user  $n$  (Every user holds a membership value ranging from 0 to 1, indicating the user's  $n$  desire in every cluster  $b$ ).

---

*Method:*

1. Set the membership degree  $md_{n,b} = randperm(b) / \sum (randperm(b))$  and initialize  $i = 1$ .
2. Compute the center of the cluster  $C_b$  ( $1 \leq b \leq f$ ) as mentioned in Eq. (4).

$$C_b = \sum_{n=1}^I md_{n,b}^i \cdot \hat{R}_{l_n} / \sum_{n=1}^I md_{n,b}^i \quad (4)$$

3. Compute membership  $md_{n,b}$  ( $1 \leq n \leq I, 1 \leq b \leq f$ ) as mentioned in Eq. (5).

$$md_{n,b} = 1 / \sum_{a=1}^f \left( \frac{\hat{R}_{l_n} - C_b}{\hat{R}_{l_n} - C_a} \right)^{\frac{2}{a-1}} \quad (5)$$

4. Compute  $Error = \max (oldmd_{n,b}, newmd_{n,b})$ .
  5. If  $Error > 0.5$ , then  $f = f + 1$ ; go to Step 3.
-

### 3.2 Recommendation Process by ESAE\_KFCM in Online Phase

During the stage of the online recommendation process, the active user is recommended depending on the most correlated users obtained from multiple clusters generated by Section 3.1.2. This process of online recommendation is explained in Algorithm 3.

---

#### Algorithm 3: Recommendation by ESAE\_KFCM

---

*Input:* Rating Vector of multicriteria ratings of active user  $RV_{au}$ , Total item number to be recommended  $N$ , Center of the cluster  $C_b$ .

*Output:* Item recommendation.

---

*Functional Specification:*

1. In case the active user is someone new where  $RV_{au}$  remains NULL (cold start new user), provide the top N-rated items in all clusters to the recommending agent.
  2. In case the active user has a sparse rating vector and is new to the system, perform steps 2.1 to 2.3. to provide the top N predicted items to the recommending agent.
    - 2.1. Compute membership degree  $md_{au,b}$  for active user  $au$  with the existing clusters using Eq. (5).
    - 2.2. Aggregate by finding the average of multi-criteria rating and retrieve the most similar users' multicriteria rating vector from multiple clusters.
    - 2.3. Estimate the active user  $au$ 's, unknown ratings using the ESAE approach as mentioned in Section 3.1.1., only using the most similar user obtained from the multiple clusters.
  3. In case the active user already exists in the training data and no modification in  $RV_{au}$ , provide the top N estimated rating of items in the training phase to the recommending agent.
  4. In case the active user already exists in training data and with added ratings, carry out the steps mentioned in Step 2.
- 

## 4 Experimental Evaluation

### 4.1 Dataset Description and Evaluation Metrics

The proposed ESAE\_KFCM technique is tested using two real-time multicriteria data sets from the Yahoo! Movies (YM) movie sector dataset [11] and TripAdvisor (TA) [29] travel dataset sectors. The Yahoo Movies dataset has 6,078 users who rated a total of 976 movies has 62,156 ratings between 1 and 13. Each user assessed a movie based on 4 different criteria: Visuals, Direction, Acting, and Story. For experimenting, this 13-level scoring is converted to the standard 5-point format. Through web scraping on the Trip Advisor website with Beautiful Soup, a total of 60,216 records were gathered from 2500 hotels located in 93 different cities. Each hotel may be assessed based on multiple criteria such as service, value, cleanliness, rooms, staff, and quality of sleep. We sustain instances of users whose rating is at least five hotels and hotels that had at least five user ratings to extract the workable subset of data from TA. The extracted subset TA 5-5 has sparse data of 99.83% for 3550 hotels with 3160 users and 9374 instances of ratings. Similarly, YM 20-20, YM 5-5, and YM 10-10 subsets of data are extracted.

