IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received 5 February 2024, accepted 18 April 2024, date of publication 25 April 2024, date of current version 15 May 2024.

Digital Object Identifier 10.1109/ACCESS.2024.3393417

RESEARCH ARTICLE

Review-Based Recommender System Using Outer Product on CNN

SEIN HONG[1], **XINZHE LI**[1], **SIGEON YANG**[1], **AND JAEKYEONG KIM**[1,2]

[1]Department of Big Data Analytics, Kyung Hee University, Seoul 02447, Republic of Korea
[2]School of Management, Kyung Hee University, Seoul 02447, Republic of Korea

Corresponding author: Jaekyeong Kim (jaek@khu.ac.kr)

This work was supported in part by the BK21 FOUR (Fostering Outstanding Universities for Research) funded by the Ministry of Education (MOE), South Korea; and in part by the National Research Foundation of Korea (NRF).

ABSTRACT The expansion of the e-commerce market has led to the challenge of information overload, necessitating the development of recommender systems. The recommender system aids users in decision-making by suggesting items that align with their preferences. However, conventional recommendation models rely solely on quantitative user behavior data, such as user ratings, and lead to limitations in recommendation performance due to the sparsity problem. To address these issues, recent research has leveraged convolutional neural networks (CNNs) to extract and incorporate semantic information from user reviews. However, several prior studies have a disadvantage in that they fail to account for the intricate interactions between users and items directly. In this study, we introduce a novel approach, the Review-based recommender system using Outer Product on CNN (ROP-CNN) model, which adeptly captures and incorporates semantic features from reviews to address the complex interactions between users and items using CNN. The experimental results, using real user-review datasets, demonstrate that the ROP-CNN model outperforms existing baseline models for prediction accuracy. And this study presents a novel theoretical and methodological perspective in recommendation research, suggesting a method that integrates user preference information from reviews into recommender systems by leveraging rich user-item interaction information.

INDEX TERMS Collaborative filtering, recommender system, convolutional neural network, online review, outer product.

I. INTRODUCTION

With advancements in information and communications technology (ICT), the e-commerce market has seen rapid expansion. Online transactions have been simplified, making it more convenient for users to access information on preferred items, thus eliminating the need to visit physical stores [1]. However, the surge in online information availability has led to information overload, creating difficulties for users in selecting suitable items [2], [3], [4]. Recommender systems address this challenge by aiding users during online information searches [5]. Many e-commerce companies utilize recommender systems to lower users' information search costs, offering personalized recommendations for items and services, which positively impacts sales [6], [7]. According to reports, 60% of the videos watched by YouTube users were suggested by its recommendation algorithms, and Netflix also disclosed that 80% of the content watched by users was selected through a recommender system [8]. Therefore, e-commerce platforms are actively introducing recommender systems to not only provide item information to users but also to offer personalized recommendation services reflecting personal preferences [9], [10].

One of the predominant techniques in recommender systems is collaborative filtering (CF). CF generates similar neighbors based on quantitative user behavior records, such as purchase history, clicks, ratings, and views, to provide users with services or content they may prefer [11], [12]. However, the problem of data sparsity persists in CF-based models, as users typically rate only a subset of

The associate editor coordinating the review of this manuscript and approving it for publication was Pasquale De Meo.

items [13], [14]. Moreover, relying solely on user behavior records might overlook the user's purchase motivation as well as the details about the user and item [15]. Recent studies have addressed this issue by incorporating user reviews as additional information. Reviews, being free-form text, enable the construction of a recommender system that captures specific user preferences as detailed evaluations, allowing users to freely express item characteristics, thereby mitigating the sparsity problem associated with CF [16], [17].

Various methods are being explored to incorporate reviews into recommender systems. Recently, in the field of natural language processing (NLP), significant efforts have been made to extract semantic expressions from reviews using convolutional neural networks (CNNs). CNNs are adept at processing text by applying convolutional layers to capture local features, such as the order of words and contextual information, thereby reducing complexity and enhancing the accuracy of predictions. This improvement in prediction accuracy positively influences the performance of recommender systems. For instance, Kim et al. [18] proposed a Convolutional Matrix Factorization (ConvMF) model that leverages review text as an additional source of information. Firstly, this model employs CNN and word embedding to capture the items' latent features within item reviews. After that, the inferred latent features are integrated into a probabilistic Matrix Factorization (PMF) model to compute the users' ratings for target items. As a result, this study confirmed that incorporating contextual information from reviews through CNN significantly contributes to performance improvement rather than solely using rating information. Also, Alfarhood and Cheng [19] proposed DeepHCF (A Deep Learning Based Hybrid Collaborative Filtering Approach for Recommendation Systems). This method employs CNN to extract semantic information in the review text and multilayer perceptron (MLP) to extract user-item interaction information. The latent features of users and items learned by each model are utilized by factorization machines (FM) for our model prediction. In general, recommender systems perform poorly on sparse data, but DeepHCF achieves superior performance by capturing intricate and high-dimensional user-item interactions.

While there has been active use of CNNs to extract semantic information from reviews and incorporate it into recommender systems, previous approaches have a disadvantage in that they do not effectively reflect high-level interactions, as the interactions between users and items are independently considered. To address this limitation, new methodologies that utilize CNNs to learn about interactions have been proposed. For example, He et al. [20] created a two-dimensional (2-D) interaction map through the outer product of embedding vectors for users and items, constructing a recommender system that effectively learns user-item interactions using CNNs, resulting in improved recommendation performance. The proposed methodology also reflects interactions with other embedding dimensions through outer products, enabling the construction of a stable and generalized model with fewer parameters.

In this study, we propose a Review-based recommender system using Outer product on CNN (ROP-CNN) model that leverages the strengths of CNNs to intricately incorporate both interactions and reviews. This study employs two CNN models to extract latent interactions between users and items, as well as semantic information contained in reviews. First, to extract multi-dimensional interactions, we generated a 2-D interaction map through the outer product and applied a 2-D CNN. Subsequently, a multi-channel CNN, employing multiple filters, is used to effectively learn rich linguistic expressions and extract semantic features from reviews. Finally, a MLP is used to learn and predict user preferences for specific items. The recommendation performance of the proposed model was experimentally validated using review data collected from Amazon.com, the world's largest e-commerce platform. The experiments demonstrated that the proposed model outperforms state-of-the-art (SOTA) recommender systems, indicating the effectiveness of using CNNs to learn latent user-item interactions and semantic features from reviews. Additional experiments show the superiority of outer products in learning user-item interactions. Finally, this study introduces changes in model performance through various parameters and discusses the implications.

