IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 25 July 2025, accepted 11 August 2025, date of publication 14 August 2025, date of current version 21 August 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3598937

RESEARCH ARTICLE

Utilizing Advanced Transformers for Summarized Review-Aware Recommender System

SUJI KIM[1], LI JIN[1], QINGLONG LI[2], XINZHE LI[1], CHULKYOON YOON[3], AND JAEKYEONG KIM[1,4]

[1]Department of Big Data Analytics, Kyung Hee University, Seoul 02447, Republic of Korea
[2]Division of Computer Engineering, Hansung University, Seoul 02876, Republic of Korea
[3]Department of Business Administration, Graduate School, Kyung Hee University, Seoul 02447, Republic of Korea
[4]School of Management, Kyung Hee University, Seoul 02447, Republic of Korea

Corresponding author: Chulkyoon Yoon (aa2508@khu.ac.kr)

ABSTRACT With the rapid growth of e-commerce and increase in the volume of online user reviews, personalized recommender systems have become essential for helping users efficiently identify items of interest. Although user review sets offer rich information for capturing user preferences and item characteristics, full-length reviews sometimes introduce noise, redundancy, and high computational costs. These issues degrade the recommendation performance. To address these challenges, this study proposes a summarized review-aware recommender system (SRARS), which leverages abstractive summarization to extract essential information from review texts. Specifically, we employed the BART, bidirectional and autoregressive transformer model to generate concise and coherent summaries of user and item reviews, thus improving the representation of preferences and attributes. To further enhance recommendation accuracy, we modeled user–item interactions through an outer product operation that captures complex multidimensional relationships and refined these interactions using an attention mechanism to emphasize informative patterns. Experimental evaluations conducted on real-world review data from Amazon.com demonstrated that SRARS significantly outperformed the baseline models in terms of recommendation accuracy and efficiency. This study contributes to the development of review-aware recommender systems by integrating text summarization and interaction modeling to improve the performance under sparse data conditions.

INDEX TERMS Recommender system, BART, summarized review, user reviews, data sparsity.

I. INTRODUCTION

Advancements in information and communication technology have facilitated rapid growth in the online e-commerce market. This environment enables users to access a wide range of items and services conveniently. As users actively share their item evaluations, the volume of available information has increased exponentially [1]. However, an excessive volume of information often hinders users from efficiently identifying items that meet their needs, resulting in significant time consumption [2]. Consequently, the demand for personalized recommender systems that recommend items based on individual user preferences has emerged [3], [4]. Major platforms such as Amazon, YouTube, and Netflix have incorporated recommender systems to enhance user experiences, thereby improving user retention and increasing sales [5]. Notably, Netflix significantly reduced subscriber churn and achieved annual cost savings of over \$ 1 billion following the implementation of a recommender system [6], [7]. Consequently, research on recommender systems has been actively pursued in a wide range of academic and industrial fields.

Early recommender systems were mainly developed based on collaborative filtering (CF), which assumes that users with similar historical behaviors (e.g., ratings, clicks, etc.) are likely to have similar preferences [8], [9]. Since CF is effective to implement and has demonstrated strong performance, it remains one of the most widely adopted recommendation techniques across various domains. However, in recent years,

The associate editor coordinating the review of this manuscript and approving it for publication was Arianna DUlizia.

most internet platforms encourage customers to write reviews on products and services, and many studies using review data have been conducted. Overall rating data typically reflect users' overall evaluations of an item; however, they are limited in their ability to capture the nuanced aspects of individual user preferences. In contrast, review texts capture users' personal experiences, including the reasons for their purchase decisions and the specific aspects they appreciated or found unsatisfactory. This enables a more detailed and explicit representation of user preferences [10], [11]. By using review sets from multiple users, it is possible to gain a deep understanding of user preferences and item characteristics from various perspectives, thereby enabling more precise recommendations [12].

Zheng et al. [13] extracted user and item features from their respective review sets and jointly trained them to predict user preferences. Similarly, Chen et al. [10] assigned varying weights to individual reviews within review sets based on their relative importance. This approach effectively incorporated salient information to generate more accurate recommendations. Moreover, Wu et al. [14] demonstrated improved recommendation performance by integrating key information extracted from review texts with user–item interaction data for rating prediction. Review set-based recommender systems leverage the rich information embedded in review texts, thereby alleviating data sparsity and enhancing the accuracy and reliability of personalized recommendations [12], [15].

However, recommender systems that utilize full-review texts face several challenges. First, review texts often include redundant or irrelevant content, which may obscure essential information or distort the original intent of the text [16], [17]. This may hinder the system's ability to extract meaningful representations of user and item characteristics accurately. Second, processing large volumes of review data incurs substantial time and computational costs, thereby reducing system efficiency [18], [19]. To gain user trust, a recommender system must ensure high accuracy, efficient resource utilization, and acceptable processing times [16]. Therefore, this study suggests that summarizing review texts can effectively address these limitations and highlights the importance of utilizing review information more efficiently.

Text summarization methods are generally categorized into extractive and abstractive approaches. Extractive summarization constructs summaries by directly selecting salient words or sentences from the original text based on criteria such as frequency or similarity without altering the content [20]. This method offers the advantage of simplicity and enables the rapid extraction of key information. However, this study has several limitations. Because it directly extracts segments from the original text, extractive summarization may generate summaries that are unnatural or lack coherence. Additionally, it often selects overlapping or similar content, resulting in redundancy and weakened alignment with the overall context [20], [21], [22]. When the generated summaries do not adequately reflect the content of the original reviews, it becomes difficult to capture user preferences and item characteristics accurately, thereby limiting the effectiveness of personalized recommendations. By contrast, abstractive summarization generates novel sentences that convey the core meaning of the original text. This approach yields concise and coherent summaries while minimizing redundancy [23]. When applied to recommender systems, abstractive summarization facilitates the generation of concise and fluent summaries. It effectively removes redundant or irrelevant content while preserving the information critical for identifying user preferences and item characteristics. Therefore, this study adopted an abstractive summarization approach to generate review summaries, leveraging its advantages in enhancing recommendation quality.

This study proposed a model called the summarized review-aware recommender system (SRARS), which aims to enhance the accuracy of preference prediction by summarizing essential information from user review sets. First, we employed a bidirectional and autoregressive transformer (BART), which can generate more natural and flexible summaries than traditional models, to summarize the review sets. The BART functions as a denoising autoencoder that removes irrelevant content from the original text and captures essential information while concisely summarizing the complex input [24]. By summarizing the review sets, the model can represent the user and item characteristics more effectively. Second, we applied an outer product operation to model the interactions between users and items. This operation captures the multidimensional interactions between feature vectors, enabling the identification of complex relationships between users and items. Finally, we incorporated an attention mechanism into the interaction map generated by the outer product operation. This mechanism emphasizes important interactions while suppressing less relevant interactions, thereby refining the prediction of user preferences for specific items. To validate the effectiveness of the proposed model, we conducted experiments using user review data collected from the e-commerce platform Amazon.com. The experimental results indicate that SRARS outperformed the baseline models in terms of recommendation accuracy. The main contributions of this study are summarized as follows:

- This study addresses the limitations of full-length reviews by effectively extracting user and item characteristics from review sets using BART, thereby improving recommendation accuracy and efficiency.
- We comprehensively captured the multidimensional interactions between user and item representations using an outer product operation. Furthermore, by incorporating an attention mechanism, we emphasize critical interactions, resulting in a more refined understanding of user–item relationships.
- The proposed model outperformed baseline methods on real-world datasets. In addition, we validated the effectiveness of the model and the impact of its components through extensive experiments.