To assess the efficacy of the ESAE\_KFCM approach, we employed prominent performance measures like, F1 score, Mean Absolute Error (MAE) [30], Good Predicted Items MAE (GPIMAE), Good Items MAE (GIMAE) [31], and computation time. Metrics for prediction accuracy, such as Mean Absolute Error (MAE), specified in Eq. (6), compare the actual ratings with the expected ratings.

$$MAE = \sum_{n=1}^I \sum_{m=1}^K \frac{|A_{n,m} - E_{n,m}|}{I * K} \quad (6)$$

where  $I, K, A_{n,m}, E_{n,m}$  denotes the total user, total items, and actual, and expected ratings of user  $n$  for item  $m$ . High recall and precision levels are ideal for a model. Recall typically decreases as precision increases, and vice versa, due to the inverse correlation. To take into account recall ( $\mathcal{R}$ ) and precision ( $\mathcal{P}$ ), the metric  $F_1$  is defined in Eq. (7):

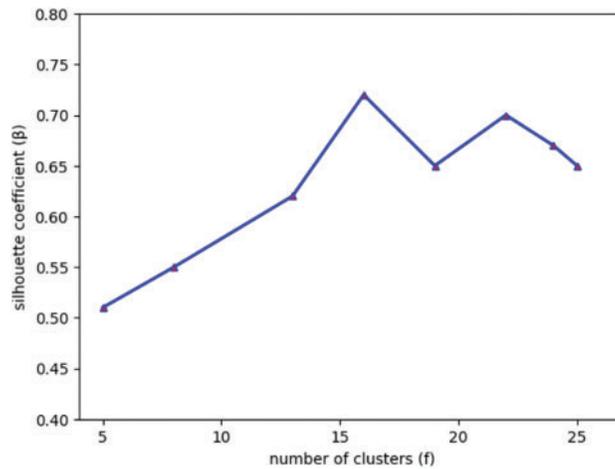
$$F1 = 2\mathcal{P}\mathcal{R}/\mathcal{P} + \mathcal{R} \quad (7)$$

GPIMAE and GIMAE calculate the MAE in the system's prediction of acceptable items and in those items, it forecasts to be good. As a result, they concentrate just on pertinent things rather than considering every item in the assessment subset. Therefore, the primary benefit of these measures is that they assess the algorithm solely for predictions that are pertinent to the user, i.e., for items that either ought to or will get placed on the list of recommendations.

#### 4.2 Training Settings

The datasets TA 5-5, YM 20-20, YM 5-5, and YM 10-10 are divided into training data (60%), cross-validation (20%), and test data (20%) for the experiment. The proposed ESAE\_KFCM paradigm is trained using Keras API in Python from the TensorFlow package. Utilizing the sci-kit-learn's GridSearchCV feature, we investigate the potential effects of several distinct parameter values on the accuracy of the ESAE\_KFCM model. Specifically, noise variance is altered with [0.1,0.2,0.3,0.4,0.5], number of hidden units [500,450,350,300,600], optimization technique as ['Nadam', 'Adamax', 'Adam', 'Adadelta', 'Adagrad', 'RMSprop', 'SGD'], epoch with [100,200,240,280,300] and learning rate are assessed. Whenever the value of epoch value is 280, the ESAE model gets converged with corruption ratio = 0.4 and hidden units 450. Learning rates with altering values (0.002, 0.001, 0.01, 0.02, and 0.03) were tested and exposed that the learning rate  $\eta = 0.02$  is given better accuracy. Similar to this, a minibatch of size 50 is used to optimize the network using a Stochastic Gradient Descent Optimizer. Also, weight decay  $l_2(0.001)$  is introduced for regularization, we chose hyperbolic tangents as a transfer function.

In this ESAE\_KFCM technique, the optimum number of clusters ( $f$ ) is found using the Silhouette Coefficient Method [43]. The max value of the silhouette coefficient ( $\beta$ ), which indicates that the number of clusters formed is ideal, is 1. The first step in the experiment to find the optimum value for  $f$  is choosing the number of clusters to be assessed, which can range from 5 to 25. Based on the smallest possible clusters, the minimum  $f = 5$ . In the interim, the maximum  $f = 25$ . The value of  $\beta$  gives the second-highest optimum value at  $f = 22$  and then starts to decline at  $f = 25$ . Since the higher  $f$  requires greater computing time, the subsequent  $f$  values do not proceed. The findings of the clustering assessment via the Silhouette Coefficient approach are displayed in Fig. 3. The ideal  $f$  value is indicated in Fig. 3, where the  $f = 16$  yields the greatest value of the Silhouette Coefficient. As a result, the process of clustering will start with some clusters  $f$  equal to 16.