The rest of the paper is organized as follows: In Section II, we present the theoretical background of the recommender systems. In Section III, ROP-CNN is described in depth. In Section IV, we present the data for experiments in this study, the evaluation metrics, the benchmark models, and the experimental results. Finally, Section V outlines the conclusions, implications, and limitations.

II. RELATED WORK

A. REVIEW-BASED RECOMMENDER SYSTEM

The recommender system is a methodology that recommends items or services that users might prefer based on their past purchase behaviors and item information. As a result, it can help reduce the efforts required to search for information and play a role in increasing user satisfaction and sales [21]. Among various recommendation algorithms, Collaborative Filtering (CF), the most popular recommendation algorithm, predicts the target user's preferences by finding neighbors with similar preferences based on the user's historical ratings [12].

However, the conventional algorithms solely rely on quantitative information, such as ratings, and face the challenge of not adequately reflecting users' specific preferences. Consequently, in recent years, researchers have been actively developing recommender systems that leverage online reviews. Users often tend to refer to online reviews before making purchases on e-commerce sites. This allows them to consider the opinions of other users about the items or services, making it a crucial role in their decision-making process [17]. Reviews play a pivotal role in reflecting latent user needs and, as a result, serve as a critical component for the development of effective user management and expansion strategies [22],

[23]. In the studies of McAuley and Leskovec [24] and Zhang and Wang [25], topic modeling techniques were applied to extract important topics from reviews. Afterward, they analyzed the latent associations between the extracted topics and user ratings, leading to the creation of a more accurate user preference prediction model. Bhojne et al. [26] conducted an analysis of restaurant reviews using the SentiWordNet sentiment dictionary. They classified reviews into positive and negative categories and used sentiment scores to create a personalized recommendation algorithm that reflects user attributes. Furthermore, Kim [27] used text-mining techniques to extract features from item reviews and calculated sentiment scores through sentiment analysis. These sentiment scores were utilized as weights when calculating PLS (Purchase Likelihood Score) and this approach outperformed traditional CF. Unlike structured rating information, reviews capture users' latent motivation and provide rich insights into user preferences, as users freely express not only their opinions or preferences but also the characteristics of items. This has been demonstrated in studies focusing on user management and expansion potential. Therefore, incorporating reviews into recommender systems yields more personalized recommendation results compared to traditional systems relying solely on user rating data. The previous existing review-based recommender systems typically employed sentiment analysis and text mining techniques to quantify and integrate online reviews into recommendation algorithms. However, values extracted using sentiment scores and text mining techniques have limitations in capturing the detailed semantic features of reviews, such as syntactic patterns or correlations.

To address these limitations and enhance performance, active research has been conducted on utilizing CNNs for reviews, thereby extracting latent preferences and incorporating them into recommendation models. One of the early models adopting this methodology is deep cooperative neural networks (DeepCoNN), proposed by Zheng et al. [28]. DeepCoNN utilizes two parallel TextCNNs, one for learning user latent features from user-written reviews and the other for learning item latent features from reviews on items. The two latent vectors thus learned are then combined through matrix factorization (MF) to predict ratings. DeepCoNN effectively mitigates the sparsity problem while enhancing recommendation performance. Building on this, Chen et al. [29] introduced the neural attentional regression model with review-level explanations (NARRE). This model not only predicts ratings but also automatically selects useful reviews by incorporating an attention mechanism into DeepCoNN. Alfarhood and Cheng [19] proposed a deep learning-based hybrid collaborative filtering (DeepHCF) model, which applies CNN to reviews and multilayer perceptron to user rating information. The model combines reviews and rating information using factorization machines (FM) to learn interactions, exhibiting excellent performance even on sparse datasets. Liu et al. [30] proposed the hybrid neural recommendation model (HRDR), which uses MLP to extract rating information and an attentional mechanism along with a CNN to extract information from reviews. This model creates a recommendation algorithm to learn and predict user preferences using the MF method. CNN, unlike bag-of-words (BoW), considers the relationship between words and extracts more sophisticated item latent factors, showcasing outstanding recommendation performance. Active research on using CNNs continues for extracting semantic features from reviews and enhancing recommendation performance. However, research on applying CNN-based approaches for capturing complex user-item interactions is still lacking.

B. USER-ITEM INTERACTION

Improving the performance of recommender systems hinges on key factors, such as extracting latent factors of users and items and modeling the interactions between them [20], [31]. In recent years, researchers have endeavored to effectively capture complex interactions between users and items to enhance recommendation performance. For instance, He et al. [32] introduced a neural collaborative filtering (NCF) method employing a deep neural network to address the limitation of conventional CF, which can only consider the linear relationship between users and items. Specifically, they utilized generalized matrix factorization (GMF) to extract linear interactions between the latent vectors of users and items, while employing MLP to capture non-linear interactions. The combination of these two pieces of information is used to predict the user's preference for the item. Additionally, He and Chua [33] proposed a neural factorization machine (NFM) technique to model high-dimensional and nonlinear interaction information and incorporated an MLP to enhance the performance of factorization.

As mentioned earlier, conventional models have the limitation of handling user and item latent vectors independently without directly considering user-item interactions [34]. Recently, CNN-based approaches have been explored to overcome this limitation by directly incorporating complex interactions between users and items. For example, convolutional neural collaborative filtering (ConvNCF) generates a 2-D interaction map based on the outer product of the embedding vectors of users and items. Since the structure of this interaction map is considered local features, which are central to image processing, CNN can be applied. This approach enables the construction of stable and generalized models with fewer parameters compared to MLP. The diagonal components of the interaction map encompass both element-wise and concatenation, while the non-diagonal components contain more information as they represent correlations with other embedding dimensions not considered in matrix factorization (MF) or NCF [35]. Furthermore, the convolutional neural networks-based deep collaborative filtering (CNN-DCF) model, proposed by Wu et al. [31], employs a recommendation algorithm that enhances both computational speed and recommendation accuracy by adding a fully connected layer to the interaction map generated by ConvNCF.

