

Received March 8, 2022, accepted April 2, 2022, date of publication April 6, 2022, date of current version April 12, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3165310

# A Multi-Criteria Collaborative Filtering Approach Using Deep Learning and Dempster-Shafer Theory for Hotel Recommendations

QUANG-HUNG LE<sup>1</sup>, TOAN NGUYEN MAU<sup>2</sup>, ROENGCHAI TANSUCHAT<sup>3</sup>,  
AND VAN-NAM HUYNH<sup>2</sup>, (Member, IEEE)

<sup>1</sup>Department of Information Technology, Quy Nhon University, Quy Nhon 55000, Vietnam

<sup>2</sup>School of Advanced Science and Technology, Japan Advanced Institute of Science and Technology, Nomi, Ishikawa 923-1211, Japan

<sup>3</sup>Center of Excellence in Econometrics, Faculty of Economics, Chiang Mai University, Chiang Mai 50200, Thailand

Corresponding authors: Quang-Hung Le (lequanghung@qnu.edu.vn) and Van-Nam Huynh (huynh@jaist.ac.jp)

This work was supported in part by the Ministry of Education and Training of Vietnam under Project B2020-DQN-08; and in part by the Centre of Excellence in Econometrics, Faculty of Economics, Chiang Mai University, Thailand.

**ABSTRACT** This paper addresses the problem of multi-criteria recommendation in the hotel industry. The main focus is to analyze user preferences from different aspects based on multi-criteria ratings and develop a new multi-criteria collaborative filtering method for hotel recommendations. Particularly, the proposed recommendation system integrates matrix factorization into a deep learning model to predict the multi-criteria ratings, and then the evidential reasoning approach is adopted to model the uncertainty of those ratings represented as mass functions in Dempster-Shafer theory of evidence. Finally, Dempster's rule of combination is utilized to aggregate those multi-criteria ratings to obtain the overall rating for recommendation. Extensive experiments conducted on a real-world dataset demonstrate the effectiveness and efficiency of the proposed method compared with other multi-criteria collaborative filtering methods.

**INDEX TERMS** Collaborative filtering, multi-criteria collaborative filtering, matrix factorization, deep learning, Dempster-Shafer theory.

## I. INTRODUCTION

Nowadays personalized recommendation plays a vital and indispensable role in most online service platforms such as e-commerce and new media websites to help users by providing suggestions regarding information that might be interest to them based on their preferences or historical data. For example, recommendation service has been incorporated into online market platforms such as e-commerce website [1], music/video/movies online store [2]–[4], and tourism [5]. In the tourism domain, travelers often spend time to search for hotels to stay from online travel websites based on their own requirements; such as a business traveler will prioritize the price and location of the hotel, while a tourist can put convenience and cleanliness first. Moreover, a traveler can only select one hotel at a time and it is quite inconvenience to change the choice if wrongly chosen. Therefore, a good

The associate editor coordinating the review of this manuscript and approving it for publication was Yin Zhang<sup>1</sup>.

hotel RS is especially helpful for saving time for travelers and reducing advertising costs for hotel owners.

Generally, RS aims to suggest items for users that they might have an interest in. Basically, RS methods can be categorized into three groups: Collaborative filtering approaches recommend items based on the user-item historical interactions such as ratings or feedback; Content-based approaches extract the mutual information between user and item to state the recommendation, these approaches are suitable with video, audio, and text data; while hybrid RS approaches can integrate multiple ways to state the recommendation [6]. Most RSs rely on the rating history of users to items, which supports learning the user preferences, item characteristics, and some additional correlation information between users and/or items [7], [8]. Then, the unknown rating of a user to an item can be predicted using such information. Such an approach is usually considered as user-based and item-based collaborative filtering methods, which take the average rating from  $k$ -nearest neighbor of user/item to predict the

unknown rating [9]. On the other hand, matrix factorization or singular value decomposition (SVD) approach can learn the preferences of each user over items by optimizing the embedding matrices for all users and items [10], [11]. However, the finding of optimal values for embedding matrices is time-consuming compared to items-based and user-based collaborative filtering methods. Recently, deep learning has emerged as a powerful technique for handling collaborative filtering problems because it can handle nonlinear transformation, representation learning, sequence modeling, and flexibility; those are end-to-end differentiable and provide suitable inductive biases for RS [6].

As for the problem of RS in the tourism domain, the collaborative filtering method is suitable for hotel recommendation because of its characteristics. However, a single-criteria rating cannot capture the information of all aspects of user experiences. In the Tripadvisor platform, perhaps the most well-known website for hotel browsing and booking, it uses a multi-criteria rating system to represent the rating of users to hotels [12]. These criteria include the overall rating and seven sub-criteria ratings: *Check in/front desk*, *Cleanliness*, *Service*, *Location*, *Room*, *Value*, and *Business Service*. Thus, a multi-criteria RS should be developed to deal with such multi-criteria ratings. Several attempts have been made on developing multi-criteria RS, and most of them are based on the conventional methods for single-criterion CF problem [13]. The overall rating is computed by aggregation of sub-criteria ratings in which the basic techniques such as average [14] or weighted-sum [15] can be used for aggregation. However, these basis aggregation methods do not take into account the uncertainty of sub-criteria ratings. As mentioned, users may have a critical bias in some criteria, and in that case, the overall rating may eliminate these facts.

In this study, we aim at developing a new collaborative filtering method that combines a deep neural network (DNN) model with evidential reasoning based on Dempster-Shafer theory of evidence for multi-criteria RS. In particular, we first design a DNN model that integrates a matrix factorization technique with a multilayer perceptron for predicting criteria ratings, each of which is considered as a piece of evidence for making prediction of the overall rating. As predicted criteria ratings are inherently associated with uncertainty, we propose a new method based on Dempster-Shafer theory of evidence for modeling criteria rating predictions and develop a discounting-and-combination scheme for multi-criteria rating aggregation to obtain the overall rating. A real-world multi-criteria rating dataset extracted from the Tripadvisor platform is used for experiments to demonstrate the effectiveness and applicability of the proposed method.

The rest of this paper is organized as follows. Section II briefly recalls some related work and Section III provides preliminaries on matrix factorization, multilayer perceptron with ReLU units and Dempster-Shafer theory of evidence that form the basis for the development of the proposed recommendation method detailed in Section IV. Section V describes the experimental results and analysis.

Finally, Section VI wraps up the paper with conclusions and future work.

## II. RELATED WORK

The collaborative filtering problem has a long history alongside the development of the Internet. The neighborhood-based methods use the similarity between users and/or items to predict the degree of preference [16], [17]. Besides that, the model-based approaches also give significantly outcome by learning the correlation between users and items; the common model-based approaches are Latent Semantic Analysis [18], Support Vector Machines [19], Bayesian Clustering [20], and Singular Value Decomposition (SVD) [11], [21]. Among these models, SVD is one of the most widely used techniques and it also has a positive effect on the deep learning based approaches.

In terms of related work on multi-criteria RS, Jannach *et al.* [22] use basic techniques such as the item-based or user-based collaborative filtering methods to predict the sub-criteria scores, and the overall rating is then obtained by an aggregation function. Moreover, several deep learning based approaches for handling multi-criteria RS problem have been also investigated in [23], [24]. Recently, Hong and Jung [25] proposed a multi-criteria tensor model that considers not only user preferences and multiple rating but also incorporating cultural factor into recommendation processes. Note that in this model the overall rating is equally treated as individual criteria ratings and tensor factorization is applied for prediction of unobserved users' preferences without using an aggregation operator.

An aggregation technique is required for combining the multi-criteria ratings, such aggregation approach could be simple as average [14] or weighted-sum [15]. For more complex aggregation methods, Nilashi *et al.* [5] introduced to use an adaptive network-based fuzzy inference system (ANFIS) to learn decision rules for prediction of the overall rating. Moreover, Nassar *et al.* [23] introduced the first DNN model for solving this problem. More recently, Shambour [26] employed deep autoencoders to exploit hidden relations between users with regard to multi-criteria preferences and the arithmetic mean is used for aggregation to compute the overall rating prediction; while Sinha and Dhanalakshmi [27] adopted the Social Spider Optimization (SSO) for seeking the optimal weights of the DNN models of multi-criteria recommender systems.

Different from the aforementioned work, in this paper we first design a DNN model for predicting individual multi-criteria ratings, which are considered as pieces of evidence supporting for the overall rating prediction. We then propose to model these criteria rating predictions by means of the so-called mass functions in Dempster-Shafer theory of evidence. This approach allows us not only to represent the uncertainty inherently associated with criteria rating predictions but also to develop a flexible framework for multi-criteria rating aggregation taking the criteria weights into account.

### III. PRELIMINARIES

In this section we first introduce the problem of multi-criteria collaborative filtering and then recall basic concepts used in the development of the proposed method.