The remainder of this paper is organized as follows. Section II reviews the related work. Section III presents a detailed description of the proposed model. Section IV describes the dataset, evaluation metrics, experimental design, and environment. Section V presents experimental results. Finally, Section VI concludes the paper and discusses limitations and directions for future research.

II. RELATED WORKS

A. RECOMMENDER SYSTEMS UTILIZING REVIEW SETS

With advances in deep-learning techniques, various methods have been developed to extract semantic information from review texts and enhance recommendation performance. When reviews from multiple users are comprehensively utilized, user preferences and item characteristics can be understood from various perspectives. This has led to the ongoing research on the use of review sets. Zheng et al. [13] proposed a recommender system that utilizes review sets to address the data sparsity issue in traditional CF-based recommendation systems. To the best of our knowledge, this study is the first to jointly learn user preferences and item characteristics from user and item review sets, experimentally demonstrating that utilizing review texts can alleviate data sparsity. In addition, previous studies introduced attention mechanisms to focus on important information while removing irrelevant noise from reviews. These studies focused on analyzing user preferences and item characteristics using review sets. Chen et al. [10] proposed a model that assigned higher weights to significant reviews within a review set to reflect their varying usefulness. This study was the first to introduce the concept of review usefulness into recommender systems and confirmed that considering review usefulness leads to a more accurate preference extraction than not accounting for the varying importance of reviews. Seo et al. [25] approached this problem using local and global attention mechanisms to reflect local and global contexts from review sets. This approach aims to comprehensively understand user preferences and item characteristics. The experimental results indicated that combining local and global attention allowed for clearer learning of user and item characteristics compared to using each attention type individually. Other studies focused on aspects that users find important regarding items. Aspect-aware studies analyzed the overall content of review texts and focused on identifying the specific aspects (e.g., quality, price, and design) that users consider important for certain items to provide more suitable recommendations. For instance, Li and Xu [26] analyzed each word in review sets from local and global perspectives to highlight significant aspects relevant to user preferences. In this study, experiments conducted using fashion-related data showed that determining the optimal number of aspects significantly affected model performance. Furthermore, Zhang et al. [27] demonstrated that emphasizing the important aspects of user review sets can enhance interpretability through highlighted review keywords and improve recommendation performance.

These review set-based studies have contributed to alleviating data sparsity and improving the recommendation performance by effectively learning user preferences and item characteristics [12], [28]. However, using all the reviews without filtering may lead to redundant or unnecessary information. This creates noise during the data processing, rendering it difficult for recommendation systems to capture user and item characteristics accurately. Consequently, the model performance may degrade [16]. Additionally, longer review data demand more computational resources and processing time, which negatively impacts the efficiency of recommendation systems [18], [19]. These issues can be addressed by summarizing reviews that retain their overall meaning while condensing important content. Therefore, this study aims to overcome the limitations of using entire reviews by summarizing and processing them. By utilizing summarized reviews, this study seeks to capture user and item features more clearly and predict user preferences more accurately.

B. REVIEW SUMMARIZATION-BASED RECOMMENDER SYSTEMS

Summarization condenses important content while retaining the meaning [29]. Previous summarization-based studies used sentence similarity or word frequency methods to extract important sentences [22]. Roul and Arora [30] employed Fuzzy C-means clustering to group similar reviews into clusters and then selected significant sentences from each cluster to generate summaries. Bai et al. [29] proposed a model using the graph-based algorithm TextRank to calculate the similarities between sentences and generate review summaries. The experimental results showed that summarized reviews improved recommendation performance and reduced computational costs compared with unsummarized reviews. This demonstrates the necessity and efficiency of summarization. However, relying solely on word frequency or sentence similarity may result in summaries lacking proper context, leading to unnatural or overly lengthy summaries that lack coherence [20], [21]. Abstractive summarization was explored to address these limitations.

Abstractive summarization generates new sentences by reflecting the context of the input text, which helps to minimize redundancy while maintaining the essence of the content [7], [11]. Through experiments, Xu et al. [31] showed that abstractive summarization can effectively capture the essence of reviews, producing summaries of similar quality to those written by humans. BART is one of the most widely used models for abstractive summarization [24]. BART is a pretrained sequence-to-sequence model that restores the input text after adding noise, such as by masking parts of the text or altering their order [24], [32]. It combines bidirectional transformers with autoregressive transformers, enabling it to understand context and generate coherent sentences. This enables the BART to produce more natural and consistent summaries than other summarization models [11]. In this study, BART was used to overcome the limitations of previous summarization-based research and effectively capture important information through consistent and accurate summaries. This helps to identify user and item characteristics, leading to more precise user preference predictions.

C. USER-ITEM INTERACTION

Recommendation systems aim to analyze the interactions between users and items to recommend the most suitable items to users [4]. To achieve this goal, many studies have attempted to effectively model complex interactions between users and items. A representative technique, MF, captures interactions by computing the relationship between user and item vectors. MF has been widely used in recommendation systems because it effectively captures user–item interactions. This technique is computationally simple and can be processed quickly even with large datasets [33]. Furthermore, various recommendation models that extend the MF have been proposed to capture more complex interactions. For example, Mnih and Salakhutdinov [34] proposed probabilistic MF (PMF), which uses Gaussian distributions to probabilistically estimate the latent factor vectors of users and items. This technique enhances traditional MF by addressing overfitting and data sparsity, resulting in a more generalized performance. He et al. [35] introduced a multilayer perceptron (MLP) to overcome the limitations of traditional MF, which models only linear interactions between users and items. Using the MLP, the model can learn nonlinear interactions. Experiments showed that this approach improved the recommendation performance compared with traditional MF-based methods. However, MF-based methods remain limited because they use an inner product to combine user and item feature vectors, which does not explicitly model the correlations between embedding dimensions. MF assumes that each embedding dimension is independent and has the same impact on the prediction. However, item attributes are often interdependent, and embedding dimensions can represent specific attributes of items. Therefore, this assumption is unrealistic [36]. To address this limitation, recommendation models using outer product operation have been proposed to explicitly capture the correlations between the embedding dimensions.

The outer product can learn all the pairwise correlations between the embedding dimensions. This approach captures comprehensive and higher-order relationships, which reflect more detailed interactions between users and items [36], [37]. He et al. [36] proposed a recommendation model that combines the latent factor vectors of users and items through outer product-to-model interactions, followed by applying a convolutional neural network (CNN) to the generated interaction map to learn the interactions between embeddings. This study demonstrated that the outer product operation effectively captures high-order user–item interactions and that modeling complex correlations between embedding dimensions can improve recommendation performance. Lee and Kim [37] combined the user and item feature vectors using an outer product to generate an interaction map. They processed the data with cross-convolutional filters to assign greater weight to significant interactions. This study confirmed that the outer product can learn meaningful high-order interactions between users and items, showing superior performance compared to existing deep-learning models that only capture linear interactions. Therefore, this study leverages the advantages of outer product modeling of user–item interactions. This approach aims to improve the recommendation performance more precisely by learning high-order interactions between users and items.