**Figure 3:** Results of silhouette coefficient ( $\beta$ )

### 4.3 Results and Discussion

The performance of ESAE\_KFCM is assessed and compared with single and multicriteria rating-based models as baseline methods of recommendation as listed below:

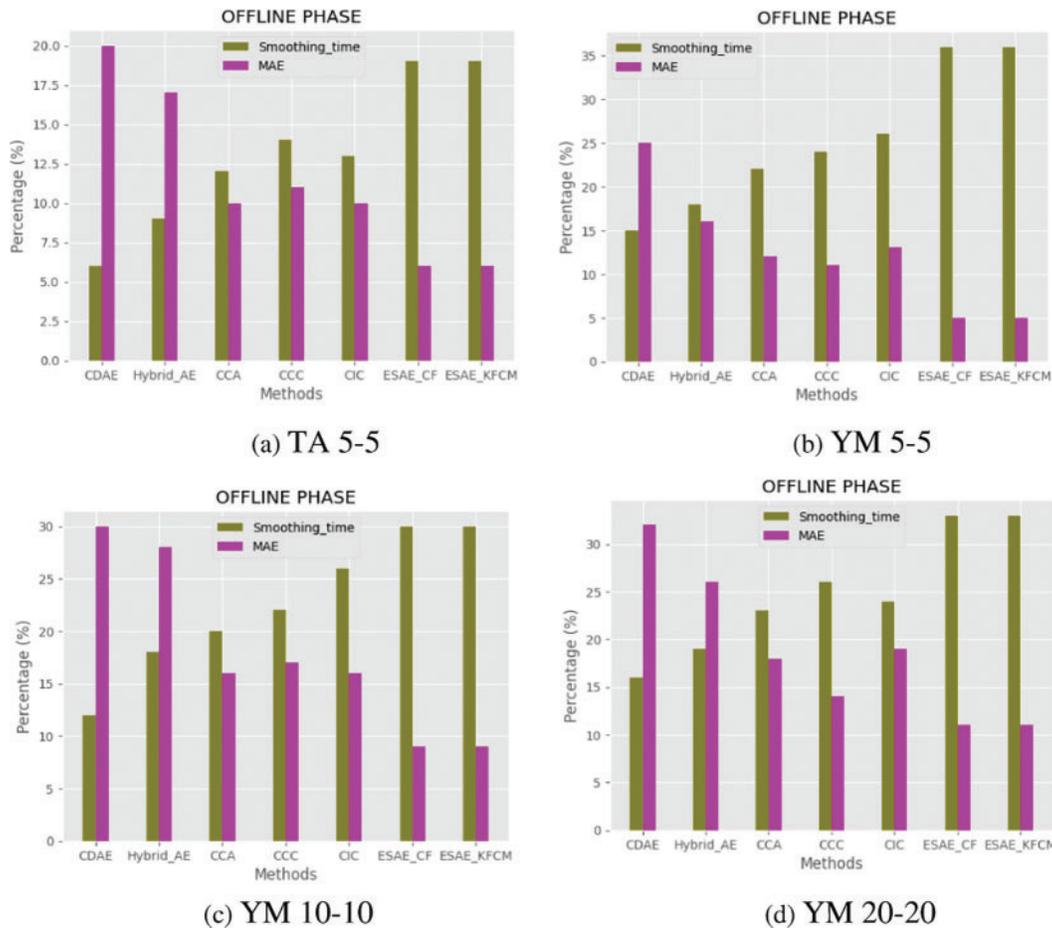
1. CDAE [8]: Collaborative DAE that uses a single rating system that incorporates latent factors for recommendation.
2. Hybrid\_AE [9]: CF-based neural network that uses a single rating system that combines user and item side information.
3. Agg\_CCA [12]: Aggregation approach that utilizes a multicriteria rating system to construct hybrid item and user-specific model.
4. Context\_CCC [12]: Estimates overall preference values based on contextual situations cast by multiple criteria rating system.
5. Ind\_CIC [12]: The BiasedMF algorithm was utilized to estimate the individual rating criteria as contexts by avoiding dependencies between multiple criteria ratings.
6. ESAE\_CF: This approach integrates the proposed ESAE approach with CF technique-based prediction for top-N item recommendation.

#### 4.3.1 Performance Analysis of ESAE-Based Smoothing Process (Offline Phase)

Fig. 4 shows the MAE and smoothing time comparison of the ESAE with conventional models on all the working datasets. The efficiency of the ESAE model's smoothing process in the offline phase is measured in terms of MAE and computational time.