#### A. MULTI-CRITERIA COLLABORATIVE FILTERING PROBLEM

Let  $U, I, C$  be the sets of users, items, and criteria with cardinalities of  $N_U, N_I$ , and  $N_C$ , respectively. Let  $R$  be a dataset consisting of the known multi-criteria ratings  $r_{uic}$  which represent the rating of user  $u$  to item  $i$  at the criterion  $c$ , for  $u \in U, i \in I$ , and  $c \in C$ . The rating values  $r_{uic}$  are normalized, for example, Tripadvisor use a Likert scale of five values  $\{1,2,3,4,5\}$  for presenting  $\{Terrible, Poor, Average, Very\_Good, Excellent\}$ , respectively. In Figure 1, the dataset  $R$  can be illustrated by a 3rd-order tensor, which is usually a sparse tensor because each user often has ratings on just several items in practice. The collaborative filtering problem aims at predicting criteria ratings  $\hat{r}_{uic}$  that are unknown for certain users, items, and criteria as well as their overall ratings  $\hat{r}_{ui0}$ .

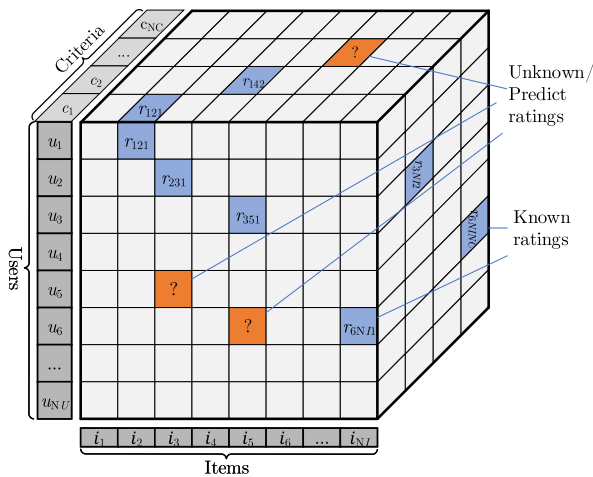


FIGURE 1. The multi-criteria dataset.

#### B. MATRIX FACTORIZATION FOR COLLABORATIVE FILTERING PROBLEM

Matrix factorization is the most used technique for collaborative filtering problem, which simultaneously decomposes the inference rating information of users and items into same length matrices. Whereas, the singular value decomposition (SVD) is the most naive approach for factorizing the single-criteria collaborative filtering [10], [11]. In particular, users and items are mapped into  $f$ -dimensional latent factor spaces, such that the prediction ratings are the inner-product on that space, and  $f$  is the embedding size. Note that, for the single-criteria collaborative filtering, each user can give a single rating for an item only, let  $r_{ui}$  and  $\hat{r}_{ui}$  be the known rating and unknown rating, respectively. Let  $q_i \in \mathbb{R}^f$  be the embedding vector of item  $i$  and  $p_u$  be the embedding vector for user  $u$ , then the prediction rating of user  $u$  to item  $i$  is the

inner-product  $\hat{r}_{ui} = p_u^T q_i$ . These values of  $p \in \mathbb{R}^{N_U \times f}$  and  $q \in \mathbb{R}^{N_I \times f}$  can be learned by a machine learning technique to minimize the loss with the known ratings in the dataset  $R$ . Formally, the objective function can be defined as:

$$\text{Minimize } \sum_{u,i \in R} (r_{ui} - p_u^T q_i)^2 + \lambda(\|q_i\|^2 + \|p_u\|^2) \quad (1)$$

where  $\lambda$  is the parameter controlling the sparsity of embedding matrices. The above objective function then can be optimized by Stochastic Gradient Descent (SGD), Adam optimizer, or Alternating Least Square (ALS) [10].

SVD can be also extended further upon addition the bias for each user and item, which can capture the variations of individuals. Let  $b_u$  and  $b_i$  denote respectively the bias embedding variables for user  $u$  and item  $i$ , the new objective function with basis information is defined as:

$$\text{Minimize } \sum_{u,i \in R} (r_{ui} - \hat{r}_{ui})^2 + \lambda(\|q_i\|^2 + \|p_u\|^2 + b_u^2 + b_i^2) \quad (2)$$

where  $\hat{r}_{ui} = \mu + b_u + b_i + p_u^T q_i$  with  $\mu$  being the average rating of all known ratings.

In this research, the SVD approach is adopted as a layer in our deep learning model.

#### C. MULTILAYER PERCEPTRON (MLP)

MLP is the basic feed-forward DNN with multiple hidden layers of threshold activation functions between the input layer and output layer. MLPs can be referred to as the universal approximators which can be represented by stacked layers of nonlinear transformations of activation functions [6]. Because the activation functions are not necessarily strictly binary classifiers, MLPs are suitable for predicting multi-criteria Likert-scale scoring [6].

In this paper, a customized MLP to estimate the preferences of users and items is proposed to handle multi-criteria RS in the travel domain.

#### D. RECTIFIED LINEAR UNIT (ReLU)

In context of DNN, ReLU is a simple neuron unit that uses an activation function to give the positive part of the input [28], [29].

$$\text{ReLU}(x) = x^+ = \max(0, x) \quad (3)$$

where  $x$  is the input value. This is also called as ramp function.

ReLU is widely used in DNN for feature extraction, computer vision, and speed recognition with high effectiveness compared to basic activation functions such as logistic sigmoid and hyperbolic tangent [29].

Our DNN model defines multiple hidden layers of ReLU to increase the accuracy of the SVD layer.

#### E. DEMPSTER-SHAFER THEORY (DST)

DST, also called evidence theory or theory of belief functions, provides a general framework for modeling and reasoning with uncertain and incomplete information as well

as a powerful tool for combining evidence from multiple sources [30], [31]. In the context of DST, a frame of discernment  $X$  is defined as a finite set of mutually exclusive and exhaustive hypotheses, e.g. the set of all possible answer to a given question, and a piece of evidence regarding the true answer to the question is represented by a so-called mass function  $m : 2^X \rightarrow [0, 1]$  satisfying:

$$\sum_{A \in 2^X} m(A) = 1 \quad \text{and} \quad m(\emptyset) = 0 \quad (4)$$

where  $2^X$  is the power set of  $X$ . For  $A \in 2^X$ , the quantity  $m(A)$  is interpreted as a measure of the belief exactly allocated to the hypothesis “the true answer is in  $A$ ”.

Two evidential operations that play an important role in the evidential reasoning are discounting operator and Dempster’s rule of combination [30]. The discounting operation is used when a source of evidence represented by a mass function  $m$  is known to have probability  $\alpha$  of reliability. Then we can discount the mass function  $m$  at discount rate of  $(1 - \alpha)$ , resulting in a new mass function  $m^\alpha$  defined by

$$m^\alpha(A) = \alpha \times m(A), \quad \text{for any } A \subset X \quad (5)$$

$$m^\alpha(X) = (1 - \alpha) + \alpha \times m(X) \quad (6)$$

When two distinct sources of information providing two corresponding pieces of evidence on the same frame  $X$  represented by two mass functions  $m_1$  and  $m_2$ , we can then use Dempster’s rule of combination to generate the combined mass function, denoted by  $m_\oplus = (m_1 \oplus m_2)$  (also called the orthogonal sum of  $m_1$  and  $m_2$ ), which is defined, for any  $A \in 2^X \setminus \emptyset$ , as follows

$$m_\oplus(A) = m_1 \oplus m_2(A) = \frac{\sum_{B \cap C = A} m_1(B)m_2(C)}{1 - \sum_{B \cap C = \emptyset} m_1(B)m_2(C)} \quad (7)$$

where

$$\sum_{B \cap C = \emptyset} m_1(B)m_2(C) \triangleq m_\oplus(\emptyset) \quad (8)$$

is the combined mass assigned to the empty set before normalization. Clearly, Dempster’s rule of combination is only applicable when  $m_\oplus(\emptyset) < 1$ .

As for decision making, a mass function  $m$  encoded the available evidence must be transformed into a so-called pignistic distribution function [32], denoted by  $Bp_m : X \rightarrow [0, 1]$ , which is defined as follows

$$Bp_m(x|x \in X) = \sum_{A \in 2^X, x \in A} \frac{m(A)}{|A|} \quad (9)$$

In this study, we apply DST for modelling uncertainty associated with individual criteria ratings predicted by a DNN model and aggregation of these criteria ratings for the overall rating. Recall that the set of rating grades used in hotel rating consists of five levels  $\{1, 2, 3, 4, 5\}$  representing  $\{Terrible, Poor, Average, Very\_Good, Excellent\}$ , respectively. The use

of DST for multi-criteria collaborative filtering problem proposed in this paper is basically similar to the evidential reasoning approach for multiple attribute decision-making under uncertainty as studied in [33]–[35].

#### IV. THE PROPOSED RECOMMENDATION METHOD

In this section we describe the proposed recommendation method consisting of two main phases of training and prediction: (1) In the first phase, a DNN model for predicting individual criteria ratings will be trained and the optimal parameters for DST modelling of uncertainty associated with predicted criteria ratings will be also determined using the collected data; (2) In the second phase, the trained DNN model will be used first to predict unknown criteria ratings, which will then serve as sources of evidence represented by DST modelling for predicting the overall rating. The outline of the proposed method is graphically illustrated in Figure 2.