III. SRARS: SUMMARIZED REVIEW-AWARE RECOMMENDER SYSTEM

Existing review set-based recommendation systems use entire reviews, which often contain repetitive or irrelevant information. This results in longer processing times and higher computational resource consumption. To address these limitations, this study proposed the SRARS model, which removes unnecessary content from reviews and extracts important information to improve efficiency. The proposed model consists of three main modules: review feature extraction, interaction learning, and rating prediction modules, as shown in Figure 1. The review feature extraction module aims to accurately capture user preferences and item features through user reviews. We use BART, a pretrained language model, to remove redundant information and noise from each review set and extract the embedding vectors. The interaction-learning module focuses on capturing intricate interactions between users and items. It combines previously generated user and item feature vectors using the outer product, thereby capturing the high-order interactions between them. The resulting interaction map is then processed using a multihead attention mechanism to emphasize important interactions. Finally, the rating prediction module predicts the final rating that a user assigns to an item based on the interactions learned from the previous module. Each module is described in detail in the following subsections.

A. REVIEW FEATURE EXTRACTION MODULE

The review feature extraction module aims to extract user preferences and item features by summarizing important information from review sets using BART, a pretrained language model. User reviews often contain valuable information such as motivations, interests, and experiences, which can help create a more detailed understanding of users and items [11], [12]. However, the use of entire review sets without filtering can lead to noise owing to redundant or irrelevant information. This noise can distort the meaning or interfere with important details, thereby reducing the performance of the recommendation model [17], [18]. In the process of summarizing review texts, it is crucial to preserve the user's original intent and sentiment embedded in the content. Among various summarization models, BART stands out for its lexical diversity and semantic appropriateness. It can generate coherent summaries that accurately reflect the core information of the original reviews [38], [39]. These benefits

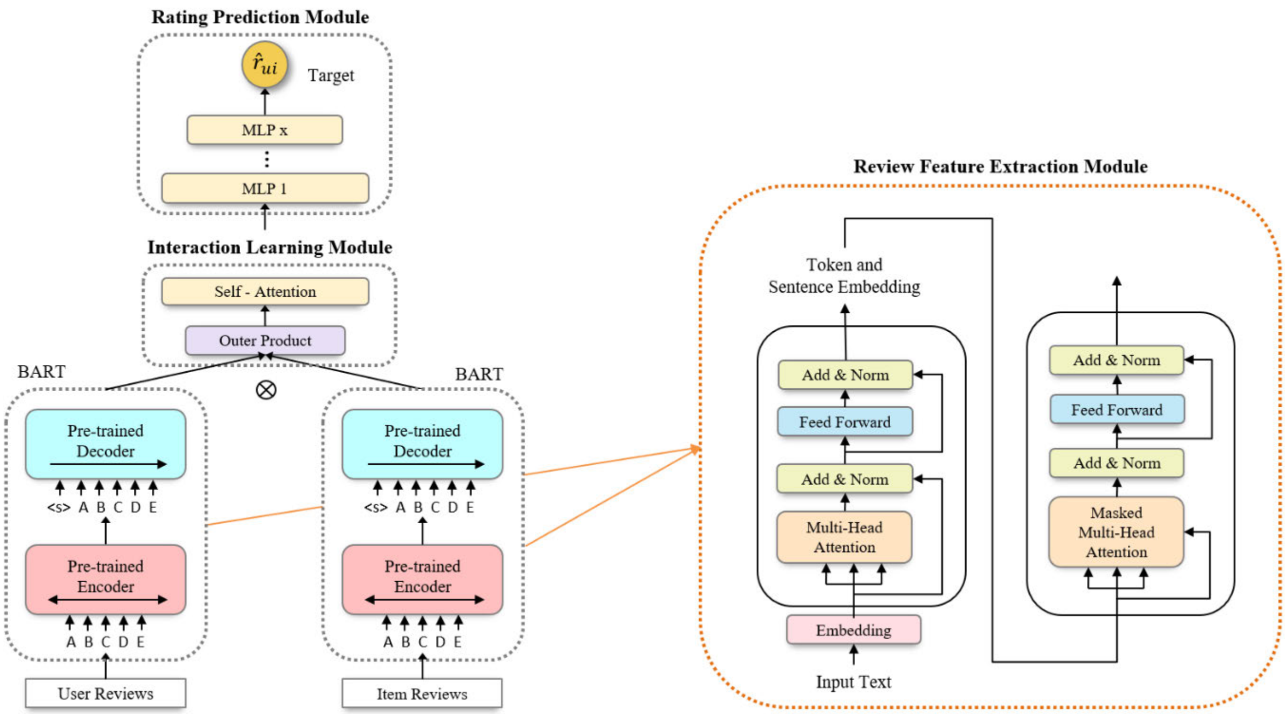

FIGURE 1. **The overall framework of the proposed model.**

are significant in recommender systems, where a nuanced understanding of user opinions directly affects performance. Therefore, we adopted BART to extract essential information from review texts, which can minimize noise and redundancy to improve the efficiency of text representation.

BART is a natural language processing model known for its strong performance in tasks such as text summarization, machine translation, and document restoration. It combines a bidirectional encoder with an autoregressive decoder to understand context and generate text. This hybrid architecture allows BART to model complex textual dependencies and generate high-quality summaries suited for downstream recommendation tasks. The model tokenizes input text sequences $S = \{w_1, w_2, \ldots, w_n\}$ using byte pair encoding and learns by restoring corrupted text through added noise, such as text infilling and sentence permutation [24], [32]. Text infilling removes parts of the original text and replaces them with [MASK] tokens, encouraging the model to learn the overall context by restoring missing segments. For instance, the original text "Summarized Review-Aware Recommender System" could be modified to "Summarized [MASK] System." Sentence permutation involves shuffling the order of sentences to train the model to understand logical sequence, such as transforming "Summarized Review-Aware Recommender System" to "Recommender System Summarized Review-Aware." The bidirectional encoder captures the complete context of the corrupted text, and the autoregressive decoder predicts subsequent tokens based on preceding tokens to restore the original content. With this pretraining process, BART can be applied effectively to various downstream tasks. In sequence classification tasks, the same input is processed by both the encoder and the decoder, and the hidden state of the final token generated by the decoder is used for multiclass classification. This final hidden state represents the core contextual information and can be used effectively for classification tasks. In the review feature extraction module, the same approach is used to extract summarized information from user and item reviews, which is then employed as user preferences and item feature vectors.