- CDAE and Hybrid\_AE work on a single rating system where the smoothing process is done by autoencoder has average performance in MAE but the smoothing time is comparatively high.
- Hybrid\_AE outperforms the based smoothing technique since it incorporates the gain from side information to avoid the sparsity problem.
- Agg\_CCA and Context\_CCC approaches utilize the CAMF\_C method, which adopts criteria chains to anticipate the various criteria ratings one at a time and has better prediction accuracy than single rating systems such as CDAE and Hybrid\_AE.

- Ind\_CIC outperforms Agg\_CCA and Context\_CCC in terms of prediction accuracy and the smoothing time was comparatively high.
- As multicriteria ratings are considered for the smoothing process by utilizing an expanded stacked autoencoder in the ESAE\_KFCM approach, the modeling time is comparatively elevated whereas outperforms MAE prediction accuracy compared to all the other models.
- The smoothing approach is similar for both ESAE\_CF and ESAE\_KFCM. Subsequently, both have the same performance.

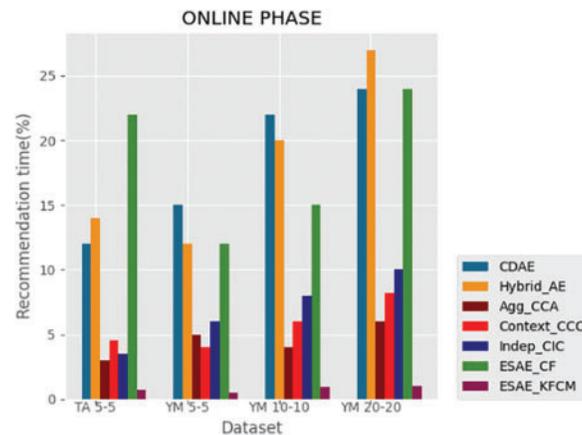


**Figure 4:** MAE comparison and smoothing time of the ESAE

#### 4.3.2 Performance Analysis of ESAE\_KFCM Process (Online Phase)

Subsequently completing the smoothing process, the dataset is clustered by the proposed Kernel Fuzzy C-Means Clustering (KFCM) approach. In the online phase the multicriteria rating vector of highly correlated users' is alone considered for rating prediction using the ESAE approach. Thereby the proposed ESAE\_KFCM approach gives the most relevant Top-N items as recommendations in minimal time and greater prediction accuracy. The time taken to recommend top-N items for the active user with the conventional model is depicted in Fig. 5. Tables 2–5 depict the results of accuracy measures in terms of MAE, F1 score, GPIMAE, and GIMAE, and the observations are listed as follows:

- As depicted in Fig. 5, ESAE\_KFCM constantly outperforms all the conventional models concerning recommendation time. Here the prediction of an intact rating takes only a minimal time for Top-N item recommendation since it inputs only the highly correlated multicriteria rating vector into the ESAE model fetched from multiple clusters in the online phase. Ind\_CIC, Agg\_CCA, and Context\_CCC outperform CDAE and Hybrid\_AE in terms of recommendation time since it works with the principle of dimensionality reduction technique. In ESAE\_CF, the recommendation generation time is high compared to all the other models.
- Ind\_CIC, Agg\_CCA, and Context\_CCC comparatively have 17% higher errors than the ESAE\_KFCM approach in terms of MAE, F1 score, GPIMAE, and GIMAE. Hybrid\_AE outperforms CDAE in terms of prediction accuracy but has comparatively 21% high GPIMAE, and GIMAE. ESAE\_CF outperform all the multicriteria rating-based recommendation system such as Ind\_CIC, Agg\_CCA, and Context\_CCC in terms of F1 score, GPIMAE, and GIMAE for the existing user with modified multicriteria rating vector and new user with no rating vector.
- The KFCM clustering process with some cluster ( $f$ ) = 16 as mentioned in Fig. 4 on all the datasets has upgraded the prediction accuracy of 98.2% in Top-N recommendation with an average online recommendation generation time per item with 0.00428346 s accordingly.
- ESAE\_KFCM has an accurate prediction of intact ratings with less error in terms of GPIMAE, GIMAE, MAE and a 22% efficient F1 score compared to ESAE\_CF, Ind\_CIC, Agg\_CCA, and Context\_CCC in addition to minimized recommendation time and better decision support.