##### A. MULTI-CRITERIA DNN MODEL

The proposed DNN model first starts with an embedding layer in which each user and item are mapped to a  $f$ -dimensional vector using the SVD method. Therefore, the model uses a 2-order tensor  $p_u \in \mathbb{R}^{N_u \times f}$  for embedding users and a 2-order tensor  $q_i \in \mathbb{R}^{N_i \times f}$  for embedding items. However, instead of using inner-product of  $p_u$  and  $q_i$  to predict the rating value for user  $u$  over item  $i$ ,  $p_u$  and  $q_i$  are concatenated into a single vector before forwarding to the next layer, as similar to the approach taken by Nassar et al. [23]. The flowchart of the proposed DNN model is depicted in Figure 3.

Then the DNN model continues with a series of layers of ReLU neurons. Because the output of the SVD layer has  $2 \times f$  neurons, the first rectified linear layer receives  $2 \times f$  signals. In this study, the number of hidden layers is designed to be around 3 and 6 to optimize the performance of the Tripadvisor dataset.

Note that the number of neurons in the output layer of the proposed DNN model must fit with the number of criteria, which is 4 for our testing dataset. For optimization of the proposed DNN model, basis optimizers such as Stochastic Gradient Descent (SGD) or Adam can be applied [36].

##### B. EVIDENCE MODELING

In the proposed recommendation method, individual criteria ratings predicted from the trained DNN model are considered as pieces of evidence for making the prediction of the overall rating. These pieces of evidence are represented as mass functions defined on the set of rating grades

$$\mathcal{H} = \{Terrible, Poor, Average, Very\_Good, Excellent\}$$

where masses are determined based on the probability density function (PDF) estimated using the training data.

Particularly, let denote  $\hat{r}_{uic}$  the predicted rating of user  $u$  to item  $i$  on criterion  $c$ . Then, the mass function representing

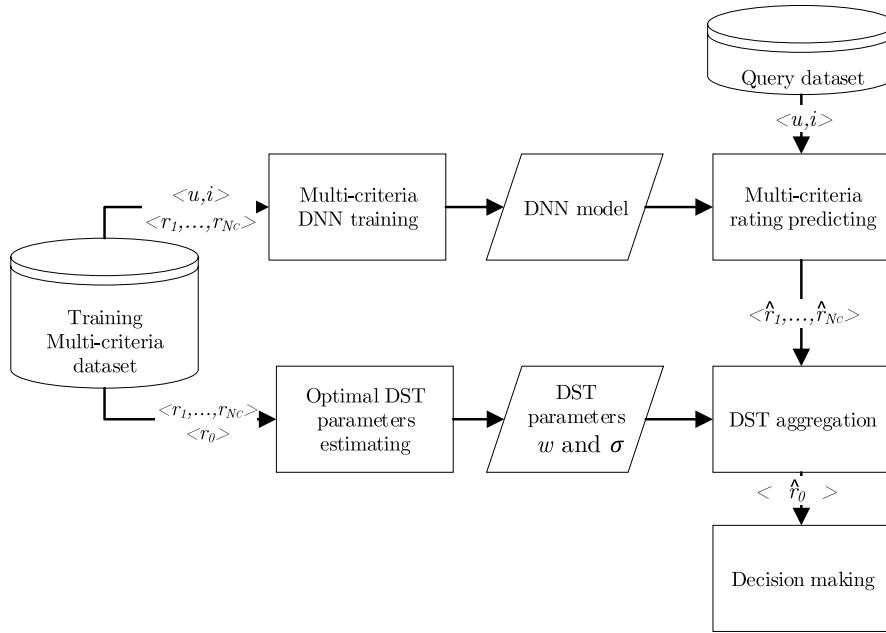


FIGURE 2. Overview of proposed method.

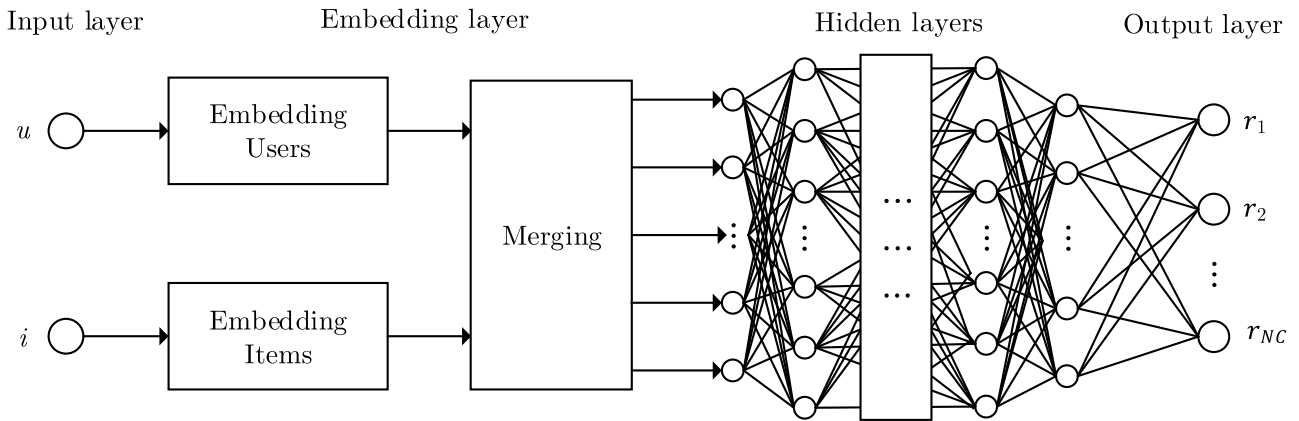


FIGURE 3. The proposed DNN model for multi-criteria collaborative filtering recommendation system.

evidence received from the predicted rating  $\hat{r}_{uic}$ , denoted by  $m_{\hat{r}_{uic}} : 2^{\mathcal{H}} \rightarrow [0, 1]$ , is defined as:

$$\begin{aligned}
 m_{\hat{r}_{uic}}(\{Terrible\}) &= \text{PDF}(\hat{r}_{uic}, \sigma, 1) \\
 m_{\hat{r}_{uic}}(\{Poor\}) &= \text{PDF}(\hat{r}_{uic}, \sigma, 2) \\
 m_{\hat{r}_{uic}}(\{Average\}) &= \text{PDF}(\hat{r}_{uic}, \sigma, 3) \\
 m_{\hat{r}_{uic}}(\{Very\_Good\}) &= \text{PDF}(\hat{r}_{uic}, \sigma, 4) \\
 m_{\hat{r}_{uic}}(\{Excellent\}) &= \text{PDF}(\hat{r}_{uic}, \sigma, 5) \\
 m_{\hat{r}_{uic}}(A) &= 0, \quad \text{for } A \in 2^{\mathcal{H}} \setminus \{\{l\}_{l \in \mathcal{H}}\} \quad (10)
 \end{aligned}$$

where  $\text{PDF}(\hat{r}_{uic}, \sigma, x)$  is the Gaussian's probability density function:

$$\text{PDF}(\hat{r}_{uic}, \sigma, x) = \frac{1}{\sigma \sqrt{2\pi}} e^{-\frac{1}{2} \left( \frac{x - \hat{r}_{uic}}{\sigma} \right)^2} \quad (11)$$

and the masses are normalized such that:

$$\sum_{l \in \mathcal{H}} m_{\hat{r}_{uic}}(\{l\}) = 1 \quad (12)$$

Figure 4 graphically illustrates two specific cases of evidence modeling.

It is worth noting here that the softening parameter  $\sigma$  used in Gaussian's probability density function PDF is also estimated using the training data. In other words, it can be learnt from the known criteria ratings  $r_{uic}$  and their overall ratings in the training dataset  $R$ . More particularly, the optimal value of  $\sigma$  is determined by minimizing root mean squared error (RMSE) score as follows:

$$\sigma = \arg \min_{\sigma^*} \text{RMSE}(R, \hat{R}_{\sigma^*}) \quad (13)$$



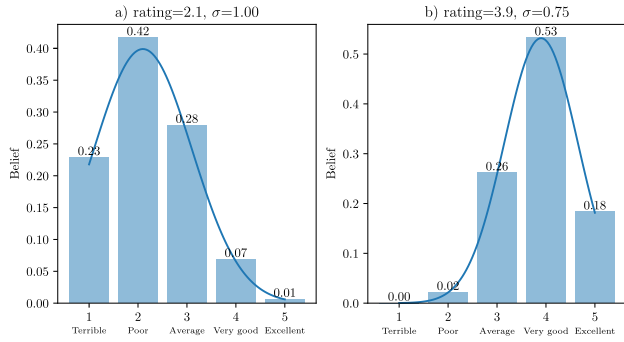


FIGURE 4. Examples for evidence modeling using normal distribution.

where  $\hat{R}_{\sigma^*}$  is the set of all predicted overall ratings, each of which is obtained by using DST aggregation of individual criteria ratings generated by applying softening parameter  $\sigma^*$ .