First, each review of both users and items is represented as follows:

$$S_i^u = \left\{w_{u,1}, w_{u,2}, \ldots, w_{u,n}\right\}, \tag{1}$$
$$S_i^v = \left\{w_{v,1}, w_{v,2}, \ldots, w_{v,m}\right\}, \tag{2}$$

where S_i^u represents the i-th review of user u, and S_i^v represents the i-th review of item. $w_{u,n}$ and $w_{v,m}$ denote the n-th and m-th words of each review, respectively, and represent the length of the review. As shown in Equations (3) and (4), individual reviews are aggregated by user and item to generate a review set sequence.

$$D^u = \left\{S_1^u, S_2^u, \ldots, S_j^u\right\}, \tag{3}$$
$$D^v = \left\{S_1^v, S_2^v, \ldots, S_k^v\right\}, \tag{4}$$

where D^u represents the review set of user u, and j denotes the total number of reviews written by user u. Similarly, D^v represents the review set of item v, where k represents the total number of reviews of item v. The generated review sets are then used with the BART, as shown in Equations (5) and (6), to extract the user and item feature vectors.

$$U_0 = BART(D^u), \quad (5)$$
$$I_0 = BART\left(D^v\right), \quad (6)$$

where the outputs of the equations, $U_0 \epsilon \mathbb{R}^d$ and $I_0 \epsilon \mathbb{R}^d$ represent the user feature vector and item feature vector, respectively, which are extracted by summarizing the review sets using BART. Here, d represents the dimensionality of the vector output obtained using the BART model[1]. This study used BART-base, and an MLP was subsequently applied to reduce the dimensionality of the summarized feature vectors. The MLP used in this study is defined in Equation (7) and the detailed process is shown in Equation (8).

$$X_L = MLP(X_0), \quad (7)$$
$$X_1 = \delta_1\left(W_1^T X_0 + b_1\right),$$
$$\ldots$$
$$X_L = \delta_L\left(W_L^T X_{L-1} + b_L\right), \quad (8)$$

X_0 is the input value of the MLP, and X_L is the final output value of the MLP. W_L^T, b_L, and δ_L represent the weights, bias, and activation function of the L-th layer, respectively, whereas T denotes the transpose. In this study, the activation function used is ReLU, which can be expressed by Equation (9).

$$ReLU\,(x) = max\,(0,x)\,, \quad (9)$$

The user and item feature vectors are fed into the MLP, as shown in Equations (10) and (11):

$$U_L = MLP\,(U_0)\,, \quad (10)$$
$$I_L = MLP\,(I_0)\,, \quad (11)$$

the outputs of the MLP, $U_L \epsilon \mathbb{R}^{d'}$ and $I_L \epsilon \mathbb{R}^{d'}$, represent the final generated user and item feature vectors, respectively. Here, d' represents the dimensionality of the final output of the MLP.

B. INTERACTION LEARNING MODULE

The interaction-learning module aims to effectively model the interactions between users and items by accepting the previously extracted user and item feature vectors as inputs and applying an outer product and attention mechanism. This module focuses on precisely capturing complex interactions in the data.

1) OUTER PRODUCT LAYER

Traditional recommendation system models based on inner product have limitations in capturing complex relationships between users and items because they only reflect linear combinations of interactions. To address this issue, our study utilized the outer product to learn richer interactions between the user and item feature vectors. The outer product can capture individual interactions between the elements of the vectors and complex interactions across different dimensions, thus allowing explicit learning of the relationships between the two vectors [36], [37]. Modeling interactions explicitly enhance the recommendation performance and enable deep-learning models to learn effectively, even in sparse data scenarios [40]. The detailed process is shown in Figure 2.

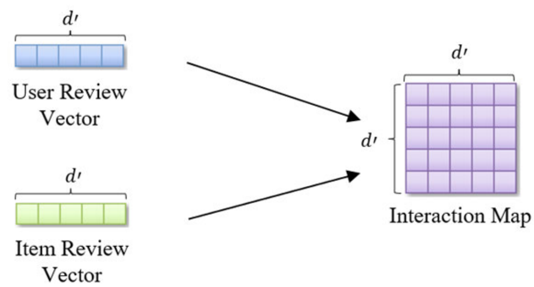

FIGURE 2. Outer product.

First, the user feature vector, U_L and item feature vector, I_L, which were generated previously, are combined using the outer product to create an interaction map. This process is expressed by Equation (12).

$$M = U_L \otimes I_L = U_L I_L^T, \quad (12)$$

where $\otimes$ denotes the outer product, and T represents the transpose. $M \epsilon \mathbb{R}^{d' \times d'}$ is the interaction map generated through the outer product, where each element represents the product of the elements from U_L and I_L.

2) MULTIHEAD SELF-ATTENTION LAYER

To further refine the user–item interaction representation, this study introduces a multihead self-attention layer on top of the interaction map generated by the outer product. This layer assigns varying importance to different interaction signals, enabling the model to focus on the most informative dimensions while suppressing irrelevant or noisy signals. By employing multiple attention heads, the model can capture diverse interaction patterns from multiple perspectives, improving its ability to extract nuanced relationships between users and items. Previous studies have attempted to apply CNN to the interaction map by treating it as an image. However, Ferrari Dacrema, Parroni [41] showed that changing the order of rows and columns in an interaction map does not significantly affect the performance. This indicates that the interaction map has properties that are different from those of the images, rendering the CNN less effective. In contrast, the attention mechanism computes pairwise relationships between all positions in the interaction map. This allows the model to dynamically learn which interactions contribute most to the recommendation task. Therefore, the self-attention mechanism enhances the representation

[1] https://huggingface.co/facebook/bart-base

capacity and provides a flexible way to model global dependencies and complex cross-dimensional correlations.

The attention mechanism process is described as follows: First, the query, key, and value vectors are generated for the interaction map M from each attention head, as expressed in Equation (13).

$$Q_i = M \times W_i^Q, K_i = M \times W_i^K, V_i = M \times W_i^V, \tag{13}$$

$$O_i = Attention\left(Q_i, K_i, V_i\right) = softmax\left(\frac{Q_i K_i^T}{\sqrt{d}}\right) V_i, \tag{14}$$

where Q_i, K_i, and V_i represent the query, key, and value vectors of the i-th head, respectively, and W_i^Q, W_i^K, and W_i^V are the weight matrices for the Q, K, and V of the i-th head, respectively. The attention mechanism calculates the attention score by computing the dot product of the query and key, as expressed in Equation (14), and subsequently converts it into a probability value using the softmax function to represent the relevance of each query and key. O_i represents the attention score calculated in the i-th head, and the outputs from each head are concatenated and linearly transformed using a projection matrix, as expressed in Equation (15).

$$O = MultiHead\left(M\right) = Concatenate\left(O_1, O_2, \ldots, O_h\right) W^o, \tag{15}$$

where W^o denotes the projection matrix and O is the final output of the multihead attention, representing the emphasized important interactions between users and items.

C. RATING PREDICTION MODULE

Finally, the rating prediction module aims to predict user ratings, representing user preferences, based on the final vector output of the interaction-learning module. The final vector, O, generated previously, is used as the input to the MLP for the rating prediction. The MLP is identical to that defined in Equation (7).

$$Z_L = MLP\left(O\right), \tag{16}$$

In the last layer, a linear transformation is applied to predict the rating, as expressed in Equation (17).

$$\hat{r}_{ui} = W_N^T Z_L + b_N, \tag{17}$$

where W_N^T and b_N denote the weight and bias of the final layer, respectively. The final output, denoted as $\hat{r}_{ui}$, represents the predicted rating provided by the user.

D. OPTIMIZATION

To train the parameters of the proposed model, the mean squared error (MSE) is used as the loss function. The MSE is widely used in recommendation systems. It calculates the squared difference between the predicted and actual values and then averages them, as expressed in Equation (18) [25], [26]:

$$MSE = \frac{1}{n} \sum\nolimits_{i=1}^{n} \left(r_{ui} - \hat{r}_{ui}\right)^2, \tag{18}$$

where N represents the number of predicted ratings, r_{ui} is the actual rating, and $\hat{r}_{ui}$ is the predicted rating. This study used an adaptive moment estimation (Adam) optimizer to minimize the objective function.