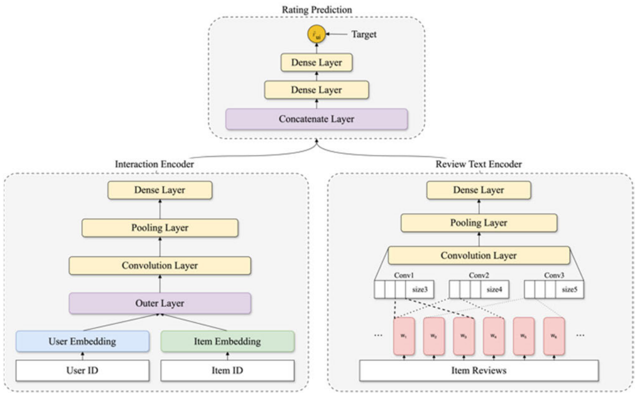


FIGURE 1. ROP-CNN architecture.

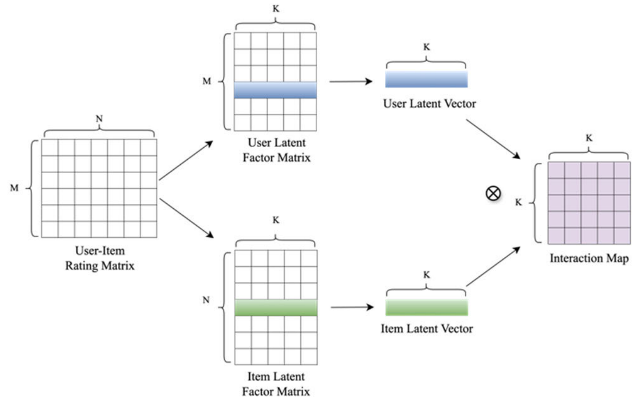


FIGURE 2. Interaction map.

These studies offer a new perspective on effectively modeling interactions between users and items, contributing significantly to the performance improvement of recommender systems. In the recommender system, the major advantage of CNNs is that they can capture more diverse and richer information by considering complex interactions between embedding vectors.

III. ROP-CNN MODEL

The ROP-CNN model is proposed to overcome the limitations of review-based recommender systems, which insufficiently learn complex interactions between users and items. This model is composed of three main components, as illustrated in Figure 1, and focuses on capturing high-dimensional and intricate user-item interactions using outer product and CNNs. First, the interaction encoder generates an interaction map based on outer products to capture latent interactions between users and items, whereby a 2-D CNN is applied to output interaction vectors. Second, the review encoder aims to extract linguistic expressions from reviews. Here, a one-dimensional CNN (1-D CNN), which is widely used in the field of natural language processing, is employed to effectively extract the high-level semantic elements inherent in reviews. Finally, the rating prediction uses MLP to predict the feature vectors extracted from the interaction encoder and review encoder, whereby they are concatenated to learn the non-linear relationship between user-item interactions and the semantic elements of reviews. The details of each part are presented below.

A. INTERACTION ENCODER

In the first part, the Interaction Encoder, a 2-D CNN is used to learn complex latent interactions between users and items. At this time, the user ID and item ID used as input to the model are converted into sparse vectors (p_u, q_i) through a one-hot encoding. This sparse vector is transformed into a latent factor vector by passing through an embedding layer as shown in (1).

$$p_u = P^T v_u^U, \quad q_i = Q^T v_i^I \tag{1}$$

Here $P \epsilon R^{M \times K}$, $Q \epsilon R^{N \times K}$ are the embedding matrix for user ID and item ID, respectively. K, M, and N denote the embedding size, number of user features, and number of item features, respectively.

The outer product, which includes both element-wise and concatenation, can capture not only interactions between elements in the embedding dimension but also interactions between different dimensions. Therefore, as shown in Figure 2, the user and item latent factor vectors are learned using outer products to generate an interaction map, which is expressed by (2).

$$E = p_u \otimes q_i = p_u q_i^T \tag{2}$$

Here $\otimes$ represents the outer product, and E is a K $\times$ K matrix. The Interaction map is 2D matrix format which is the same as an image. This works in a manner like the extraction of local features in images, thus, in this study, we can extract features using a 2-D CNN [20].

Following this, a hidden layer was added to the output interaction feature vector to reduce the dimension of the interaction feature vector, as shown in (3).

$$\begin{aligned} \phi_{E_1}(E) &= f\left(W_{E_1}^T E + b_{E_1}\right), \\ &\cdots \\ E' = \phi_{E_L}(E_{L-1}) &= f(W_{E_L}^T E_{L-1} + b_{E_L}) \end{aligned} \tag{3}$$

In (3), ϕ_{E_x} represents the mapping function for the x-th layer in the process of reducing the dimensionality of the interaction feature vector; W_{E_x}, b_{E_x}, and f represent the weight, bias, and ReLU activation function for the x-th layer, respectively. Thus, the interaction feature vector can be expressed as E'.

B. REVIEW ENCODER

In the second part, the review encoder uses 1-D CNN, which has shown excellent performance in natural language processing, to extract users' latent preferences contained in reviews [36]. Since this method can effectively extract the semantic features contained in a review, it can obtain the user's specific evaluation of a particular product [37]. Filters of different sizes are used to extract the n-gram semantic features contained in reviews. By doing so, the information loss that may occur during the training process can be minimized [38]. The detailed description is as follows:

In this study, first, we applied the embedding technique to map each word contained in the review into a dense vector. In this process, the semantic information of the review is captured, and words with similar meanings are grouped. Here, each word is represented as $x \in R^D$, where D is the dimensionality of the embedding vector. If the number of words contained in the review $s = \{x_1, x_2, \ldots, x_n\}$ is N, then the embedding matrix of the review is represented as X $\in R^{N \times D}$.