**C. EVIDENCE COMBINATION FOR OVERALL RATING PREDICTION**

In the previous step, each predicted criterion rating  $\hat{r}_{uic}$  is represented by a mass function  $m_{\hat{r}_{uic}}$  defined by (10)–(12). Then we have  $N_C$  mass functions representing  $N_C$  pieces of evidence from criteria predictions  $\hat{r}_{uic}$ , for  $c = 1, \dots, N_C$ . Consider further that each individual criterion  $c$  is associated with a weight  $w_c$  representing its relative importance in contribution to the overall rating, we now combine these  $N_C$  mass functions  $m_{\hat{r}_{uic}}$  taking their relative importance into account to generate a combined mass function for predicting the overall rating. This can be done within the framework of DST making use of the discounting and Dempster’s rule of combination. Formally, the overall mass function, denoted by  $m_{\hat{r}_{ui}}$ , generated by combining  $m_{\hat{r}_{uic}}$ ’s taking their weights  $w_c$ ’s into account is defined by:

$$m_{\hat{r}_{ui}} = \bigoplus_{c=1}^{N_C} m_{\hat{r}_{uic}}^{w_c} \quad (14)$$

where  $m_{\hat{r}_{uic}}^{w_c}$  is the discounted mass function as defined in (5)–(6), while  $\oplus$  is Dempster’s combination operator as defined by (7) above. Specifically, for two criteria  $c_1$  and  $c_2$  with their weights  $w_1$  and  $w_2$ , respectively, by (5)–(6) we first obtain two discounted mass functions  $m_{\hat{r}_{uic_1}}^{w_1}$  and  $m_{\hat{r}_{uic_2}}^{w_2}$  as follows:

$$\begin{aligned} m_{\hat{r}_{uic_1}}^{w_1}(\{l\}) &= w_1 \times m_{\hat{r}_{uic_1}}(\{l\}), \quad \text{for } l \in \mathcal{H} \\ m_{\hat{r}_{uic_1}}^{w_1}(\mathcal{H}) &= 1 - w_1 \\ m_{\hat{r}_{uic_1}}^{w_1}(A) &= 0, \quad \text{for } A \in 2^{\mathcal{H}} \setminus \{\{l\}_{l \in \mathcal{H}}, \mathcal{H}\} \end{aligned}$$

and

$$\begin{aligned} m_{\hat{r}_{uic_2}}^{w_2}(\{l\}) &= w_2 \times m_{\hat{r}_{uic_2}}(\{l\}), \quad \text{for } l \in \mathcal{H} \\ m_{\hat{r}_{uic_2}}^{w_2}(\mathcal{H}) &= 1 - w_2 \\ m_{\hat{r}_{uic_2}}^{w_2}(A) &= 0, \quad \text{for } A \in 2^{\mathcal{H}} \setminus \{\{l\}_{l \in \mathcal{H}}, \mathcal{H}\} \end{aligned}$$

Then by (7) we easily get the combined mass function resulting from the combination of these two discounted mass functions as

$$\begin{aligned} m_{\hat{r}_{uic_1}}^{w_1} \oplus m_{\hat{r}_{uic_2}}^{w_2}(\{l\}) &= K[m_{\hat{r}_{uic_1}}^{w_1}(\{l\})m_{\hat{r}_{uic_2}}^{w_2}(\{l\}) \\ &\quad + m_{\hat{r}_{uic_1}}^{w_1}(\mathcal{H})m_{\hat{r}_{uic_2}}^{w_2}(\mathcal{H}) \\ &\quad + m_{\hat{r}_{uic_1}}^{w_1}(\mathcal{H})m_{\hat{r}_{uic_2}}^{w_2}(\{l\})] \end{aligned} \quad (15)$$

$$m_{\hat{r}_{uic_1}}^{w_1} \oplus m_{\hat{r}_{uic_2}}^{w_2}(\mathcal{H}) = K[m_{\hat{r}_{uic_1}}^{w_1}(\mathcal{H})m_{\hat{r}_{uic_2}}^{w_2}(\mathcal{H})] \quad (16)$$

$$m_{\hat{r}_{uic_1}}^{w_1} \oplus m_{\hat{r}_{uic_2}}^{w_2}(A) = 0, \quad \text{for } A \in 2^{\mathcal{H}} \setminus \{\{l\}_{l \in \mathcal{H}}, \mathcal{H}\} \quad (17)$$

where  $K$  is the normalizing factor defined by

$$K = \left[ 1 - \sum_{l \in \mathcal{H}} \sum_{l' \neq l \in \mathcal{H}} m_{\hat{r}_{uic_1}}^{w_1}(\{l\})m_{\hat{r}_{uic_2}}^{w_2}(\{l'\}) \right]^{-1}$$

The mass assigned to the whole set  $\mathcal{H}$  of the combined mass function

$$m_{\hat{r}_{uic_1}}^{w_1} \oplus m_{\hat{r}_{uic_2}}^{w_2}(\mathcal{H}) = K(1 - w_1)(1 - w_2)$$

is called the ignorance score, which essentially reflects the ignorance regarding prediction of the overall rating as a result of multi-criteria rating combination. Intuitively, the higher the weight of a criterion rating, the less ignorance it attributed to the combination. This observation motivated us to determine criteria weights based on the data. In particular, the weight  $w_c$  of a criterion  $c$  is defined as the normalized inverse-value of the difference between the criterion’s average rating and the overall average rating as follows:

$$w_c = \frac{|\bar{r}_c - \bar{r}|^{-1}}{\sum_{c'=1}^{N_C} |\bar{r}_{c'} - \bar{r}|^{-1}} \quad (18)$$

where  $\bar{r}_c$  is the average rating score of the criterion  $c$  and  $\bar{r}$  is the average overall rating. Practically, the interpretation of this weighting method is that the larger the difference of a criterion’s rating and the overall rating, the less important of the criterion’s rating in contribution to the overall rating. It is of interest to note that the weighting technique studied in [15] can also be used to determine the criteria weights.

Finally, the aggregated mass function  $m_{\hat{r}_{ui}}$ , i.e. (14), is considered as uncertain assessment of the overall rating of user  $u$  over item  $i$ , denoted by  $\hat{r}_{ui}$ . In order to make prediction for the overall rating  $\hat{r}_{ui}$ , this mass function  $m_{\hat{r}_{ui}}$  is eventually transformed into its corresponding pignistic distribution function, denoted by  $Bp_{ui}$  via (9) and then the overall rating  $\hat{r}_{ui}$  is obtained as the following expectation:

$$\begin{aligned} \hat{r}_{ui} &= Bp_{ui}(\text{Terrible}) \times 1 + Bp_{ui}(\text{Poor}) \times 2 \\ &\quad + Bp_{ui}(\text{Average}) \times 3 + Bp_{ui}(\text{Very\_Good}) \times 4 \\ &\quad + Bp_{ui}(\text{Excellent}) \times 5 \end{aligned} \quad (19)$$

where particularly

$$Bp_{ui}(l) = m_{\hat{r}_{ui}}(\{l\}) + \frac{m_{\hat{r}_{ui}}(\mathcal{H})}{5}, \quad \text{for any } l \in \mathcal{H}$$

As stated, the proposed recommendation method follows a multi-criteria-rating-prediction-and-combination-based approach that interestingly allows for incorporating the uncertainty associated with multi-criteria rating predictions into the combination by means of DST to generate the overall rating prediction. This makes our model different from the previously developed models mostly based on the aggregation-function-based approach introduced in [14].

### D. DNN\_DST ALGORITHM

The main steps of the proposed method previously described are summarized into Algorithm 1 below.

---

#### Algorithm 1 DNN\_DST Algorithm

---

**Require:** Multi-criteria dataset  $R$ , Embedding size  $f$ ,  $\mathcal{H}$ ,  $N_U, N_I, N_C$ , weights  $W = \{w_1, \dots, w_{N_C}\}$ .

**Ensure:** Multi-criteria DNN model, softening parameter  $\sigma$

- 1: Configure the DNN model following the architecture depicted in Figure 3, which takes user  $u$  and item  $i$  as the inputs and returns  $N_C$  outputs.
  - 2: Train the DNN model using known multi-criteria ratings  $r_{uic}$  in the dataset  $R$ .
  - 3: Learn the optimal softening parameter  $\sigma$  for evidence modeling using (13).
  - 4: Represent  $N_C$  individual criteria ratings as  $N_C$  pieces of evidence as detailed in Section IV.B.
  - 5: Combine  $N_C$  pieces of evidence using (14) and make prediction of the overall rating using (19).
  - 6: **return** Trained multi-criteria DNN model,  $\sigma$ .
- 

In the following section, we present experiments conducted on a real-world dataset to demonstrate the applicability and efficiency of the proposed multi-criteria recommendation method.

## V. EXPERIMENTS AND RESULTS

### A. DATA SET

A real-world dataset of 533,430 multi-criteria reviews from 291,793 users for 8,297 hotels in Vietnam extracted from Tripadvisor.com is used for experiments. After eliminating isolated users and hotels, the testing dataset comprises of 274,572 multi-criteria ratings from 84,579 users to 6,854 hotels with a sparsity of 99.9526%. The collected criteria are value rating (27.507%), location rating (27.478%), cleanliness rating (27.341%), service rating (7.511%), and the overall rating (100%). A small portion of the testing dataset is shown in Table 1. In order not to affect the processing speed of the tensors, the missing ratings are filled with the average rating on each row in the dataset.