IV. EXPERIMENTS

In this section, we evaluate the performance of the proposed model. The experiments were designed to answer the following research questions.

- RQ1: Does the proposed model outperform baseline models?
- RQ2: Is representing users and items using only summarized review information without latent factor vectors effective in improving recommendation performance?
- RQ3: Does using the outer product and attention mechanism to capture user–item interactions contribute to improved recommendation performance?
- RQ4: How does the sequence length of the BART model affect the recommendation performance?

A. DATASETS

Online review data collected from Amazon.com was used to validate the proposed model. The data were divided into several categories. In this study, we used three datasets: Video Games, Industrial and Scientific, and Musical Instruments. To measure the performance of the proposed model effectively, we filtered the customer data to include only those who left five or more reviews. The data were split into training, validation, and test datasets with a ratio of 7:1:2, respectively [42]. The statistics for each dataset is presented in Table 1.

TABLE 1. Summary of the datasets.

Dataset	Number of reviews	Number of users	Number of items	Sparsity
Musical Instruments	774,268	82,878	148,819	99.372%
Industrial and Scientific	867,300	102,925	238,768	99.647%
Video Games	545,025	59,833	47,061	98.064%

B. EVALUATION METRICS

To evaluate the performance of the proposed model, we used two widely adopted metrics in recommendation system research: the mean absolute error (MAE) and root mean squared error (RMSE). The MAE was calculated by summing the absolute errors between the predicted and actual ratings and dividing them by the number of evaluation subjects, as expressed in Equation (19). This metric assigns equal weights to all errors, regardless of their magnitude. The RMSE was computed by calculating the square root of the average squared errors, that is, the MSE between the predicted and actual ratings, as expressed in Equation (20).

Compared with the MAE, the RMSE assigns a greater weight to larger errors and is more sensitive to large deviations.

$$MAE = \frac{1}{N} \sum_{i=1}^{N} \left| y_i - \hat{y}_i \right|, \tag{19}$$

$$RMSE = \sqrt{\frac{1}{N} \sum_{i=1}^{N} \left(y_i - \hat{y}_i \right)^2}. \tag{20}$$

In Equations (19) and (20), y_i represents the actual rating, $\hat{y}_i$ the predicted rating, and N represents the number of evaluation subjects.

C. BASELINE MODELS

To evaluate the performance of the proposed model effectively, we conducted comparative experiments with models whose performance has been demonstrated in previous recommendation system studies. The descriptions of the baseline models are as follows:

- Neural Collaborative Filtering (NCF) [35]: This model extends CF by using a neural network to learn the interactions between the latent features of users and items based on rating data, thereby predicting user preferences.
- Dual-attention-based CNN model (D-attn) [25]: This model learns latent features from the user and item review sets using dual CNNs. Local attention (L-Attn) captures important keyword expressions from context, whereas global attention (G-Attn) learns the overall meaning of reviews. These are combined to predict user preferences.
- Deep Cooperative Neural Networks (DeepCoNN) [13]: This model consists of two parallel CNNs, one of which learns the latent features of users from their review sets, whereas the other learns the latent features of items from their respective review sets. The learned features are then jointly modeled using a factorization machine to predict user preferences.
- Neural Attentional Regression model with Review-level Explanations (NARRE) [10]: This model utilizes two parallel CNNs and an attention mechanism to learn the latent features of users and items from review sets. The attention mechanism assigns different weights to each review within a set based on its importance, allowing more influential reviews to have a greater impact. This effectively reflects useful reviews for predicting user preferences.
- Aspect-aware Explainable Neural Attentional Recommender (AENAR) [27]: This model learns the latent features of users and items from review sets using CNN and combines these with latent factor vectors. It then emphasizes the important aspects through attention to predict user preferences.

D. EXPERIMENTAL SETUP

In this study, unnecessary symbols were removed, and the sequence length of the review input into the BART was limited to 512 tokens to efficiently process the review texts. To optimize the performance of the proposed model, the hyperparameters were fine-tuned across various ranges. Specifically, the batch size was tuned within [64, 128, 256, and 512], the learning rate was adjusted within [0.001, 0.002, 0.005, 0.0001, 0.0002, and 0.0005], and the dropout rate was tested across [0.1, 0.2, 0.3, 0.4, and 0.5]. Finally, the optimal settings were determined to be a batch size of 64, learning rate of 0.0001, and dropout rate of 0.1. Additionally, the number of heads used in the multihead attention mechanism of the proposed model was set to eight. The Adam optimizer was used for optimization, and early stopping was set with a patience of five epochs, meaning that training was stopped if the validation loss did not decrease for five consecutive epochs. For comparison, the parameters of the baseline models were set according to the original studies. All experiments were repeated five times to reduce experimental variance, and the results were averaged. The experiments used TensorFlow 2.17.0 on an environment with an Intel(R) Xeon(R) CPU @ 2.20GHz, 83.5GB RAM, and an A100-SXM4-40GB GPU.

V. EXPERIMENTAL RESULTS

A. PERFORMANCE COMPARISON OF THE PROPOSED MODEL (RQ1)

To evaluate the performance of the proposed SRARS model, we compared its recommendation performance with that of several baseline models using online review data collected from Amazon.com. The experimental results are listed in Table 2. The proposed model showed improvements in MAE of 3.44–23.66%, 5.91–25.00%, and 6.54–21.32%, and in RMSE of 0.29–13.55%, 0.81–11.56%, and 0.60–10.07%, for the Video Games, Industrial and Scientific, and Musical Instruments datasets, respectively.

TABLE 2. Performance comparison with baseline models.

Model	Musical Instruments		Industrial and Scientific		Video Games	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
NCF	0.835	1.102	0.828	1.116	0.955	1.203
D-attn	0.797	0.998	0.769	0.992	0.867	1.078
DeepCoNN	0.730	1.004	0.716	1.012	0.797	1.066
NARRE	0.716	0.997	0.692	0.986	0.782	1.043
AENAR	0.703	1.017	0.660	0.996	0.755	1.058
SRARS	**0.657**	**0.991**	**0.621**	**0.978**	**0.729**	**1.040**

The experimental results are summarized as follows. First, the NCF model, which relied only on rating data, showed the lowest performance. This indicates the limitations of predicting user preferences using only ratings data. Second, models utilizing reviews—D-attn, DeepCoNN, NARRE, and AENAR—outperformed rating-based models. Notably, the NARRE and AENAR models demonstrated superior performance by effectively capturing important information from user reviews. NARRE focused on influential reviews, whereas AENAR emphasized significant words or sentences within reviews, highlighting aspects that are important to users. In contrast, DeepCoNN handled all reviews equally and failed to focus on important information effectively. This limitation reduces the ability to accurately capture user and item characteristics. This suggests that emphasizing crucial information is important for improving recommendation performance. Finally, the proposed model outperformed all other baseline models for the following reasons: (1) In contrast to the NCF model, which uses only rating data, the proposed model incorporates diverse opinions found in reviews. This approach leads to an improved performance. (2) Models such as D-attn, DeepCoNN, NARRE, and AENAR use entire reviews, whereas our proposed model summarizes reviews to eliminate redundant or unnecessary content. This approach effectively captures the core information. (3) Other baseline models analyze interactions using MF-based methods and element-wise operations, which can only capture relationships within specific dimensions. In contrast, the proposed model employs an outer product to represent richer interactions across the dimensions. It also applies an attention mechanism to emphasize crucial interaction elements, thereby enabling an effective understanding of user preferences. These results demonstrate that the proposed model can enhance the recommendation performance by accurately extracting user and item characteristics from reviews and precisely learning their interactions.