Next, semantic features are extracted from the review embedding matrix X by applying filters of different sizes to the convolutional layer. In this process, a convolutional operation is performed, as shown in (4), using the sliding window and applying filters of different sizes to capture the interactions of neighboring words in the review.

$$c_j = f(X * W_j + b_j) \tag{4}$$

In (4), * represents the convolutional operator and b_j represents the bias; f is the activation function and is used to add non-linearity to the deep learning model, to learn complex relationships in more detail. In this study, we applied the rectified linear unit (ReLU) activation function, which has the advantages of effectively solving the vanishing gradient problem and presents relatively low computational complexity, as expressed in (5).

$$ReLU\ (x) = max(0, x) \tag{5}$$

Next, the extracted feature map is passed through the max pooling layer to extract the maximum value in the feature map and the most prominent semantic feature, as shown in (6).

$$O_j = max([c_1, c_2, \ldots, c_{N-j+1}]) \tag{6}$$

The model uses m different filters to obtain m maximum feature values. Here, O is defined as a fixed size vector, as shown in (7)

$$O = [O_1, O_2, \ldots, O_m] \tag{7}$$

By adding a hidden layer to the review feature vector that is output, the dimensionality of the review feature vector is reduced, as shown in (8). The main purpose of this dimensionality reduction is to adjust the dimensionality in the process of learning the non-linear relationship between user-item interaction and semantic information of the review so that these two pieces of information can play an equal role.

$$\begin{gathered} \phi_{O_1}(O) = f(W_{O_1}^T O + b_{O_L}) \\ \ldots \\ O' = \phi_{O_L}(O_{L-1}) = f(W_{O_L}^T O_{L-1} + b_{O_L}) \end{gathered} \tag{8}$$

In (8), ϕ_{O_x} denotes the mapping function for the x-th layer in the process of reducing the dimensionality of the review feature vector; W_{O_x}, b_{O_x}, and f denote the weight, bias, and ReLU activation functions for the x-th layer, respectively. The new review feature vector obtained through this process can be expressed as O'.

C. RATING PREDICTION

Finally, the extracted user-item interaction vector and the user's latent preference vector are combined and passed through the MLP to predict the final rating r_{ui}. The interaction between the interaction map and review is defined by (9).

$$H = E' \oplus O' = \begin{bmatrix} E' \\ O' \end{bmatrix} \tag{9}$$

where $\oplus$ denotes the concatenation operator, and H denotes the vector obtained by concatenating the user-item interaction vector and semantic vector extracted from the review. Next, MLP is added to learn the nonlinear relationship between the two pieces of information and predict the rating, as shown in (10).

$$\hat{y} = \sigma(W_H \cdot H + b_H) \tag{10}$$

where W_H and b_H denote the weight and bias of the feature vector H, respectively, in the process of predicting the rating. As the activation function of the last dense layer, the sigmoid function represented by (11) was used to predict the result as a real number between [0, 1] [39].

$$\sigma\ (\theta) = \frac{1}{1 + e^{-\theta}} \tag{11}$$

The predicted value derived above, is normalized to ensure that it does not exceed the minimum and maximum ranges of the existing ratings. In this process, the normalization method defined in (12) is applied to obtain the final output value. As a result, the normalized predicted rating is within the range of the actual ratings, and the performance of the recommender system is finally evaluated based on this value.

$$4 \times \frac{\hat{y} - min(\hat{y})}{max\ (\hat{y}) - min(\hat{y})} + 1 \tag{12}$$

IV. EXPERIMENTS

A. DATASETS AND EVALUATION METRICS

To evaluate the performance of the proposed model, this study utilized user-review data collected from Amazon.com from May 1996 to October 2018. Amazon is the world's largest e-commerce platform, which consists of various domains and holds a vast amount of online reviews that are being utilized in various research domains [19]. The Amazon data is composed of various information, including user details, item information, ratings, and reviews. More details about the datasets are shown in Table 1. In this study, as detailed in Table 2, we utilized a total of 2,874,654 reviews from the Electronics category, 569,732 reviews from the Automotive category, and a total of 824,167 reviews from the Pet Supplies category. This category is critical for most of the existing approaches due to the high data sparsity, which exceeds 99.9%.

TABLE 1. An example of Amazon review dataset.

Attribute	Value
User ID	A1XSPKZ8HHSBX2
Item ID	073530498X
Rating	5.0
Review Time	01 27, 2018
User Name	Problematic1963
Review Title	great buy
Review	I made a photo album for a senior friend who was moving and I really liked the old camera picture style on this small album. great buy

TABLE 2. Summary of the dataset.

Datasets	#Users	#Items	#Ratings	Sparsity
Electronics	199,299	59,463	2,874,654	99.98%
Automotive	50,176	18,611	569,732	99.94%
Pet Supplies	60,436	16,542	824,167	99.92%

To preprocess the data, we first remove missing values and duplicate data. Since the recommender system learns from interactions between users and items, it's challenging to accurately model users with limited interactions. Accordingly, to optimize model training, items with fewer than 10 reviews have been removed from the dataset, and users with fewer than 10 ratings have also been removed from Amazon datasets [31], [40].

Each dataset was divided into training, validation, and test sets randomly in a 7:1:2 ratio to evaluate the performance of the proposed model. For evaluation metrics, we used the mean absolute error (MAE) and root mean squared error (RMSE) functions, which are commonly used in data analysis research on recommender systems [41]. These metrics are defined in (13) and (14), respectively, where n denotes the total number of data samples; y_i is the actual rating value; and $\hat{y}_i$ is the predicted rating. The lower the values of MAE and RMSE, the smaller the prediction error, which means that the model has good prediction performance.

$$MAE = \frac{1}{n}\sum\nolimits_{i=1}^{n} \left|y_i - \hat{y}_i\right| \tag{13}$$

$$RMSE = \sqrt{\frac{1}{n}\sum\nolimits_{i=1}^{n} (y_i - \hat{y}_i)^2} \tag{14}$$

The MAE gives the same weight regardless of the magnitude of the error between the actual and predicted ratings. However, RMSE is calculated by squaring the error between the actual and predicted ratings, giving greater weight to large errors and less weight to small errors. For this reason, the RMSE is generally preferred over MAE, but the RMSE has the disadvantage of being sensitive to outliers. Therefore, depending on the data characteristics, the MAE may be preferred [42]. In this study, considering the advantages of these two indicators, we simultaneously compared and analyzed the MAE and RMSE values for model performance.

TABLE 3. Explanation of benchmark models.