### B. EVALUATION METRICS

To evaluate the predicting performance of the proposed method on the Tripadvisor dataset, we divide the dataset into training and testing datasets using 5-fold cross validation technique. Let denote  $R$  and  $R^*$  the training and

testing datasets, respectively. The recommendation systems give the outcome set  $\hat{R}$  of predicting ratings which has the corresponding pair of users and items as in the testing dataset  $R^*$ . To measure the effectiveness, the basis loss functions and coefficient of determination adopted are as follows:

- Mean absolute error (MAE): MAE measures the average magnitude of the errors in a set of forecasts:

$$\text{MAE}(\hat{R}, R^*) = \frac{1}{|R^*|} \sum_{u,i,c \in R^*} |\hat{r}_{uic} - r_{uic}^*| \quad (20)$$

- Root mean squared error (RMSE):

$$\text{RMSE}(\hat{R}, R^*) = \sqrt{\sum_{u,i,c \in R^*} \frac{(\hat{r}_{uic} - r_{uic}^*)^2}{|R^*|}} \quad (21)$$

- Coefficient of Determination (CoD): This metric can show the proportion if the variance in the dependent variable that is predictable from the independent variable(s).

$$\text{CoD}(\hat{R}, R^*) = 1 - \frac{\sum_{u,i,c \in R^*} (\hat{r}_{uic} - r_{uic}^*)^2}{\sum_{u,i,c \in R^*} (r_{uic}^* - \bar{r})^2} \quad (22)$$

where  $\bar{r}$  is the mean rating of  $R^*$ .

In terms of complexity analysis, the processing time including training and testing time of the proposed and compared methods are also measured.

Note that, for MAE, RMSE, and computation time, lower values mean better performance.

### C. COMPETITORS

To demonstrate the efficiency and advantages of the proposed method, we conducted experiments using not only the conventional approaches such as CF\_user [16] and CF\_item [17] but also advanced models such as SVD [11] that is based on matrix factorization techniques, SVD++ [37] that extends the SVD model by considering implicit information to further improve the prediction accuracy, and RNN4Rec [38] that is a recurrent neural networks (RNN) based approach for session-based recommendations. As those models were developed for the single-criterion CF problem, for applying them to the multi-criteria CF problem, we deploy the models for criteria rating prediction and use the arithmetic mean to predict the overall rating as done in [26]. This strategy basically follows the aggregation-function-based approach presented in [14]. The multi-criteria versions of SVD, SVD++ and RNN4Rec are denoted by MSVD, MSVD++ and MRNN4Rec, respectively. The Python implementations of the aforementioned models can be found in LibRecommender<sup>1</sup> package in the PyPI repository.

In terms of multi-criteria CF approaches, the state-of-the-art methods such as HOSVD\_ANFIS [5], [39] and DNN\_DNN [23] are implemented for comparative study.

<sup>1</sup><https://pypi.org/project/LibRecommender/>

TABLE 1. A small portion of the testing dataset.

user	item	overall	value	location	cleanliness	service
Brian	1592633	5	5	5		5
Brian	300325	5	4	3	5	5
Janet B	2151597	5	5	3	5	5
Janet B	587110	2				
-Global-Galivanter-	12384745	5	5	4	5	5
-Global-Galivanter-	10892617	5				
-Global-Galivanter-	12422625	3				
-Global-Galivanter-	8757855	4			5	5
-Global-Galivanter-	12950280	5	5	4		5
-IrishRover2010-	8595053	3			5	5
-IrishRover2010-	11952439	5	5	4	5	5

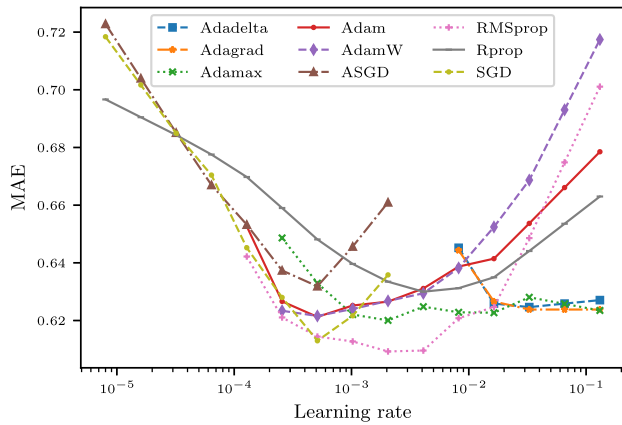


FIGURE 5. DNN performances for different optimizers.

HOSVD\_ANFIS provides a multi-criteria recommendation model that not only improves the recommendation quality and predictive accuracy but also is able to handle the scalability and sparsity problems in multi-criteria collaborative filtering. Basically, this model integrates the HOSVD [37] for dimensionality reduction and cosine-based clustering with ANFIS for inducing fuzzy rules to predict the overall ratings. DNN\_DNN method consists of building two deep neural networks, one for predicting criteria ratings and the other for learning an aggregation function to be used for the overall rating prediction. In addition, we have also implemented other combinations of the multi-criteria rating prediction and aggregation approaches mentioned above. The first combination is denoted by DNN\_ANFIS, which integrates a DNN model for predicting multi-criteria scores with ANFIS model for the overall score prediction; the second combination is denoted by HOSVD\_DNN which uses HOSVD to train the prediction model for multi-criteria ratings and a DNN for the overall score aggregation. These DNN models are trained using the LibRecommender package and TensorFlow<sup>2</sup> library.

All experiments were conducted using a high-end computer with Intel Xeon G-6240M 2.6GHz (18 Cores x4) CPU.

<sup>2</sup><https://github.com/tensorflow/tensorflow>

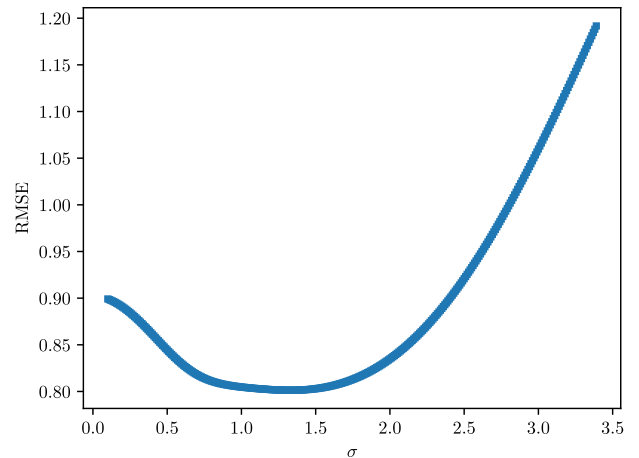


FIGURE 6. The impact of softening parameter  $\sigma$ .

For the proposed method, we use PyTorch<sup>3</sup> library to train the DNN model, evidential reasoning<sup>4</sup> package to analyze the uncertainty, and pyds<sup>5</sup> package to aggregate the criteria ratings.

D. PERFORMANCE ANALYSIS

Figure 5 demonstrates the effectiveness of different optimizers which can be used to optimize the used DNN model with different learning rates. As the result, different optimizers have different optimal learning rate values for the testing dataset. In particular, Root Mean Square Propagation (RMSprop) and Stochastic Gradient Descent (SGD) can achieve the lowest MAE scores, because SGD tends to take less computation time than RMSprop, we recommend using SGD to optimize the proposed method. For the rest of the experiments in this paper, the SGD optimizer is used by default.

We also conduct an experiment to see how the selection of softening parameter  $\sigma$  affects the performance of the proposed method. Figure 6 depicts the relationship between soft-