B. EFFICIENCY OF SUMMARIZED REVIEWS (RQ2)

Conventional review-based recommender systems typically represent users and items using latent factor vectors, treating review texts as auxiliary information. In this study, we propose an alternative approach that models users and items based on their summarized reviews to capture preferences explicitly. To evaluate the effectiveness of this strategy, we conducted an ablation study based on three variants: (1) ''summarized review'' approach represents that utilize only summarized reviews for representation, (2)''summarized review & latent factor vector'' approach represents that combines summarized reviews with latent factor vectors, and (3) ''latent factor vector'' approach represents that utilize only traditional latent vectors. The comparative results are shown in Figure 3.

The experimental results demonstrate that the proposed model, which uses only summarized reviews, consistently outperformed other variants across all datasets. This indicates that the summarized reviews accurately reflect user preferences and provide excellent recommendation performance without latent factor vectors. Furthermore, the results highlight the critical role of review texts in recommendation tasks. In contrast, the approach that combined summarized reviews with latent factor vectors showed slightly lower performance. This suggests that adding latent factor vectors may introduce unnecessary information, negatively impacting the performance. The approach that utilized only latent factor vectors exhibited the lowest performance across all the datasets. This result highlights the limitations of capturing user preferences using only interaction data.

C. EFFICIENCY COMPARISON OF INTERACTION OPERATION METHODS (RQ3)

To model user–item interactions more effectively, this study employed an outer product operation and applied multi-head self-attention to the resulting interaction map. To verify the effectiveness of the proposed approach, we conducted an ablation study comparing multiple interaction learning methods and assessing the contribution of the attention mechanism to recommendation performance. The experimental results, shown in Figure 4, indicate that the proposed SRARS model consistently outperforms all baselines across the datasets. These findings demonstrate that combining outer product–based interaction representation with self-attention mechanism can accurately model complex user–item relationships compared to alternative approaches. As for the other methods, the outer product consistently achieved superior performance. This is because of the outer product's ability to capture fine-grained interactions by considering all pairwise relationships between embedding dimensions. In contrast, the inner product aggregates the products of corresponding dimensions, which limits representational capacity by focusing on diagonal elements. Although the element-wise product computes efficiently, it restricts interaction modeling to same-position dimensions and lacks higher-order expressiveness. Meanwhile, the concatenation preserves the original information from user and item vectors but fails to capture cross-dimensional dependencies effectively [43].

Although the outer product alone can effectively model user–item interactions, this study further enhances performance by incorporating an attention mechanism. The self-attention mechanism attenuates less significant interactions while assigning greater weight to the more salient ones, enabling the model to focus on meaningful information. This selective emphasis reduces noise during training and highlights essential relationships to improve recommendation performance.

D. IMPACT OF SEQUENCE LENGTH ON RECOMMENDATION PERFORMANCE (RQ4)

This section investigates the impact of review sequence length on the recommendation performance for learning user preferences. Experiments were conducted with sequence lengths of 256, 512, 768, and 1024 as inputs to the BART model. The experimental results are listed in Table 3.

The results indicated that the performance of the model varied based on the length of the input review sequence. Specifically, the performance consistently improved as the sequence length increased from 256 to 768, with the best performance observed at a sequence length of 768. This suggests that longer sequences enable the model to learn more information, resulting in more accurate predictions. However, when the sequence length was increased to 1024, the

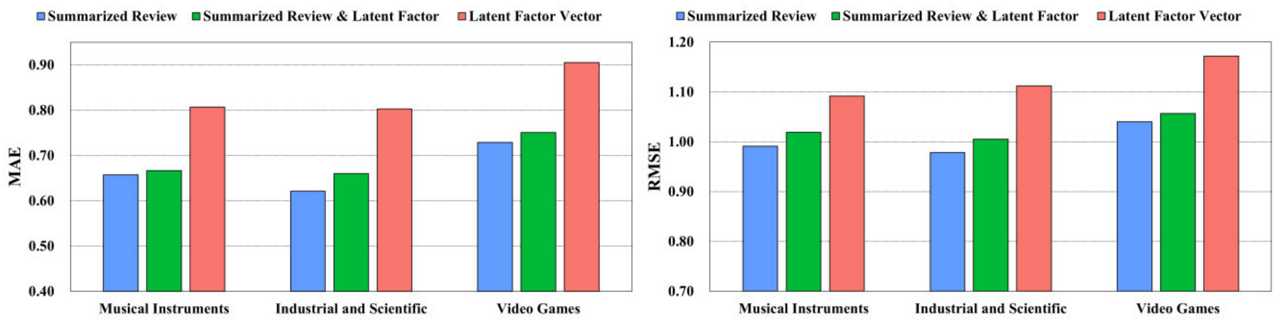

FIGURE 3. Performance comparison of latent factor vector and summarized review.

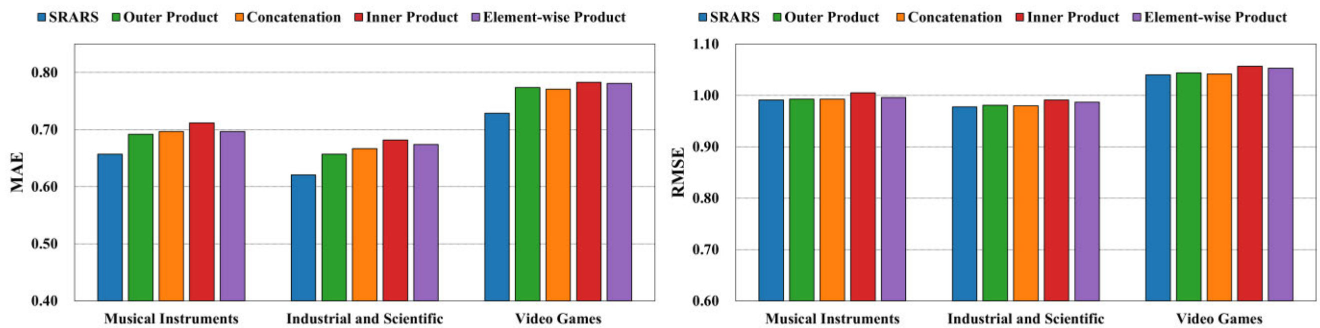

FIGURE 4. Performance comparison of interaction operation methods.

TABLE 3. Performance comparison based on sequence length.