Model	Description	Reference
MLP	A simple model that predicts user preferences by combining the latent factors of users and items using a MLP structure.	Covington, et al. [45], Cheng, et al. [46]
NeuMF	A model that combines GMF and MLP to predict user preferences by learning linear and non-linear relationships between latent factors of users and items and combining them in the final layer.	He, et al. [32], Chen, et al. [47]
ConvNCF	A model that captures multidimensional interactions between latent factors of users and items through outer product and uses CNN to predict user preferences.	He, et al. [20], Wu, et al. [31]
DeepHCF	A model that predicts user preferences using MLP to learn the interaction between latent factors of users and items, and CNN to extract preference information from review text.	Alfarhood and Cheng [19], Gan and Zhang [48], Aljunid and Doddaghatta Huchaiah [49]

B. EXPERIMENTS

In this study, the natural language toolkit (NLTK) package was used to effectively analyze online reviews. Lemmatization was extracted from the reviews after removing stop-words, special characters, symbols, and numbers [43]. Words with a length of three or less and rare words with a frequency of less than three appearances in the entire review were removed [44].

To evaluate the performance of the ROP-CNN proposed in this study, we compared it with models that have shown good recommendation performance in previous studies, as detailed in Table 3. Since MLP and NeuMF (Neural Matrix Factorization) are proven recommendation models that have shown excellent recommendation performance, we selected them for the final recommendation performance result comparison. The comparison was conducted using ConvNCF, as the objective of this study was to capture the interactions between users and items by utilizing CNN. Finally, the performance was compared with that of DeepHCF, which applies CNN to the review data.

The basic parameters of the proposed model were set as follows. The embedding vector dimension of the CNN model used in the review encoder was set to 300, the filter size to 100, and the kernel sizes to 3, 4, and 5. The review length was fixed at 90% of the maximum review length following previous research [50]. To solve the overfitting problem of the model, a dropout layer with a ratio of 0.1 was added to each dense layer [51]. In the model training optimization process, the adaptive moment estimation (Adam) optimizer was adopted with a learning rate of 0.0005 [52]. The default num-

TABLE 4. Performance comparison with benchmarks.

Dataset	Model	MAE	RMSE
Electronics	MLP	0.855	1.100
	NeuMF	0.725	1.041
	ConvNCF	0.748	1.045
	DeepHCF	0.521	0.862
	ROP-CNN	**0.392**	**0.680**
Automotives	MLP	0.979	1.235
	NeuMF	0.814	1.073
	ConvNCF	0.768	1.002
	DeepHCF	0.635	0.941
	ROP-CNN	**0.376**	**0.614**
Pet Supplies	MLP	1.034	1.266
	NeuMF	0.816	1.100
	ConvNCF	0.839	1.089
	DeepHCF	0.473	0.780
	ROP-CNN	**0.406**	**0.657**

ber of epochs was set to 100, and early stopping was applied to determine the optimal number of training epochs [39]. The experiments were performed in an environment with an Intel Core i9-900KF CPU, 64 GB RAM, and GeForce RTX 2080 Ti, using the TensorFlow and Keras packages.

V. RESULTS

A. PERFORMANCE COMPARISON WITH BENCHMARK

To evaluate the performance of the proposed ROP-CNN model, it was compared with the benchmark models, and the experimental results are shown in Table 4.

The proposed model showed significant performance improvements in both MAE and RMSE over the four benchmark models. Specifically, on the Electronics dataset, the performance of the ROP-CNN model improved by 24% to 54% using the MAE metric and by 21% to 38% using the RMSE metric. On the Automotive dataset, the prediction performance improved by 40% to 61% based on the MAE metric and by 34% to 50% based on the RMSE metric. Similarly, on the Pet Supplies dataset, the prediction performance improved by 14% to 60% using the MAE metric and 15% to 48% using the RMSE metric.

The MLP and NeuMF showed relatively low recommendation accuracy because of the limitations of including user interactions and rating information as the only input sources. Similarly, ConvNCF considers only user interaction and rating information as input information; however, it also considers outer products and CNN to capture richer user-item interactions to learn and predict user preferences. As a result, ConvNCF outperforms MLP and NeuMF. However, since these previous studies all utilized only interaction data and rating information, they have the problem of not fully reflecting users' preferences for items, which limits performance improvement. To overcome this limitation, reviews

TABLE 5. Performance comparison of user-item interaction learning methodology.

Dataset	Embedding Function	MAE	RMSE
Electronics	Concatenate	0.424	0.684
	Element-wise	0.484	0.723
	Outer product	**0.392**	**0.680**
Automotives	Concatenate	0.393	0.676
	Element-wise	0.394	0.645
	Outer product	**0.376**	**0.614**
Pet Supplies	Concatenate	**0.399**	0.690
	Element-wise	0.424	0.665
	Outer product	0.406	**0.657**

that contain user-specific preference information should be considered.

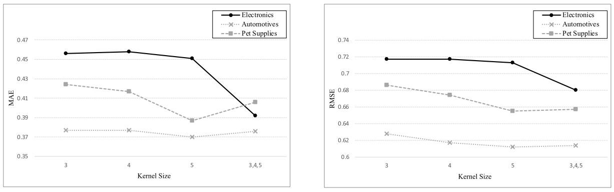
DeepHCF, a model that considers review data, achieved significant performance improvements over previous models because it reflects users' preference information. However, previous studies, such as those on DeepHCF, that use reviews as an auxiliary metric, only focus on methods for embedding reviews. These studies have the limitation that complex latent interactions cannot be learned effectively because they use concatenation or element-wise to learn user-item interactions.

Therefore, in this study, we propose the ROP-CNN model, which can capture interactions between other dimensions by utilizing outer products in the process of learning user-item interactions, allowing the extraction of richer interaction information. ROP-CNN also utilizes reviews as an additional source of information, to enhance the semantic interpretation of user preferences and items. In brief, the proposed model can contribute significantly to improving recommendation performance by extracting user-item interaction information and semantic features existing in the reviews, utilizing CNN for the previously obtained interactions and reviews.

