applied sciences

MDPI

Article

A Hybrid CNN-Based Review Helpfulness Filtering Model for Improving E-Commerce Recommendation Service

Qinglong Li [1], Xinzhe Li [1], Byunghyun Lee [1] and Jaekyeong Kim [1,2,*]

[1] Department of Big Data Analytics, Kyung Hee University, 26, Kyungheedae-ro, Dongdaemun-gu, Seoul 02447, Korea; leecy@khu.ac.kr (Q.L.); lixz@khu.ac.kr (X.L.); leebh1129@khu.ac.kr (B.L.)
[2] School of Management, Kyung Hee University, 26, Kyungheedae-ro, Dongdaemun-gu, Seoul 02447, Korea
* Correspondence: jaek@khu.ac.kr; Tel.: +82-2-961-9355

Abstract: As the e-commerce market grows worldwide, personalized recommendation services have become essential to users' personalized items or services. They can decrease the cost of user information exploration and have a positive impact on corporate sales growth. Recently, many studies have been actively conducted using reviews written by users to address traditional recommender system research problems. However, reviews can include content that is not conducive to purchasing decisions, such as advertising, false reviews, or fake reviews. Using such reviews to provide recommendation services can lower the recommendation performance as well as a trust in the company. This study proposes a novel review of the helpfulness-based recommendation methodology (RHRM) framework to support users' purchasing decisions in personalized recommendation services. The core of our framework is a review semantics extractor and a user/item recommendation generator. The review semantics extractor learns reviews representations in a convolutional neural network and bidirectional long short-term memory hybrid neural network for review helpfulness classification. The user/item recommendation generator models the user's preference on items based on their past interactions. Here, past interactions indicate only records in which the user-written reviews of items are helpful. Since many reviews do not have helpfulness scores, we first propose a helpfulness classification model to reflect the review helpfulness that significantly impacts users' purchasing decisions in personalized recommendation services. The helpfulness classification model is trained about limited reviews utilizing helpfulness scores. Several experiments with the Amazon dataset show that if review helpfulness information is used in the recommender system, performance such as the accuracy of personalized recommendation service can be further improved, thereby enhancing user satisfaction and further increasing trust in the company.

Keywords: collaborative filtering; convolutional neural networks; review helpfulness; personalized recommendation services

check for updates

Citation: Li, Q.; Li, X.; Lee, B.; Kim, J. A Hybrid CNN-Based Review Helpfulness Filtering Model for Improving E-Commerce Recommendation Service. *Appl. Sci.* **2021**, *11*, 8613. https://doi.org/10.3390/app11188613

Academic Editor: Ha Young Kim

Received: 26 August 2021
Accepted: 14 September 2021
Published: 16 September 2021

Publisher's Note: MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Copyright: © 2021 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

As the e-commerce market overgrows worldwide with the development of information technology and the popularization of mobile devices, various types of products continue to be released [1,2]. However, users face a time-consuming information overload problem in the purchasing decision-making process. Significantly, the issue of information overload multiplies because the user experiences the product indirectly online. Therefore, personalized recommendation services have been becoming important in providing personalized items or services to users. Global e-commerce companies such as Netflix, Amazon, and Google have introduced personalized recommendation services to help users make purchasing decisions [3–5]. They can decrease the cost of user information exploration and have a positive impact on corporate sales growth. For example, 75% of videos viewed by users on Netflix are provided through personalized recommendation services. Amazon generates 35% of its total revenue from items recommended to users through personal recommendation services [6].

Collaborative Filtering (CF) is the state-of-the-art recommendation model, which identifies users' and items' interactions and provides personalized recommendation services from quantitative information such as clicking, rating, and viewing [7–11]. However, such a methodology only models the action pattern without capturing qualitative preferences such as a motivation and a purchase reason for the item [12,13]. Therefore, such methodologies can raise the issue where recommendation performance decreases [1,14]. Recently, many studies have been conducted using various additional information to address the limitation of existing studies. Most e-commerce websites provide review modules where the users write reviews of their purchased items. According to Moore [15], 88% of users make purchasing decisions by referring to reviews when purchasing products. A review text can be helpful because it includes specific and reliable information such as the reason for purchasing and evaluating the item [14]. However, the existing studies of personalized recommendation services using reviews mainly focused on extracting sentiment features or exploring several attributes and utilized them by combining with the CF approach [16]. However, reviews include unhelpful content for inconducive purchasing decisions such as advertising, unmeaningful content, or fake reviews [17]. It is therefore indisputable that providing recommendation services without any considerations of the quality of the review may decrease recommendation performance [18].

In order to address the limitation of the existing study problems, this study aims to review helpfulness information in the personalized recommendation service that can affect users' purchase decisions. Recently, the number of reviews of items has been increasing as more users purchase items on e-commerce websites. In Table 1, users can identify the product's characteristics from reviews and utilize much of the information in the purchase decision-making process. However, users cannot refer to all reviews in the purchase decision-making process. Therefore, users have difficulties in exploring helpful reviews in the product purchase process. To address this issue, Amazon provides a review helpfulness voting module to confirm whether reviews are helpful in the purchase decisions process since 2007 [19]. The ranking of reviews is sorted through the number of review helpfulness votes, and the most voted reviews are marked at the top of the list. Because the review helpfulness information has an significant role in the user's purchase decision-making process, and it plays an essential role in providing personalized recommendation services [20].

Table 1. Number of reviews received by Amazon Best Sellers items.