Sequence Length	Musical Instruments		Industrial and Scientific		Video Games	
	MAE	RMSE	MAE	RMSE	MAE	RMSE
256	0.658	0.995	0.644	0.982	0.733	1.045
512	0.657	0.991	0.621	0.978	0.729	1.040
768	0.645	0.973	0.614	0.956	0.727	1.024
1024	0.664	0.988	0.624	0.975	0.730	1.038

performance declined. This indicates that excessively long reviews are more likely to include unnecessary information, which hinders model training. These findings suggest that sequences that are extremely short lack sufficient information, reducing the performance. By contrast, sequences that are exceedingly long may include irrelevant content, negatively affecting model performance.

VI. CONCLUSION

With the development of information and communication technologies, users are exposed to a vast amount of information. Such exposure leads to challenges related to information overload. Personalized recommendation systems have been researched to address these challenges. This study comprehensively utilized user reviews to understand user preferences and item characteristics from multiple perspectives. Previous studies have used reviews to predict user preferences. However, unnecessary or repetitive information in review texts can act as noise, thereby increasing the time and cost of information extraction. To address these issues, this study proposes the SRARS model. The proposed model summarizes reviews to retain their overall meaning while capturing essential information, thereby enabling a clear understanding of user preferences. Specifically, a pretrained BART model was used to summarize reviews, and this summarized information was subsequently used to extract user preferences and item characteristics. These extracted features were combined using the outer product to effectively learn user–item interactions. An attention mechanism was then applied to emphasize the significant interactions, thereby enhancing the learning process more effectively. Various experiments were conducted to validate the superiority of the proposed model.

The academic contributions of this study are as follows. First, this study empirically validated the effectiveness of capturing user preferences through summarized reviews. Whereas previous recommender-system research has largely relied on full-length review texts, we identified that redundant or irrelevant content can obscure an accurate reflection of user preferences. To overcome this limitation, we proposed a method that distils essential information by summarizing review texts and infers preferences from these distilled representations. Experiments demonstrated that the summarized reviews delivered superior performance—in terms of MAE and RMSE—compared with models using entire reviews, indicating that the summaries more effectively embody user preferences and item characteristics. Second, unlike conventional review-based recommender systems that represent users and items with latent factor vectors while treating review texts as supplementary information, we directly represented users and items solely through summarized review texts. This approach explicitly captures the inherent characteristics of users and items and alleviates data sparsity when user–item rating data are limited. Experimental results confirmed that high recommendation accuracy can be achieved using only summarized reviews, thereby validating the proposed approach as a practical alternative to latent factor-based models. Third, we compared several interaction-learning techniques to capture user–item relationships. By combining the outer product operation with a multihead attention mechanism, the proposed model captured more sophisticated and complex interactions than conventional dot-product methods. This structural modelling strategy enables a deeper understanding of user–item dynamics and ultimately enhances both the representational capacity and predictive accuracy of the recommender system. Consequently, this study provides an integrated contribution to recommender-system research by introducing a novel pipeline encompassing preference estimation, representation learning, and interaction modelling.

The practical implications of this study are as follows. First, the proposed model is not confined to Amazon.com; it is scalable to any online platform that hosts user reviews. Because traditional review-based recommenders process entire reviews—including irrelevant content—they are limited in identifying user preferences and item attributes. In contrast, the proposed model leverages only the essential information extracted through summarization, enabling clearer representations and delivering more personalized recommendations that improve user satisfaction. Second, we analysed how the length of the input review sequence affects the representation of preferences and characteristics. Although longer sequences enrich the captured features, performance declined once the sequence length exceeded a certain threshold. Experiments with the bidirectional and autoregressive transformer (BART) at various token lengths showed that excessively short sequences lack sufficient information, whereas excessively long sequences introduce information overload that hampers learning efficiency. These findings offer practical guidance on selecting an appropriate review length to capture user and item attributes effectively in real-world applications.

The limitations of this study and directions for future research are as follows. First, this study only used review texts written by users and did not consider other types of information such as user profiles or item descriptions. Previous research has indicated that using additional information such as user data or item descriptions can help alleviate data sparsity. Therefore, future studies should examine whether incorporating various types of additional information can further improve recommendation performance. Second, the proposed model was validated using the Amazon.com dataset. This may limit the generalizability of the results to other domains. Future research should evaluate the generalization performance of the proposed model using datasets from diverse domains such as movies, restaurants, and hotels. Third, although this study employed the BART model to generate summaries that preserve the original intent of review texts, other summarization models have different strengths and capabilities. BART can generate coherent and creative summaries while effectively retaining the meaning of complex texts. In contrast, BERT demonstrates strong performance in extractive summarization by capturing bidirectional contextual information and preserving key content. T5, based on a text-to-text framework, is well-suited for generating concise and easily readable summaries [38], [39]. Therefore, future research should investigate and compare a broader range of pretrained language models, including BERT and T5, to further enhance summary quality and recommendation performance. Finally, this study set rating prediction as its primary objective and used MAE and RMSE as the evaluation metrics. In practical recommender-system settings, the quality of the top-K items recommended to a user is also important. Accordingly, future research should complement the current evaluation by incorporating ranking-based metrics such as Recall and NDCG. Reporting both numerical prediction accuracy (MAE/RMSE) and top-K recommendation quality (Recall/NDCG) will provide a more complete assessment of model performance.

REFERENCES

[1] D. Roy and M. Dutta, "A systematic review and research perspective on recommender systems," *J. Big Data*, vol. 9, no. 1, p. 59, Dec. 2022.

[2] F. Ricci, L. Rokach, and B. Shapira, *Introduction to Recommender Systems Handbook, in Recommender Systems Handbook*. Cham, Switzerland: Springer, 2010, pp. 1–35.

[3] K. Wei, J. Huang, and S. Fu, "A survey of e-commerce recommender systems," in *Proc. Int. Conf. Service Syst. Service Manage.*, Jun. 2007, pp. 1–5.

[4] 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.

[5] J. B. Schafer, J. A. Konstan, and J. Riedl, "E-commerce recommendation applications," *Data Mining Knowl. Discovery*, vol. 5, pp. 115–153, Jan. 2001.

[6] C. A. Gomez-Uribe and N. Hunt, "The Netflix recommender system: Algorithms, business value, and innovation," *ACM Trans. Manag. Inf. Syst.*, vol. 6, no. 4, pp. 1–19, 2015.

[7] 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, 2019.

[8] J. B. Schafer, D. Frankowski, J. Herlocker, and S. Sen, "Collaborative filtering recommender systems," in *The Adaptive Web: Methods and Strategies of Web Personalization*. Cham, Switzerland: Springer, 2007, pp. 291–324.

[9] R. Garapati and M. Chakraborty, "Recommender systems in the digital age: A comprehensive review of methods, challenges, and applications," *Knowl. Inf. Syst.*, vol. 67, no. 8, pp. 1–45, Aug. 2025.

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

[11] M. Hernández-Rubio, I. Cantador, and A. Bellogín, "A comparative analysis of recommender systems based on item aspect opinions extracted from user reviews," *User Model. User-Adapted Interact.*, vol. 29, no. 2, pp. 381–441, Apr. 2019.

[12] L. Chen, G. Chen, and F. Wang, "Recommender systems based on user reviews: The state of the art," *User Model. User-Adapted Interact.*, vol. 25, no. 2, pp. 99–154, Jun. 2015.

[13] 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.

[14] L. Wu, C. Quan, C. Li, Q. Wang, B. Zheng, and X. Luo, "A context-aware user-item representation learning for item recommendation," *ACM Trans. Inf. Syst.*, vol. 37, no. 2, pp. 1–29, Apr. 2019.