<sup>3</sup><https://github.com/pytorch/pytorch>

<sup>4</sup><https://pypi.org/project/evidentialreasoning/>

<sup>5</sup><https://github.com/reineking/pyds>



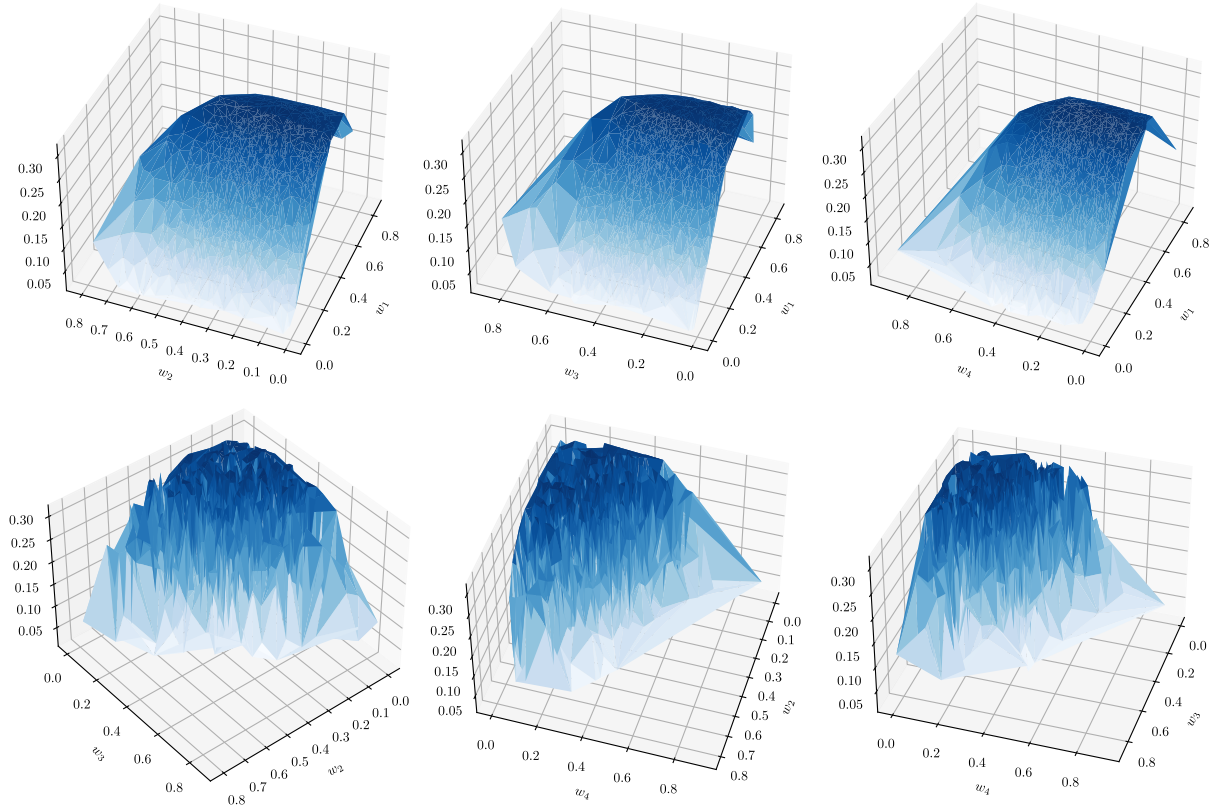


FIGURE 7. Ignorance scores over the criteria weights; The vertical axes show the ignorance score.

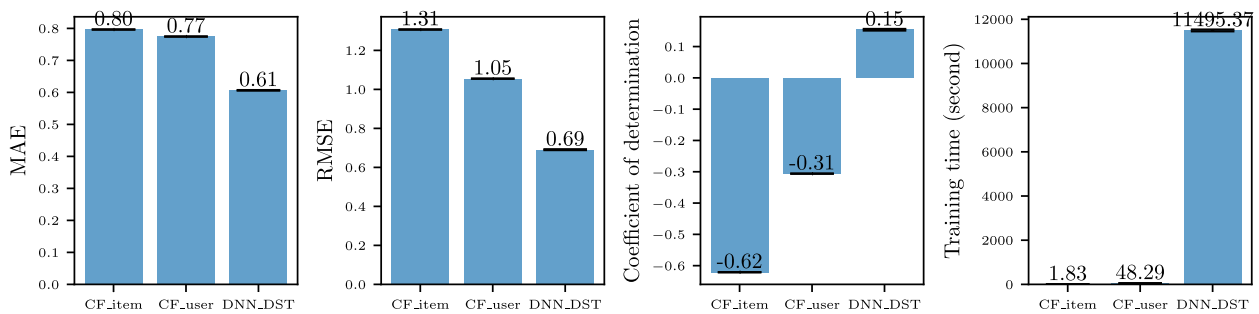


FIGURE 8. DNN\_DST versus CF; 128 runs.

ening parameter  $\sigma$  and the performance in terms of RMSE. From this experiment, the optimal  $\sigma$  values for the Tripadvisor dataset are between 1.3 and 1.4, and we set  $\sigma = 1.35$  for the following experiments.

### E. IGNORANCE ANALYSIS

Additionally we conduct an experiment to analyze the ignorance regarding prediction of the overall ratings resulted from multi-criteria rating aggregation. Figure 7 shows the ignorance scores over the criteria weights involved in criteria rating combination for the testing dataset. Specifically, the

ignorance scores resulted from evidence combination of all pairs of criteria are shown.

It is shown that the first criterion (*Value* criterion with its weight  $w_1$ ) highly contributes to the ignorance score in criteria rating combination. This can be intuitively explained that because most of travelers prepay the hotel fee done long before the date they start staying at the hotel. Thus, they might be satisfied with the price long before the time they are providing ratings that are usually given during or right after travelers leave the hotel. Consequently, it makes *Location*, *Cleanliness*, and *Service* become more important and sensitivity. The numerical results also give the same support,

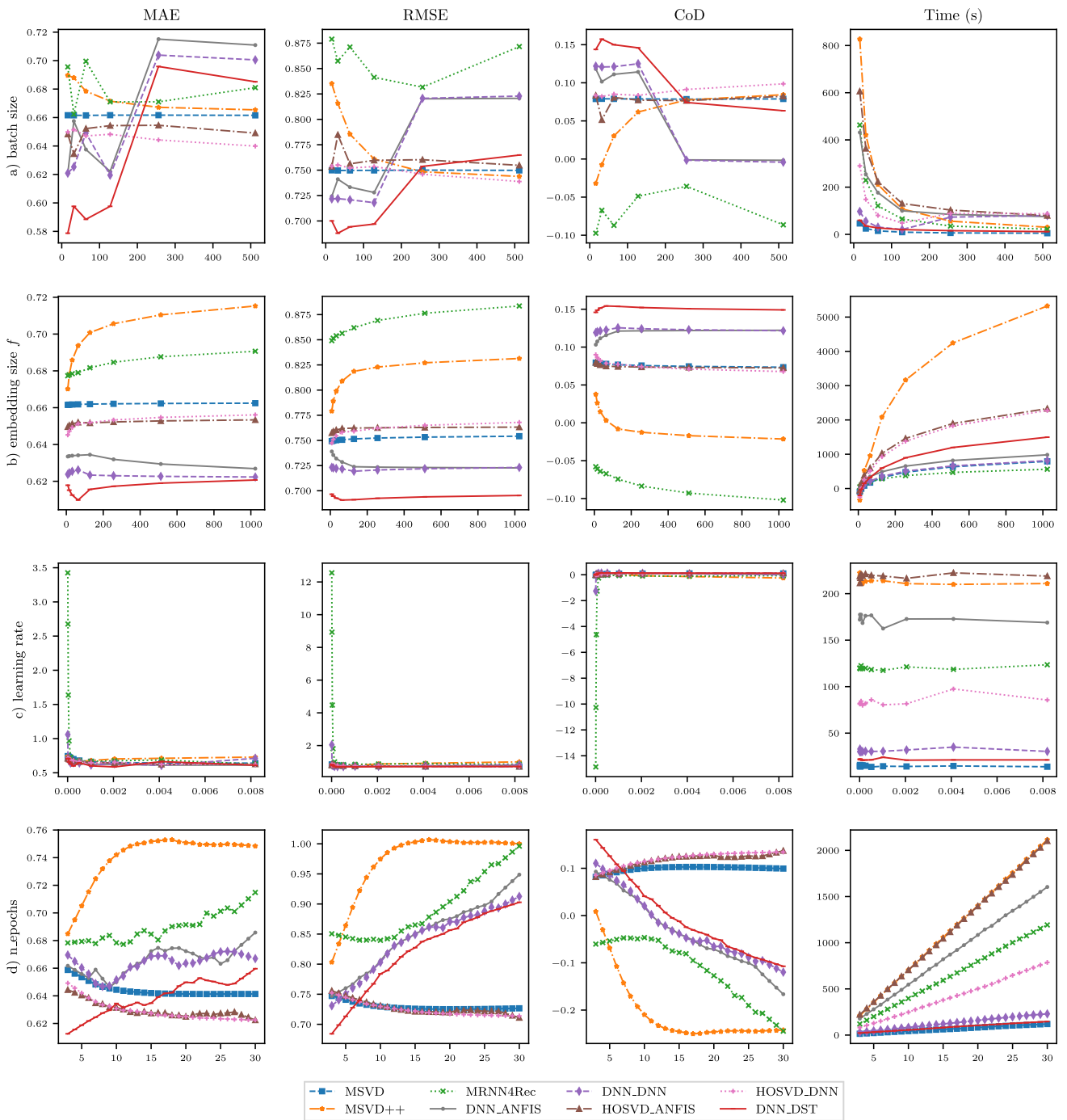


FIGURE 9. Performance comparison.

the *Value* has the largest difference from the *Overall rating* value, which is 0.596 on average, while the other criteria *Location*, *Cleanliness*, and *Service* take 0.512, 0.492, and 0.509, respectively. The second fact that can be seen from the figure is that the *Location* ( $w_2$ ), *Cleanliness* ( $w_3$ ), and *Service* ( $w_4$ ) are similar in their contribution to the ignorance score in criteria rating combination.