**Figure 5:** Recommendation time evaluation on the dataset

**Table 2:** The performance comparison on the YM 5-5 dataset in the Online phase

Model	F1 score	GPIMAE	GIMAE	MAE
CDAE	0.6491	0.7913	0.5672	0.6306
Hybrid_AE	0.6789	0.8406	0.6022	0.6531
Agg_CCA	0.4497	0.5901	0.5878	0.6737
Context_CCC	0.4826	0.6095	0.6124	0.6914
Indep_CIC	0.4636	0.6814	0.6536	0.7129
ESAE_CF	0.7458	0.5379	0.5210	0.5674
<b>ESAE_KFCM</b>	<b>0.7816</b>	<b>0.5111</b>	<b>0.4993</b>	<b>0.5258</b>

**Table 3:** The performance comparison on the TA 5-5 dataset in the Online phase

Model	F1 score	GPIMAE	GIMAE	MAE
CDAE	0.6545	0.8693	0.6976	0.7806
Hybrid_AE	0.6798	0.8244	0.6314	0.7691
Agg_CCA	0.5640	0.5972	0.5417	0.6891
Context_CCC	0.5380	0.7577	0.6242	0.6888
Indep_CIC	0.5370	0.7439	0.6420	0.7012
ESAE_CF	0.7109	0.5592	0.4636	0.6080
<b>ESAE_KFCM</b>	<b>0.7916</b>	<b>0.5289</b>	<b>0.4407</b>	<b>0.5758</b>

**Table 4:** The performance comparison on the TA 10-10 dataset in the Online phase

Model	F1 score	GPIMAE	GIMAE	MAE
CDAE	0.6196	0.8453	0.6792	0.8284
Hybrid_AE	0.7042	0.8269	0.6595	0.7811
Agg_CCA	0.5343	0.7990	0.6015	0.6618
Context_CCC	0.5361	0.7857	0.6240	0.6374
Indep_CIC	0.5327	0.7743	0.6542	0.6719
ESAE_CF	0.7313	0.5592	0.4870	0.5783
<b>ESAE_KFCM</b>	<b>0.8165</b>	<b>0.5289</b>	<b>0.4512</b>	<b>0.5403</b>

**Table 5:** The performance comparison on the TA 20-20 dataset in the Online phase

Model	F1 score	GPIMAE	GIMAE	MAE
CDAE	0.7200	0.8164	0.6237	0.7541
Hybrid_AE	0.7578	0.7830	0.6008	0.7205
Agg_CCA	0.5641	0.6971	0.6042	0.6691
Context_CCC	0.5585	0.7159	0.6095	0.6798
Indep_CIC	0.5677	0.7064	0.6218	0.7029
ESAE_CF	0.8070	0.6523	0.4870	0.5906
<b>ESAE_KFCM</b>	<b>0.8632</b>	<b>0.6179</b>	<b>0.4324</b>	<b>0.5345</b>

## 5 Conclusion and Future Enhancement

This work developed and estimated a scalable novel ESAE\_KFCM model to enhance Top N recommendation by Kernel Fuzzy C-Means Clustering. In the KFCM technique, the method of clustering efficiently retrieves the highly correlated user from multiple clusters in the recommendation phase. In the offline phase, ESAE ESAE-based smoothing is performed by considering the multicriteria rating vector thereby overcoming the sparsity and enhancing prediction accuracy. Compared with the existing models, the proposed ESAE\_KFCM model can make online recommendations using only the most similar users' multicriteria rating vector retrieved from multiple clusters enhancing the high

quality of Top-N item recommendation with minimal computation time. This model can be enhanced further by involving additional criteria like reviews, and user and item features to improve the accuracy of the recommendation process. Besides, variational or sparse autoencoders can be utilized for further investigation to enhance the Top-N recommendation.

**Acknowledgement:** We thank Vellore Institute of Technology, Chennai, for supporting us with the article processing charge (APC).

**Funding Statement:** The authors received no specific funding for this study.

**Author Contributions:** The authors confirm contribution to the paper as follows: study conception and design: Abinaya S; data collection: Uttej Kumar K; analysis and interpretation of results: Abinaya S, Uttej Kumar K draft manuscript preparation: Abinaya S, Uttej Kumar K. All authors reviewed the results and approved the final version of the manuscript.

**Availability of Data and Materials:** The TripAdvisor datasets analyzed during the current study are available at <https://www.kaggle.com/datasets/andrewmvd/trip-advisor-hotel-reviews>, and Yahoo movies dataset, <https://webscope.sandbox.yahoo.com/> accessed on 17 May 2023.

**Conflicts of Interest:** The authors declare that they have no conflicts of interest to report regarding the present study.