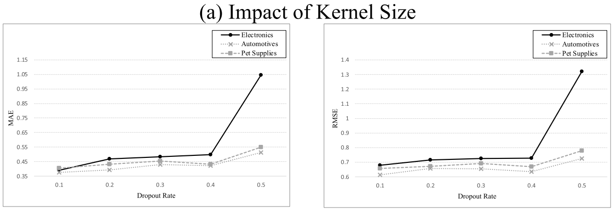
B. EFFICACY OF OUTER PRODUCT

In this study, the outer product was used to model the interaction between user and item. Furthermore, to demonstrate the effectiveness and validity of the outer product, we conducted an additional experiment in which the interaction part was compared with the case of replacing it with concatenation and element-wise. The experimental results (see Table 5) show that the performance of the proposed model, which uses outer product to capture interactions, is superior to those using other operations.

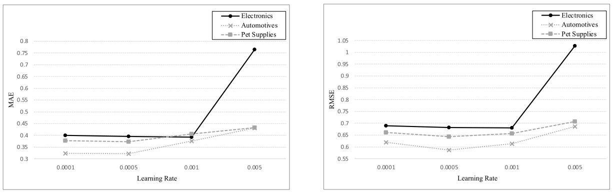
Element-wise refers to the method of multiplying elements in the same position, which has the advantage of low computational cost in capturing direct interactions. However, it has the limitation of only capturing the interaction of a certain dimension, which does not fully reflect the complex user-item interactions in the computation process. Furthermore, concatenation can minimize information loss because it directly



(a) Impact of Kernel Size



(b) Impact of Dropout Rate



(c) Impact of Learning Rate

FIGURE 3. Performance of hyperparameters.

uses the information of two embedding vectors. However, the outer product proposed in this study leads to significantly higher performance compared to other operations. This means that the interaction map generated using outer products can capture both local and global correlations between different embedding dimensions. This feature enables the proposed model to learn more comprehensive interactions between users and items, which leads to an overall improvement in recommendation performance.

C. SENSITIVITY OF HYPERPARAMETERS

In this section, the parameter tuning performed to optimize the model is presented. The proposed model uses CNN to extract semantic features from online reviews. In the experimental process, the effect of different kernel sizes on the recommendation performance was examined [27]. Figure 3(a) shows the MAE and RMSE results for each kernel size, indicating that the extraction of N-gram information from reviews is more effective when various kernel sizes are used rather than a single kernel size. Based on these experimental results, the proposed model utilizes kernel sizes of 3, 4, and 5 as the optimal parameters.

To prevent the proposed model from being overfitted during the training process, a dropout layer was applied to each dense layer. Figure 3(b) shows the experimental performance of the model as a function of dropout ratio. As the dropout ratio increases, the performance decreases due to applying too much constraint to the model, and the best performance is obtained when the dropout ratio is 0.1. Finally, we also experimented with the learning rate by updating the learning weight to optimize the model. According to the experimental results in Figure 3(c), the prediction error is optimal when the learning rate is 0.0005. These results indicate that the problem is not optimized until the end of learning when the learning rate is low, with the solution diverging when the learning rate is high. Therefore, using an appropriate level of learning rate is effective in model optimization. Accordingly, the proposed model used 0.1 for dropout and 0.0005 for the learning rate.

VI. CONCLUSION

A. CONCLUSION

Conventional recommender systems face performance challenges due to data sparsity. To address this issue, methods leveraging review data as an auxiliary indicator have been explored. In this study, we introduced the ROP-CNN model, which applies CNN to both user-item interactions and reviews. Previous studies incorporating reviews primarily focused on text embedding methods and used simple operations for user-item interactions, which is the core of recommendation algorithms. In contrast, this study proposes a deep learning-based methodology capable of directly modeling user-item interactions. The proposed model effectively learns the interaction between user and item feature vectors through CNN, demonstrating superior performance over conventional recommendation algorithms when evaluated on the Amazon dataset. This empirically underscores the necessity for a sophisticated analysis method for both user-item interactions and reviews. Additionally, this study examined the impact of various parameters on performance, providing an interpretation of the results based on hyperparameter tuning.

This study proposes a personalized recommender system that analyzes qualitative preference information in online reviews while simultaneously employing an outer product to capture multidimensional latent user-item interactions. This holds significant theoretical implications for recommender systems. First, conventional systems mostly rely on quantitative user behavior records for recommendation algorithms. However, our study combines these with qualitative data, such as reviews, to construct a personalized recommender system. Multi-channel CNN is applied to review data to extract semantic preference information, including user attributes, preferences, and item characteristics. The model performance, validated with real online data, surpassed that of baseline models. Second, the recommender system predicts ratings by using CNN to learn complex user-item interactions and generates a 2-D interaction map by combining two latent vectors into an outer product. The experimental results indicate the superiority of the outer product used in this study over concatenation and element-wise. Hence, this study is significant in suggesting a new approach for learning interactions and rating prediction, opening venues for further research. Finally, this study delves into the impact of parameters on model performance, providing an interpretation based on parameter value changes.

Furthermore, this study presents several practical implications. First, the proposed recommendation methodology can enhance recommender systems on various e-commerce platforms where users provide reviews, leading to more accurate recommendations, reduced search efforts, and improved user satisfaction. This is expected to positively impact the revenue and profit margins of online platforms. Second, leveraging reviews to identify user preferences can help companies offer personalized services and boost customer satisfaction. Finally, the proposed approach not only provides personalized recommendations but also stimulates new user demands, making it useful in marketing and customer management enterprises.

B. LIMITATION AND FUTURE WORK

Despite the contributions, this study has limitations that future research should address. First, while this study leverages user reviews to predict user preferences, it does not incorporate additional item information such as images, descriptions, prices, or brands. Future research can develop more advanced models by incorporating diverse information. Second, although this study utilized Amazon data to validate the recommendation performance, it is crucial to test the proposed model on a diverse range of online platforms. Evaluations should not be limited to e-commerce data but should also encompass data from the service sector, such as Yelp. Third, this study did not utilize attention mechanisms, which are increasingly recognized for their effectiveness in learning user-item interactions. Future work can adopt attention mechanisms to improve the performance of the proposed model. Additionally, recent advancements in deep learning models, such as CNN-LSTM and vision transformers in natural language processing, can be explored in future research, broadening the scope of embedding representation methods.

REFERENCES

[1] L. Lü, M. Medo, C. H. Yeung, Y.-C. Zhang, Z.-K. Zhang, and T. Zhou, "Recommender systems," *Phys. Rep.*, vol. 519, no. 1, pp. 1–49, 2012.