### F. PERFORMANCE COMPARISON

The proposed method (DNN\_DST) was first compared with conventional CF approaches, namely User-based (CF\_user) and Item-based (CF\_item) methods. Figure 8 shows that DNN\_DST outperforms these two conventional methods in terms of recommendation accuracy. However, as DNN\_DST needs time for training the DNN model, it takes significantly

**TABLE 2.** Average metric scores of the compared methods over different tuning parameters.

batch size	Method	MSVD	MRNN4Rec	DNN_DNN	HOSVD_DNN	MSVD++	DNN_ANFIS	HOSVD_ANFIS	DNN_DST
	MAE	0.662±0.21	0.761±0.29	0.653±0.25	0.645±0.20	0.678±0.21	0.661±0.25	0.648±0.21	<b>0.624±0.24</b>
	RMSE	0.750±0.24	1.122±0.64	0.754±0.29	0.746±0.24	0.775±0.25	0.761±0.29	0.761±0.24	<b>0.716±0.27</b>
	CoD	0.099±0.02	-0.113±0.07	-0.027±0.07	<b>0.121±0.03</b>	-0.209±0.08	-0.038±0.08	0.118±0.03	-0.007±0.08
	Time(s)	<b>68.32±34.4</b>	657.39±344.6	132.32±65.6	420.35±224.7	1161.73±607.8	892.64±460.5	1154.73±600.7	86.72±41.1
<i>f</i>	Method	MSVD	MRNN4Rec	DNN_DNN	HOSVD_DNN	MSVD++	DNN_ANFIS	HOSVD_ANFIS	DNN_DST
	MAE	0.662±0.22	0.682±0.23	0.624±0.21	0.651±0.22	0.696±0.23	0.633±0.21	0.652±0.22	<b>0.615±0.21</b>
	RMSE	0.751±0.25	0.862±0.29	0.721±0.24	0.758±0.25	0.811±0.27	0.728±0.24	0.761±0.25	<b>0.693±0.23</b>
	CoD	0.077±0.03	-0.074±0.03	0.123±0.04	0.078±0.03	0.001±0.03	0.117±0.04	0.075±0.03	<b>0.152±0.05</b>
	Time(s)	<b>280.70±350.9</b>	263.16±221.5	311.18±359.18	817.96±1017.48	1844.73±2597.0	447.37±378.6	920.69±1018.4	523.41±671.2
<i>lr</i>	Method	MSVD	MRNN4Rec	DNN_DNN	HOSVD_DNN	MSVD++	DNN_ANFIS	HOSVD_ANFIS	DNN_DST
	MAE	0.695±0.18	1.221±0.86	0.696±0.23	0.674±0.17	0.705±0.45	0.665±0.20	0.674±0.17	<b>0.647±0.19</b>
	RMSE	0.782±0.21	3.083±3.46	0.878±0.45	0.778±0.20	0.827±1.83	0.776±0.23	0.779±0.20	<b>0.745±0.22</b>
	CoD	0.040±0.05	-2.880±4.33	-0.050±0.39	0.054±0.05	-0.019±2.26	0.056±0.05	0.052±0.05	<b>0.088±0.06</b>
	Time(s)	<b>14.59±3.8</b>	120.00±30.0	31.09±9.1	83.86±21.9	213.44±53.3	172.66±50.0	218.60±56.3	21.60±6.2
<i>n_epochs</i>	Method	MSVD	MRNN4Rec	DNN_DNN	HOSVD_DNN	MSVD++	DNN_ANFIS	HOSVD_ANFIS	DNN_DST
	MAE	0.644	0.689	0.663	<b>0.629</b>	0.741	0.665	0.630	0.639
	RMSE	0.729	0.892	0.840	<b>0.724</b>	0.974	0.849	0.727	0.822
	CoD	0.099	-0.113	-0.027	0.121	-0.209	-0.038	<b>0.118</b>	-0.007
	Time(s)	<b>68.32±34.4</b>	657.39±344.6	132.32±65.6	420.35±224.2	1161.73±607.8	892.64±460.5	1154.73±600.7	86.72±41.1

<sup>1</sup> *f* – Embedding size, *lr* – Learning rate

more time compared to the work of finding the nearest neighbors.

Then, as previously described, a comprehensive experiment was conducted to evaluate the performance of the proposed DNN\_DST method by comparing it with the recent state-of-the-art methods for the multi-criteria CF problem, namely MSVD, MSVD++, MRNN4Rec, DNN\_DNN, HOSVD\_DNN, HOVSD\_ANFIS, DNN\_ANFIS. Figure 9 shows the experimental results with evaluation metrics of MAE, RMSE, CoD and computation time according to four hyperparameters used for tuning. Table 2 summarizes the average scores of the compared methods over different tuning parameters. As the results indicated, MSVD is the fastest one but also the most inaccurate approach because of its simplicity. Besides that, DNN\_DNN method which is somewhat sophisticated as using two separated DNN models shows relatively good results in both accuracy and computation time. Furthermore, DNN\_ANFIS and HOSVD\_ANFIS tend to be the slowest methods as significant additional time is required in those methods for training the ANFIS network, while their results are not better than that of the proposed approach as well as other DNN-based methods.

Especially, the results show that in most cases the proposed DNN\_DST method achieves the best performance with the lowest MAE, RMSE and the highest CoD among compared methods, and it is also the second fastest method.

## VI. CONCLUSION

In this study, we proposed a new approach named DNN\_DST that integrates a DNN model with Dempster-Shafer theory of evidence for multi-criteria CF problem. Essentially, the proposed approach first designed a DNN model that incorporates a SVD technique as its first layer for predicting multi-criteria

ratings, and then adopted the evidential reasoning approach for modeling those ratings as pieces of evidence by means of mass functions to be aggregated for prediction of the overall rating. Interestingly, by modeling criteria ratings predicted from the DNN model as pieces of evidence, we are able to analyze the uncertainty inherently associated with these predictions to be incorporated into multi-criteria rating aggregation. We also proposed a data-driven method for determining criteria weights which represent relative importance of criteria ratings in prediction of the overall rating. Experiments conducted on a real-world dataset showed that the proposed recommendation method outperforms other state-of-the-art methods in terms of MAE, RMSE, and CoD scores in most testing cases, while it also shows a comparable efficiency compared to the existing CF techniques.

As for the future work, we plan to explore the following two aspects to further improve the proposed DNN\_DST approach. First, we will investigate the problem of how to design a unified architecture for the proposed DNN\_DST approach that allows to optimize multiple parameters simultaneously so as to significantly reduce the training time while maintaining the recommendation performance of the proposed model. Second, we will further investigate methods for estimating softening parameter  $\sigma$  and for determining criteria weights and evaluate the impact of these parameters on modeling uncertainty associated with criteria rating predictions and ignorance resulted in multi-criteria rating aggregation. Moreover, it is also interesting to investigate how the proposed approach can be extended for sequential recommendation models [40], [41] and graphical recommendation models [42]. Finally, we will explore the applicability of the proposed approach in real-world systems.