## References

- [1] M. Pazzani and D. Billsus, "Learning and revising user profiles: The identification of interesting web sites," *Mach. Learn.*, vol. 27, pp. 313–331, 1997. doi: [10.1023/A:1007369909943](https://doi.org/10.1023/A:1007369909943).
- [2] G. Linden, B. Smith, and J. York, "Amazon. com recommendations: Item-to-item collaborative filtering," *IEEE Internet. Comput.*, vol. 7, no. 1, pp. 76–80, 2003. doi: [10.1109/MIC.2003.1167344](https://doi.org/10.1109/MIC.2003.1167344).
- [3] Z. Sun, X. Zhang, H. Li, Y. Xiao and H. Guo, "Recommender systems based on tensor decomposition," *Comput. Mater. Contin.*, vol. 66, pp. 621–630, 2020. doi: [10.32604/cmc.2020.012593](https://doi.org/10.32604/cmc.2020.012593).
- [4] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discov. Data. Min.*, Las Vegas, Nevada, USA, 2008, pp. 426–434.
- [5] S. Rendle, "Factorization machines," in *2010 IEEE Int. Conf. Data. Min.*, Sydney, NSW, Australia: IEEE, 2010, pp. 995–1000.
- [6] S. Abinaya and M. K. Kavitha Devi, "Trust-based context-aware collaborative filtering using denoising autoencoder," in *Pervasive. Comp. Soc Netw: Proc. ICPCSN 2021*, Singapore, Springer, 2022, pp. 35–49.
- [7] S. Abinaya, A. S. Alphonse, S. Abirami, and M. K. Kavithadevi, "Enhancing context-aware recommendation using trust-based contextual attentive autoencoder," in *Neural. Proc. Letters.*, 2023, pp. 1–22. doi: [10.1007/s11063-023-11163-x](https://doi.org/10.1007/s11063-023-11163-x).
- [8] Y. Wu, C. DuBois, A. X. Zheng, and M. Ester, "Collaborative denoising auto-encoders for top-N recommender systems," in *Proc. 9th ACM Int. Conf. Web Search Data Min.*, 2016, pp. 153–162.
- [9] F. Strub, R. Gaudel, and J. Mary, "Hybrid recommender system based on autoencoders," in *Proc. 1st Workshop Deep Learn Recommender Syst.*, 2016, pp. 11–16. doi: [10.1145/2988450.2988456](https://doi.org/10.1145/2988450.2988456).
- [10] G. Adomavicius and Y. Kwon, "New recommendation techniques for multicriteria rating systems," *IEEE Intell. Syst.*, vol. 22, no. 3, pp. 48–55, 2007. doi: [10.1109/MIS.2007.58](https://doi.org/10.1109/MIS.2007.58).
- [11] D. Jannach, Z. Karakaya, and F. Gedikli, "Accuracy improvements for multi-criteria recommender systems," in *Proc. 13th ACM Conf. Electron. Commer.*, 2012, pp. 674–689.
- [12] Y. Zheng, "Criteria chains: A novel multi-criteria recommendation approach," in *Proc. 22nd Int. Conf. Intell. User Interfaces*, 2017, pp. 29–33.