[2] P. M. Alamdari, N. J. Navimipour, M. Hosseinzadeh, A. A. Safaei, and A. Darwesh, "A systematic study on the recommender systems in the e-commerce," *IEEE Access*, vol. 8, pp. 115694–115716, 2020.

[3] H. Costa and L. Macedo, "Emotion-based recommender system for overcoming the problem of information overload," in *Highlights on Practical Applications of Agents and Multi-Agent Systems*, Salamanca, Spain. Cham, Switzerland: Springer, May 2013, pp. 178–189.

[4] A. Edmunds and A. Morris, "The problem of information overload in business organisations: A review of the literature," *Int. J. Inf. Manage.*, vol. 20, no. 1, pp. 17–28, Feb. 2000.

[5] J. Wei, 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.

[6] X. Li, M. Wang, and T.-P. Liang, "A multi-theoretical kernel-based approach to social network-based recommendation," *Decis. Support Syst.*, vol. 65, pp. 95–104, Sep. 2014.

[7] Y. Yu, L. Zhang, C. Wang, R. Gao, W. Zhao, and J. Jiang, "Neural personalized ranking via Poisson factor model for item recommendation," *Complexity*, vol. 2019, pp. 1–16, Jan. 2019.

[8] 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, Jan. 2020.

[9] D. Lee and K. Hosanagar, "How do recommender systems affect sales diversity? A cross-category investigation via randomized field experiment," *Inf. Syst. Res.*, vol. 30, no. 1, pp. 239–259, Mar. 2019.

[10] I. MacKenzie, C. Meyer, and S. Noble, "How retailers can keep up with consumers," *McKinsey Company*, vol. 18, no. 1, pp. 1–10, 2013.

[11] H. K. Kim, J. K. Kim, and Y. U. Ryu, "Personalized recommendation over a customer network for ubiquitous shopping," *IEEE Trans. Services Comput.*, vol. 2, no. 2, pp. 140–151, Apr. 2009.

[12] X. Su and T. M. Khoshgoftaar, "A survey of collaborative filtering techniques," *Adv. Artif. Intell.*, vol. 2009, pp. 1–19, Oct. 2009.

[13] R. Chen, Q. Hua, Y.-S. Chang, B. Wang, L. Zhang, and X. Kong, "A survey of collaborative filtering-based recommender systems: From traditional methods to hybrid methods based on social networks," *IEEE Access*, vol. 6, pp. 64301–64320, 2018.

[14] Y. Ma, G. Chen, and Q. Wei, "Finding users preferences from large-scale online reviews for personalized recommendation," *Electron. Commerce Res.*, vol. 17, no. 1, pp. 3–29, Mar. 2017.

[15] X. He, T. Chen, M.-Y. Kan, and X. Chen, "TriRank: Review-aware explainable recommendation by modeling aspects," in *Proc. 24th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2015, pp. 1661–1670.

[16] C. Sundermann, M. Domingues, R. Sinoara, R. Marcacini, and S. Rezende, "Using opinion mining in context-aware recommender systems: A systematic review," *Information*, vol. 10, no. 2, p. 42, Jan. 2019.

[17] Z. Zhang, D. Zhang, and J. Lai, "urCF: User review enhanced collaborative filtering," in *Proc. 20th Amer. Conf. Inf. Syst.*, Savannah, GA, USA, Aug. 2014, pp. 1–14.

[18] D. Kim, C. Park, J. Oh, S. Lee, and H. Yu, "Convolutional matrix factorization for document context-aware recommendation," in *Proc. 10th ACM Conf. Recommender Syst.*, Sep. 2016, pp. 233–240.

[19] M. Alfarhood and J. Cheng, "DeepHCF: A deep learning based hybrid collaborative filtering approach for recommendation systems," in *Proc. 17th IEEE Int. Conf. Mach. Learn. Appl. (ICMLA)*, Dec. 2018, pp. 89–96.

[20] X. He, X. Du, X. Wang, F. Tian, J. Tang, and T.-S. Chua, "Outer product-based neural collaborative filtering," 2018, *arXiv:1808.03912*.

[21] G. Adomavicius and Y. Kwon, "Improving aggregate recommendation diversity using ranking-based techniques," *IEEE Trans. Knowl. Data Eng.*, vol. 24, no. 5, pp. 896–911, May 2012.

[22] E. W. Anderson and M. W. Sullivan, "The antecedents and consequences of customer satisfaction for firms," *Marketing Sci.*, vol. 12, no. 2, pp. 125–143, May 1993.

[23] Z. Y. Yao, Y.-K. Park, and T.-H. Hong, "A study on the effect of reviewer's attributes on the usefulness of online review," *J. Inf. Syst.*, vol. 29, no. 2, pp. 173–195, 2020.

[24] J. McAuley and J. Leskovec, "Hidden factors and hidden topics: Understanding rating dimensions with review text," in *Proc. 7th ACM Conf. Recommender Syst.*, 2013, pp. 165–172.

[25] W. Zhang and J. Wang, "Integrating topic and latent factors for scalable personalized review-based rating prediction," *IEEE Trans. Knowl. Data Eng.*, vol. 28, no. 11, pp. 3013–3027, Nov. 2016.

[26] N. Bhojne, S. Deore, R. Jagtap, G. Jain, and C. Kalal, "Collaborative approach based restaurant recommender system using Naive Bayes," *Int. J. Adv. Res. Comput. Commun. Eng*, vol. 6, no. 4, pp. 6–13, 2017.

[27] Y. Kim, "Convolutional neural networks for sentence classification," 2014, *arXiv:1408.5882*.

[28] L. Zheng, V. Noroozi, and P. S. Yu, "Joint deep modeling of users and items using reviews for recommendation," in *Proc. 10th ACM Int. Conf. Web Search Data Mining*, Feb. 2017, pp. 425–434.

[29] C. Chen, M. Zhang, Y. Liu, and S. Ma, "Neural attentional rating regression with review-level explanations," in *Proc. World Wide Web Conf. World Wide Web (WWW)*, 2018, pp. 1583–1592.