## REFERENCES

- [1] 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, Jan./Feb. 2003.
- [2] J. Pérez-Marcos and V. L. Batista, "Recommender system based on collaborative filtering for spotify's users," in *Proc. Int. Conf. Practical Appl. Agents Multi-Agent Syst.* Cham, Switzerland: Springer, 2017, pp. 214–220.
- [3] P. Covington, J. Adams, and E. Sargin, "Deep neural networks for YouTube recommendations," in *Proc. 10th ACM Conf. Recommender Syst.*, Sep. 2016, pp. 191–198.
- [4] W. Jian, J. He, K. Chen, Y. Zhou, and Z. Tang, "Collaborative filtering and deep learning based recommendation system for cold start items," *Expert Syst. Appl.*, vol. 69, pp. 29–39, Mar. 2017.
- [5] M. Nilashi, A. Ahani, M. D. Esfahani, E. Yadegaridehkordi, S. Samad, O. Ibrahim, N. M. Sharef, and E. Akbari, "Preference learning for eco-friendly hotels recommendation: A multi-criteria collaborative filtering approach," *J. Cleaner Prod.*, vol. 215, pp. 767–783, Apr. 2019.
- [6] S. Zhang, L. Yao, A. Sun, and Y. Tay, "Deep learning based recommender system: A survey and new perspectives," *ACM Comput. Surv.*, vol. 52, no. 1, pp. 1–38, Feb. 2019.
- [7] R. Wang, H. K. Cheng, Y. Jiang, and J. Lou, "A novel matrix factorization model for recommendation with LOD-based semantic similarity measure," *Expert Syst. Appl.*, vol. 123, pp. 70–81, Jun. 2019.
- [8] M. He, S. Zhang, and Q. Meng, "Learning to style-aware Bayesian personalized ranking for visual recommendation," *IEEE Access*, vol. 7, pp. 14198–14205, 2019.
- [9] J. Wang, A. P. de Vries, and M. J. T. Reinders, "Unifying user-based and item-based collaborative filtering approaches by similarity fusion," in *Proc. 29th Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr. (SIGIR)*, 2006, pp. 501–508.
- [10] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *IEEE Comput.*, vol. 42, no. 8, pp. 30–37, Aug. 2009.
- [11] Y. Koren, "Factorization meets the neighborhood: A multifaceted collaborative filtering model," in *Proc. 14th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining (KDD)*, 2008, pp. 426–434.
- [12] *Tripadvisor Website*. Accessed: Apr. 13, 2021. [Online]. Available: <https://www.tripadvisor.com/>
- [13] G. Adomavicius, N. Manouselis, and Y. Kwon, "Multi-criteria recommender systems," in *Recommender Systems Handbook*. Boston, MA, USA: Springer, 2011, pp. 769–803.
- [14] G. Adomavicius and Y. Kwon, "New recommendation techniques for multicriteria rating systems," *IEEE Intell. Syst.*, vol. 22, no. 3, pp. 48–55, May/Jun. 2007.
- [15] T. Y. Tang and G. McCalla, "The pedagogical value of papers: A collaborative-filtering based paper recommender," *J. Digit. Inf.*, vol. 10, no. 2, pp. 1–12, 2009.
- [16] B. Sarwar, G. Karypis, J. Konstan, and J. Reidl, "Item-based collaborative filtering recommendation algorithms," in *Proc. 10th Int. Conf. World Wide Web (WWW)*, 2001, pp. 285–295.
- [17] Z.-D. Zhao and M.-S. Shang, "User-based collaborative-filtering recommendation algorithms on Hadoop," in *Proc. 3rd Int. Conf. Knowl. Discovery Data Mining*, Jan. 2010, pp. 478–481.
- [18] T. Hofmann, "Latent semantic models for collaborative filtering," *ACM Trans. Inf. Syst.*, vol. 22, no. 1, pp. 89–115, Jan. 2004.
- [19] Z. Xia, Y. Dong, and G. Xing, "Support vector machines for collaborative filtering," in *Proc. 44th Annu. Southeast Regional Conf. (ACM-SE)*, 2006, pp. 169–174.
- [20] J. S. Breese, D. Heckerman, and C. Kadie, "Empirical analysis of predictive algorithms for collaborative filtering," 2013, *arXiv:1301.7363*.
- [21] A. Paterek, "Improving regularized singular value decomposition for collaborative filtering," in *Proc. KDD Cup Workshop*, 2007, pp. 5–8.
- [22] D. Jannach, Z. Karakaya, and F. Gedikli, "Accuracy improvements for multi-criteria recommender systems," in *Proc. 13th ACM Conf. Electron. Commerce (EC)*, 2012, pp. 674–689.
- [23] N. Nassar, A. Jafar, and Y. Rahhal, "A novel deep multi-criteria collaborative filtering model for recommendation system," *Knowl.-Based Syst.*, vol. 187, Jan. 2020, Art. no. 104811.
- [24] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.
- [25] M. Hong and J. J. Jung, "Multi-criteria tensor model for tourism recommender systems," *Expert Syst. Appl.*, vol. 170, May 2021, Art. no. 114537.
- [26] Q. Shambour, "A deep learning based algorithm for multi-criteria recommender systems," *Knowl.-Based Syst.*, vol. 211, Jan. 2021, Art. no. 106545.
- [27] 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.*, 2022, doi: [10.1007/s00521-022-07012-y](https://doi.org/10.1007/s00521-022-07012-y).
- [28] K. Fukushima and S. Miyake, "Neocognitron: A self-organizing neural network model for a mechanism of visual pattern recognition," in *Competition and Cooperation in Neural Nets*. Berlin, Germany: Springer, 1982, pp. 267–285.
- [29] X. Glorot, A. Bordes, and Y. Bengio, "Deep sparse rectifier neural networks," in *Proc. 14th Int. Conf. Artif. Intell. Statist.*, 2011, pp. 315–323.
- [30] G. Shafer, *A Mathematical Theory of Evidence*. Princeton, NJ, USA: Princeton Univ. Press, 1976.
- [31] K. Sentz and S. Ferson, *Combination of Evidence in DempstersShafer Theory*, vol. 4015. Albuquerque, NM, USA: Sandia National Laboratories, 2002.
- [32] P. Smets, "Decision making in the TBM: The necessity of the pignistic transformation," *Int. J. Approx. Reasoning*, vol. 38, no. 2, pp. 133–147, 2005.
- [33] J.-B. Yang and M. G. Singh, "An evidential reasoning approach for multiple-attribute decision making with uncertainty," *IEEE Trans. Syst., Man, Cybern.*, vol. 24, no. 1, pp. 1–18, Jan. 1994.
- [34] V.-N. Huynh, Y. Nakamori, T.-B. Ho, and T. Murai, "Multiple-attribute decision making under uncertainty: The evidential reasoning approach revisited," *IEEE Trans. Syst., Man, Cybern. A, Syst. Humans*, vol. 36, no. 4, pp. 804–822, Jul. 2006.
- [35] N. D. K. Ngo, T. Q. Le, R. Tansuchat, T. Nguyen-Mau, and V.-N. Huynh, "Evaluating innovation capability in banking under uncertainty," *IEEE Trans. Eng. Manag.*, early access, Jan. 20, 2022, doi: [10.1109/TEM.2021.3135556](https://doi.org/10.1109/TEM.2021.3135556).
- [36] D. Choi, C. J. Shallue, Z. Nado, J. Lee, C. J. Maddison, and G. E. Dahl, "On empirical comparisons of optimizers for deep learning," 2019, *arXiv:1910.05446*.
- [37] L. de Lathauwer, N. D. Moor, and J. Vandewalle, "A multilinear singular value decomposition," *SIAM J. Matrix Anal. Appl.*, vol. 21, no. 4, pp. 1253–1278, Jul. 2000.
- [38] B. Hidasi, A. Karatzoglou, L. Baltrunas, and D. Tikk, "Session-based recommendations with recurrent neural networks," 2015, *arXiv:1511.06939*.
- [39] M. Nilashi, O. B. Ibrahim, N. Ithnin, and R. Zakaria, "A multi-criteria recommendation system using dimensionality reduction and neuro-fuzzy techniques," *Soft Comput.*, vol. 19, no. 11, pp. 3173–3207, Nov. 2015.
- [40] B. Schifferer, C. Deotte, J.-F. Puget, G. D. S. P. Moreira, G. Titericz, J. Liu, and R. Ak, "Using deep learning to win the booking.com wsdm webtour21 challenge on sequential recommendations," in *Proc. WebTour@ WSDM*, 2021, pp. 22–28.
- [41] S. Wang, L. Hu, Y. Wang, L. Cao, Q. Z. Sheng, and M. Orgun, "Sequential recommender systems: Challenges, progress and prospects," in *Proc. 28th Int. Joint Conf. Artif. Intell. (IJCAI)*, Aug. 2019, pp. 6332–6338, doi: [10.24963/ijcai.2019/883](https://doi.org/10.24963/ijcai.2019/883).
- [42] S. Wang, L. Hu, Y. Wang, X. He, Q. Z. Sheng, M. A. Orgun, L. Cao, F. Ricci, and P. S. Yu, "Graph learning based recommender systems: A review," 2021, *arXiv:2105.06339*.



**QUANG-HUNG LE** received the Ph.D. degree in computer science from the University of Engineering and Technology, Vietnam National University, Hanoi, in 2016. He is currently a Lecturer with the Department of Information Technology, Quy Nhon University, Vietnam. His research interests include natural language processing, machine learning, artificial intelligence, and recommender systems. He has published a number of conferences and journal papers in these areas.





**TOAN NGUYEN MAU** received the bachelor’s degree (Hons.) from the Ho Chi Minh City University of Science (HCMUS), Vietnam National University Ho Chi Minh City (VNU-HCM), Vietnam, in 2013, and the master’s and Ph.D. degrees in information science from the Japan Advanced Institute of Science and Technology (JAIST), in 2016 and 2021, respectively. His research interests include data mining, machine learning, and parallel processing using GPGPU.



**ROENGCHAI TANSUCHAT** received the bachelor’s, master’s, and Ph.D. degrees in economics from Chiang Mai University, Thailand, in 1997, 2001, and 2006, respectively. He is currently an Associate Professor with the Faculty of Economics and the Director of the Center of Excellence in Econometrics, Chiang Mai University. His research interests include econometrics, data analytics, risk modeling, energy finance, commodity y finance, international commodity trade, agricultural economics and supply chain, quantum econometrics, and finance. He has received the Golden Elephant Award 2019 for Outstanding Researcher in Social and Human Science from Chiang Mai University and the National Research Council Award (Research Award) 2016 from the National Research Council of Thailand (NRTC).



**VAN-NAM HUYNH** (Member, IEEE) received the Ph.D. degree in mathematics from the Vietnam Academy of Science and Technology, in 1999. He is currently a Professor at the School of Knowledge Science, Japan Advanced Institute of Science and Technology (JAIST). His current research interests include machine learning, data mining, AI reasoning, argumentation, multi-agent systems, decision analysis and management science, and kansei information processing and applications.

He also serves as an Area Editor for *International Journal of Approximate Reasoning*, the Editor-in-Chief for *International Journal of Knowledge and Systems Science*, and the Editorial Board Member for the *Array* journal.

...