- [13] S. Abinaya and M. K. Devi, “Enhancing top-N recommendation using stacked autoencoder in context-aware recommender system,” *Neural. Process. Lett.*, vol. 53, pp. 1865–1888, 2021. doi: [10.1007/s11063-021-10475-0](https://doi.org/10.1007/s11063-021-10475-0).
- [14] A. Kumar and A. Sharma, “Alleviating sparsity and scalability issues in collaborative filtering based recommender systems,” in *Proc. Int Conf Front Intell Comput: Theory Appl. (FICTA)*, Berlin Heidelberg: Springer, 2013, pp. 103–112.
- [15] M. Singh, “Scalability and sparsity issues in recommender datasets: A survey,” *Knowl. Inf. Syst.*, vol. 62, pp. 1–43, 2020. doi: [10.1007/s10115-018-1254-2](https://doi.org/10.1007/s10115-018-1254-2).
- [16] S. Abinaya, K. Indira, S. Karthiga, and T. Rajasenbagam, “Time cluster personalized ranking recommender system in multi-cloud,” *Math.*, vol. 11, no. 6, pp. 1300, 2023. doi: [10.3390/math11061300](https://doi.org/10.3390/math11061300).
- [17] M. K. Devi and P. Venkatesh, “Smoothing approach to alleviate the meager rating problem in collaborative recommender systems,” *Future. Gener. Comput. Syst.*, vol. 29, no. 1, pp. 262–270, 2013. doi: [10.1016/j.future.2011.05.011](https://doi.org/10.1016/j.future.2011.05.011).
- [18] S. Bakshi, A. K. Jagadev, S. Dehuri, and G. N. Wang, “Enhancing scalability and accuracy of recommendation systems using unsupervised learning and particle swarm optimization,” *Appl. Soft. Comput.*, vol. 15, pp. 21–29, 2014. doi: [10.1016/j.asoc.2013.10.018](https://doi.org/10.1016/j.asoc.2013.10.018).
- [19] Y. Zhang, Y. Zhuang, J. Wu, and L. Zhang, “Applying probabilistic latent semantic analysis to multi-criteria recommender system,” *Ai. Commun.*, vol. 22, no. 2, pp. 97–107, 2009. doi: [10.3233/AIC-2009-0446](https://doi.org/10.3233/AIC-2009-0446).
- [20] L. Liu, N. Mehandjiev, and D. L. Xu, “Multi-criteria service recommendation based on user criteria preferences,” in *Proc. Fifth ACM Conf. Recomm. Syst.*, 2011, pp. 77–84. doi: [10.1145/2043932.2043950](https://doi.org/10.1145/2043932.2043950).
- [21] Q. Shambour and J. Lu, “A hybrid multi-criteria semantic-enhanced collaborative filtering approach for personalized recommendations,” in *In 2011 IEEE/WIC/ACM Int. Conf. Web. Intell. Intell. Agent. Technol.*, IEEE, vol. 1, 2011, pp. 71–78.
- [22] M. Nilashi, D. Jannach, O. bin Ibrahim, and N. Ithnin, “Clustering-and regression-based multi-criteria collaborative filtering with incremental updates,” *Inf. Sci.*, vol. 293, pp. 235–250, 2015. doi: [10.1016/j.ins.2014.09.012](https://doi.org/10.1016/j.ins.2014.09.012).
- [23] M. Nilashi, O. bin Ibrahim and N. Ithnin, “Hybrid recommendation approaches for multi-criteria collaborative filtering,” *Expert. Syst. Appl.*, vol. 41, no. 8, pp. 3879–3900, 2014. doi: [10.1016/j.eswa.2013.12.023](https://doi.org/10.1016/j.eswa.2013.12.023).
- [24] R. S. Sreepada, B. K. Patra, and A. Hernando, “Multi-criteria recommendations through preference learning,” in *Proc. 4th ACM IKDD Conf. Data. Sci.*, 2017, pp. 1–11.
- [25] M. Braunhofer, V. Codina, and F. Ricci, “Switching hybrid for cold-starting context-aware recommender systems,” in *Proc. 8th ACM Conf. Recomm. Syst.*, 2014, pp. 349–352.
- [26] B. B. Sinha and R. Dhanalakshmi, “DNN-MF: Deep neural network matrix factorization approach for filtering information in multi-criteria recommender systems,” *Neural. Comput. Appl.*, vol. 34, no. 13, pp. 10807–10821, 2022. doi: [10.1007/s00521-022-07012-y](https://doi.org/10.1007/s00521-022-07012-y).
- [27] D. Tallapally, R. S. Sreepada, B. K. Patra, and K. S. Babu, “User preference learning in multi-criteria recommendations using stacked auto encoders,” in *Proc. 12th ACM Conf. Recomm Syst.*, 2018, pp. 475–479.
- [28] D. Q. Zhang and S. C. Chen, “Clustering incomplete data using kernel-based fuzzy c-means algorithm,” *Neural. Process. Lett.*, vol. 18, pp. 155–162, 2003. doi: [10.1023/B:NEPL.0000011135.19145.1b](https://doi.org/10.1023/B:NEPL.0000011135.19145.1b).
- [29] C. V. M. Krishna, G. A. Rao, and S. Anuradha, “Analysing the impact of contextual segments on the overall rating in multi-criteria recommender systems,” *J. Big Data*, vol. 10, no. 1, pp. 16, 2023. doi: [10.1186/s40537-023-00690-y](https://doi.org/10.1186/s40537-023-00690-y).
- [30] J. L. Herlocker, J. A. Konstan, L. G. Terveen, and J. T. Riedl, “Evaluating collaborative filtering recommender systems,” *ACM Trans. Inf. Syst. (TOIS)*, vol. 22, no. 1, pp. 5–53, 2004. doi: [10.1145/963770.963772](https://doi.org/10.1145/963770.963772).
- [31] F. Cacheda, V. Carneiro, D. Fernández, and V. Formoso, “Comparison of collaborative filtering algorithms: Limitations of current techniques and proposals for scalable, high-performance recommender systems,” *ACM Trans. Web. (TWEB)*, vol. 5, no. 1, pp. 1–33, 2011. doi: [10.1145/1921591.1921593](https://doi.org/10.1145/1921591.1921593).
- [32] Z. Shu, Y. Bai, D. Zhang, J. Yu, Z. Yu, and X. Wu, “Specific class center guided deep hashing for cross-modal retrieval,” *Inf. Sci.*, vol. 609, pp. 304–318, 2022. doi: [10.1016/j.ins.2022.07.095](https://doi.org/10.1016/j.ins.2022.07.095).