[30] H. Liu, Y. Wang, Q. Peng, F. Wu, L. Gan, L. Pan, and P. Jiao, "Hybrid neural recommendation with joint deep representation learning of ratings and reviews," *Neurocomputing*, vol. 374, pp. 77–85, Jan. 2020.

[31] Y. Wu, J. Wei, J. Yin, X. Liu, and J. Zhang, "Deep collaborative filtering based on outer product," *IEEE Access*, vol. 8, pp. 85567–85574, 2020.

[32] 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.

[33] X. He and T.-S. Chua, "Neural factorization machines for sparse predictive analytics," in *Proc. 40th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Aug. 2017, pp. 355–364.

[34] S. Lee and D. Kim, "Deep learning based recommender system using cross convolutional filters," *Inf. Sci.*, vol. 592, pp. 112–122, May 2022.

[35] L. Han, H. Wu, N. Hu, and B. Qu, "Convolutional neural collaborative filtering with stacked embeddings," in *Proc. Asian Conf. Mach. Learn.*, 2019, pp. 726–741.

[36] R. van Dinter, C. Catal, and B. Tekinerdogan, "A multi-channel convolutional neural network approach to automate the citation screening process," *Appl. Soft Comput.*, vol. 112, Nov. 2021, Art. no. 107765.

[37] N. Majumder, S. Poria, A. Gelbukh, and E. Cambria, "Deep learning-based document modeling for personality detection from text," *IEEE Intell. Syst.*, vol. 32, no. 2, pp. 74–79, Mar. 2017.

[38] H. Chen, F. X. Han, D. Niu, D. Liu, K. Lai, C. Wu, and Y. Xu, "MIX: Multi-channel information crossing for text matching," in *Proc. 24th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Jul. 2018, pp. 110–119.

[39] K. R, P. Kumar, and B. Bhasker, "DNNRec: A novel deep learning based hybrid recommender system," *Expert Syst. Appl.*, vol. 144, Apr. 2020, Art. no. 113054.

[40] J.-L. Chang, H. Li, and J.-W. Bi, "Personalized travel recommendation: A hybrid method with collaborative filtering and social network analysis," *Current Issues Tourism*, vol. 25, no. 14, pp. 2338–2356, Jul. 2022.

[41] T. Chai and R. R. Draxler, "Root mean square error (RMSE) or mean absolute error (MAE)—Arguments against avoiding RMSE in the literature," *Geoscientific Model Develop.*, vol. 7, no. 3, pp. 1247–1250, Jun. 2014.

[42] M. Lee, W. Kwon, and K.-J. Back, "Artificial intelligence for hospitality big data analytics: Developing a prediction model of restaurant review helpfulness for customer decision-making," *Int. J. Contemp. Hospitality Manage.*, vol. 33, no. 6, pp. 2117–2136, Aug. 2021.

[43] L. Hickman, S. Thapa, L. Tay, M. Cao, and P. Srinivasan, "Text preprocessing for text mining in organizational research: Review and recommendations," *Organizational Res. Methods*, vol. 25, no. 1, pp. 114–146, Jan. 2022.

[44] M. K. Sohrabi and F. Hemmatian, "An efficient preprocessing method for supervised sentiment analysis by converting sentences to numerical vectors: A Twitter case study," *Multimedia Tools Appl.*, vol. 78, no. 17, pp. 24863–24882, Sep. 2019.

[45] 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.

[46] H.-T. Cheng, "Wide & deep learning for recommender systems," in *Proc. 1st Workshop Deep Learn. Recommender Syst.*, 2016, pp. 7–10.

[47] W. Chen, F. Cai, H. Chen, and M. D. Rijke, "Joint neural collaborative filtering for recommender systems," *ACM Trans. Inf. Syst.*, vol. 37, no. 4, pp. 1–30, Oct. 2019.

[48] M. Gan and H. Zhang, "DeepFusion: Fusing user-generated content and item raw content towards personalized product recommendation," *Complexity*, vol. 2020, pp. 1–12, Mar. 2020.

[49] M. F. Aljunid and M. Doddaghatta Huchaiah, "Multi-model deep learning approach for collaborative filtering recommendation system," *CAAI Trans. Intell. Technol.*, vol. 5, no. 4, pp. 268–275, Dec. 2020.

[50] J. Dong, F. He, Y. Guo, and H. Zhang, "A commodity review sentiment analysis based on BERT-CNN model," in *Proc. 5th Int. Conf. Comput. Commun. Syst. (ICCCS)*, May 2020, pp. 143–147.

[51] 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.

[52] R. Catherine and W. Cohen, "TransNets: Learning to transform for recommendation," in *Proc. 11th ACM Conf. Recommender Syst.*, Aug. 2017, pp. 288–296.

SEIN HONG received the B.S. degree in mathematics from Kwangwoon University, in 2022. She is currently pursuing the M.S. degree in big data analytics with Kyung Hee University. Her research interests include NLP, recommender systems, big data analytics, and machine learning.

XINZHE LI received the M.S. degree in big data analytics from Kyung Hee University, where he is currently pursuing the Ph.D. degree. He has published numerous articles, which have appeared in *Applied Artificial Intelligence*, *Applied Sciences*, *Journal of Hospitality and Tourism Technology*, and *Data Technologies and Applications*. His current research interests include recommender systems, deep learning, online reviews, and natural language processing.

SIGEON YANG received the B.S. degree in hospitality management from Sejong University, in 2020. He is currently pursuing the M.S. degree in big data analytics with Kyung Hee University. He has published articles on IEEE ACCESS. His research interests include text mining, recommender systems, big data analytics, and machine learning.

JAEKYEONG KIM received the B.S. degree in industrial engineering from Seoul National University, and the M.S. and Ph.D. degrees in management information systems (MIS) from Korea Advanced Institute of Science and Technology (KAIST). He is currently a Professor with the School of Management, Kyung Hee University. He has published numerous articles which have appeared in IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS, IEEE TRANSACTIONS ON SERVICES COMPUTING, *Information & Management*, *International Journal of Information Management*, *Applied Artificial Intelligence*, *Artificial Intelligence Review*, *International Journal of Human-Computer Studies*, *Expert Systems with Applications*, and *Group Decision and Negotiation*. His current research interests include business analytics, recommender systems, big data analysis, and deep learning.