[15] G. Ganu, Y. Kakodkar, and A. Marian, "Improving the quality of predictions using textual information in online user reviews," *Inf. Syst.*, vol. 38, no. 1, pp. 1–15, Mar. 2013.

[16] M. Peng, B. Gao, J. Zhu, J. Huang, M. Yuan, and F. Li, "High quality information extraction and query-oriented summarization for automatic query-reply in social network," *Expert Syst. Appl.*, vol. 44, pp. 92–101, Feb. 2016.

[17] A. Aries, D. E. Zegour, and W. K. Hidouci, "Automatic text summarization: What has been done and what has to be done," 2019, *arXiv:1904.00688*.

[18] W. S. El-Kassas, C. R. Salama, A. A. Rafea, and H. K. Mohamed, "Automatic text summarization: A comprehensive survey," *Expert Syst. Appl.*, vol. 165, Mar. 2021, Art. no. 113679.

[19] C.-F. Tsai, K. Chen, Y.-H. Hu, and W.-K. Chen, "Improving text summarization of online hotel reviews with review helpfulness and sentiment," *Tourism Manage.*, vol. 80, Oct. 2020, Art. no. 104122.

[20] A. K. Yadav, Ranvijay, R. S. Yadav, and A. K. Maurya, "State-of-the-art approach to extractive text summarization: A comprehensive review," *Multimedia Tools Appl.*, vol. 82, no. 19, pp. 29135–29197, Aug. 2023.

[21] H. Li, A. Einolghozati, S. Iyer, B. Paranjape, Y. Mehdad, S. Gupta, and M. Ghazvininejad, "EASE: Extractive-abstractive summarization with explanations," 2021, *arXiv:2105.06982*.

[22] A. P. Widyassari, S. Rustad, G. F. Shidik, E. Noersasongko, A. Syukur, A. Affandy, and D. R. I. M. Setiadi, "Review of automatic text summarization techniques & methods," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 34, no. 4, pp. 1029–1046, 2022.

[23] G. Sharma and D. Sharma, "Automatic text summarization methods: A comprehensive review," *Social Netw. Comput. Sci.*, vol. 4, no. 1, p. 33, Oct. 2022.

[24] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," 2019, *arXiv:1910.13461*.

[25] S. Seo, J. Huang, H. Yang, and Y. Liu, "Interpretable convolutional neural networks with dual local and global attention for review rating prediction," in *Proc. 11th ACM Conf. Recommender Syst.*, Aug. 2017, pp. 297–305.

[26] W. Li and B. Xu, "Aspect-based fashion recommendation with attention mechanism," *IEEE Access*, vol. 8, pp. 141814–141823, 2020.

[27] T. Zhang, C. Sun, Z. Cheng, and X. Dong, "AENAR: An aspect-aware explainable neural attentional recommender model for rating predication," *Expert Syst. Appl.*, vol. 198, Jul. 2022, Art. no. 116717.

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

[29] Y. Bai, Y. Li, and L. Wang, "A joint summarization and pre-trained model for review-based recommendation," *Information*, vol. 12, no. 6, p. 223, May 2021.

[30] R. K. Roul and K. Arora, "A nifty review to text summarization-based recommendation system for electronic products," *Soft Comput.*, vol. 23, no. 24, pp. 13183–13204, Dec. 2019.

[31] H. Xu, H. Liu, W. Zhang, P. Jiao, and W. Wang, "Rating-boosted abstractive review summarization with neural personalized generation," *Knowl.-Based Syst.*, vol. 218, Apr. 2021, Art. no. 106858.

[32] L. Cui, Y. Wu, J. Liu, S. Yang, and Y. Zhang, "Template-based named entity recognition using BART," 2021, *arXiv:2106.01760*.

[33] Y. Koren, R. Bell, and C. Volinsky, "Matrix factorization techniques for recommender systems," *Computer*, vol. 42, no. 8, pp. 30–37, Aug. 2009.

[34] A. Mnih and R. R. Salakhutdinov, "Probabilistic matrix factorization," in *Proc. Adv. Neural Inf. Process. Syst.*, 2007, pp. 1257–1264.

[35] 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.

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

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

[38] N. A. Maghfiroh, F. Abdurrachman Bachtiar, and L. Muflikhah, "Comparative analysis of summarization methods for skin care product reviews: A study on BERT, BART, and T5 models," in *Proc. Int. Conf. Adv. Mechatronics, Intell. Manuf. Ind. Autom. (ICAMIMIA)*, Nov. 2023, pp. 593–598.

[39] S. Das, D. Basak, and A. Bhattacharjee, "Text simplification using T5 model and BART model," in *Proc. 8th Int. Conf. Electron., Mater. Eng. Nano-Technol. (IEMENTech)*, Jan. 2025, pp. 1–6.

[40] S. Hong, X. Li, S. Yang, and J. Kim, "Review-based recommender system using outer product on CNN," *IEEE Access*, vol. 12, pp. 65650–65659, 2024.

[41] M. Ferrari Dacrema, F. Parroni, P. Cremonesi, and D. Jannach, "Critically examining the claimed value of convolutions over user-item embedding maps for recommender systems," in *Proc. 29th ACM Int. Conf. Inf. Knowl. Manage.*, Oct. 2020, pp. 355–363.

[42] R.-P. Shen, H.-R. Zhang, H. Yu, and F. Min, "Sentiment based matrix factorization with reliability for recommendation," *Expert Syst. Appl.*, vol. 135, pp. 249–258, Nov. 2019.

[43] A. Beutel, P. Covington, S. Jain, C. Xu, J. Li, V. Gatto, and E. H. Chi, "Latent cross: Making use of context in recurrent recommender systems," in *Proc. 11th ACM Int. Conf. Web Search Data Mining*, Feb. 2018, pp. 46–54.

SUJI KIM received the B.S. degree in convergence engineering from Sahmyook University. She is currently pursuing the M.S. degree in big data analytics with Kyung Hee University. Her research interests include big data analytics, natural language processing, deep learning, and recommender systems.

LI JIN received the B.S. degree in network engineering from Dalian Maritime University, China, in 2023. She is currently pursuing the M.S. degree in big data analytics with Kyung Hee University, South Korea. Her research interests include big data analytics, natural language processing, and recommender systems.

QINGLONG LI received the B.S. degree in business administration and the M.S. and Ph.D. degrees in big data analytics from Kyung Hee University, Seoul, South Korea. He is currently an Assistant Professor with the Division of Computer Engineering, Hansung University, Seoul. His work has been published in journals, such as *Applied Intelligence*, *Information Systems*, *International Journal of Hospitality Management*, and *Expert Systems*. His research interests include recommender systems, deep learning, online reviews, and big data applications.

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, and natural language processing.

CHULKYOON YOON received the M.S. degree in executive business administration from Kyung Hee University, where he is currently pursuing the Ph.D. degree in business administration. His research interests include sales and marketing, big data analytics, and deep 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 several articles, which have appeared in IEEE Transactions on Systems, Man, and Cybernetics, IEEE Transactions on Services Computing, *Information and 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 Negotiations*. His current research interests include business analytics, recommender systems, big data analysis, and deep learning.