- [33] Z. Shu, K. Yong, D. Zhang, J. Yu, Z. Yu, and X. J. Wu, "Robust supervised matrix factorization hashing with application to cross-modal retrieval," *Neural. Comput. Appl.*, vol. 35, no. 9, pp. 6665–6684, 2023. doi: [10.1007/s00521-022-08006-6](https://doi.org/10.1007/s00521-022-08006-6).
- [34] P. N. Ramesh and S. Kannimuthu, "Context-aware practice problem recommendation using learners' skill level navigation patterns," *Intell. Autom. Soft. Comput.*, vol. 35, no. 3, pp. 3845–3860, 2023. doi: [10.32604/iasc.2023.031329](https://doi.org/10.32604/iasc.2023.031329).
- [35] L. Li, Z. Shu, Z. Yu, and X. J. Wu, "Robust online hashing with label semantic enhancement for cross-modal retrieval," *Pattern. Recognit.*, vol. 145, pp. 109972, 2024. doi: [10.1016/j.patcog.2023.109972](https://doi.org/10.1016/j.patcog.2023.109972).
- [36] Z. Shu, K. Yong, J. Yu, S. Gao, C. Mao and Z. Yu, "Discrete asymmetric zero-shot hashing with application to cross-modal retrieval," *Neurocomputing*, vol. 511, pp. 366–379, 2022. doi: [10.1016/j.neucom.2022.09.037](https://doi.org/10.1016/j.neucom.2022.09.037).
- [37] V. S. Devi and S. Kannimuthu, "Author profiling in code-mixed WhatsApp messages using stacked convolution networks and contextualized embedding-based text augmentation," *Neural. Process. Lett.*, vol. 55, no. 1, pp. 589–614, 2023. doi: [10.1007/s11063-022-10898-3](https://doi.org/10.1007/s11063-022-10898-3).
- [38] K. C. Raja and S. Kannimuthu, "Conditional generative adversarial network approach for autism prediction," *Comput. Syst. Sci. Eng.*, vol. 44, no. 1, 2023. doi: [10.32604/csse.2023.025331](https://doi.org/10.32604/csse.2023.025331).
- [39] S. Wu, F. Sun, W. Zhang, X. Xie, and B. Cui, "Graph neural networks in recommender systems: A survey," *ACM Comput. Surv.*, vol. 55, no. 5, pp. 1–37, 2022. doi: [10.1145/3535101](https://doi.org/10.1145/3535101).
- [40] C. Gao *et al.*, "A survey of graph neural networks for recommender systems: Challenges, methods, and directions," *ACM Trans. Recomm. Syst.*, vol. 1, no. 1, pp. 1–51, 2023. doi: [10.1145/3568022](https://doi.org/10.1145/3568022).
- [41] Y. Gao *et al.*, "Chat-REC: Towards interactive and explainable llms-augmented recommender system," arXiv preprint arXiv:2303.14524, 2023.
- [42] Y. Deng, W. Zhang, W. Xu, W. Lei, T. S. Chua, and W. Lam, "A unified multi-task learning framework for multi-goal conversational recommender systems," *ACM Trans. Inf. Syst.*, vol. 4, no. 3, pp. 1–25, 2023. doi: [10.1145/3570640](https://doi.org/10.1145/3570640).
- [43] H. Jiawei, K. Micheline, and P. Jian, "Dm concepts and techniques preface and introduction," in S. Kusumadewi (Ed.), *Klasifikasi Status Gizi Menggunakan*, 2012, vol. 3, pp. 6–11.