Item Category	Item Name	Number of Reviews
Automotive	THISWORX Car Vacuum Cleaner	170,864
Sports and Outdoors	Iron Flask Sports Water Bottle	68,956
Pet Supplies	Amazon Basics Dog and Puppy Pads	125,848
Electronics	Echo Dot (3rd Gen)	895,176
Home and Kitchen	Mellanni Queen Sheet Set	241,804

This study proposes a novel reviews helpfulness-based recommendation methodology (RHRM) framework that can support users' purchasing decisions in personalized recommendation services. Our framework consists of three phases: a review semantics extractor, a user profile producer, and a user/item recommendation generator. First, in the review semantics extractor phase, we generate review representations hierarchically for review helpfulness classification. We first extract the review's semantic representation using Convolutional Neural Network (CNN), then obtain two-way representations using the Bi-directional Long Short-Term Memory (BiLSTM) attention network and combine such representations to generate a final semantic representation. Since many reviews do not have helpfulness scores, we first propose a helpfulness classification model to reflect the review helpfulness that significantly impacts users' purchasing decisions in personalized recommendation services. This CNN–BiLSTM hybrid model utilizes generated semantic representation to classify the helpfulness of the reviews. After review helpfulness

information is classified, we send it to the user profile producer phase. Second, the user profile producer phase also utilizes helpfulness information classification results to update user profiles based on helpful reviews that the user has written about the item. Here, the updated user profile contains only user/item interactions that correspond to written helpful reviews by the user. Finally, the user/item recommendation generator utilized the most popular CF techniques to model users' preferences on items based on their interactions profile produced in phase 2. We applied User-Based CF (UBCF), Singular Value Decomposition (SVD), and Neural Collaborative Filtering (NCF), the most popular models in CF techniques. We have conducted extensive experiments with the Amazon dataset. The results demonstrate that our framework can effectively improve the performance recommendations when reflecting the review helpfulness information. The contributions that this paper have made are summarized as follows:

- This study first proposes the RHRM framework that has filtered the review helpfulness and reflected upon personalized recommendation services. It can enhance the recommendation performance because it reflects the purchasing behavior of the users who consider reviews when purchasing items.
- This study has built a review helpfulness classification model using the combined CNN and BiLSTM that demonstrates excellent performance in the Natural Language Processing (NLP) study. We confirm the advantages of the combined CNN–BiLSTM hybrid model in semantic representation extraction through various experiments.
- This study has conducted several experiments with the Amazon dataset. The results indicate that reflecting review helpfulness information can enhance the prediction performance of personalized recommendation services, increase user satisfaction, and raise confidence in the company.

The rest of the composition of this study is as follows. Section 2 describes the theoretical background for personalized recommendation services, review-based personalized recommendation services, and review text classification with deep learning approaches. Section 3 describes the proposed recommendation framework. Section 4 describes the experimental dataset, evaluation metric, and results. Finally, Section 5 discusses the discussion, limitations, and the future study.

2. Related Work

2.1. Collaborative Filtering

A personalized recommendation service uses ratings, purchase history, and browsing history to provide products or services to users [5]. Furthermore, such a personalized recommendation service provides convenience for users who have difficulty making purchasing decisions on several types of items and services. Global companies such as Netflix, Amazon, and Google generate revenues by introducing personalized recommendation services in e-commerce to support users' decision making [6,21]. Therefore, personalized recommendation services are used in various industries, and related studies are conducted continuously [1,22]. Currently, CF algorithms are widely used in academia and industry with excellent recommendation performance [10,23].

CF is a recommended approach based on the similarity between users or items, assuming that users with preferences for the same item have similar preferences for other items [24,25]. CF algorithms are divided into memory-based CF and model-based CF. Memory-based CF is divided into two categories: UBCF and Item-Based CF (IBCF) [26]. UBCF is a method of recommendation items purchased by users with similar preferences to the recommended users. The recommendations are provided through three stages: First, measure the similarity between users to select neighbor users similar to the recommended users. Next, calculate the item preference prediction rating for the recommended user. Finally, the product with the highest preference prediction value is recommended to the user [27]. The IBCF recommendation method is that users prefer items similar to historical purchases items. In other words, the target user recommends the most similar items based on the historical purchase's items. Model-based CF uses the previous datasets to train a

model with machine learning or data-mining techniques to improve the performance of the CF method [28]. These techniques can quickly recommend a series of items for the fact that they use a precomputed model, and they have proved to produce recommendation results that are similar to neighborhood-based recommender techniques [29]. In addition, the techniques need to be used in the categorization model if the user preference is categorical data. Suppose user preference is continuous data, techniques such as SVD, NCF, and Regression, should be used [30]. Despite the success of the CF-based recommender system, some problems have been revealed, such as the following: This method essentially recommends items based on users' past purchasing history and preferences. However, recommender systems experience a cold-start issue in new users, as there is insufficient data available to measure similarity; therefore, user preferences cannot be predicted [25]. Furthermore, a first-start issue exists in which users' preferred items are not recommended because they have not yet been purchased [25].

The existing studies on the CF have predicted users' preferences, which used quantitative data such as clicking, rating, and viewing. However, such a traditional approach without understanding behavior motivation can reduce the recommendation performance. To address the limitations of CF approach, most studies use additional information. Typically, review text is among them. In this study, we propose a framework considering the review text, which represents the unstructured data to improve the limitations of existing CF approaches. We hope to address the limitations of the CF approach, which only considers quantitative information, to provide excellent recommendation performance.

2.2. Review-Based Recommender System

Reviews are qualitative data as they refer to users' written review about the item information or experience. Such reviews are an important feature in which users can represent detailed expressing opinions about the items [31]. Therefore, most studies develop various recommender systems using reviews to overcome existing recommender systems' limitations that only use quantitative data. Leung et al. [32] applied sentiment analysis to movie reviews and developed a model to estimate the review's sentiment. Then, the calculated sentiment index from models and is reflected in the CF. It is the first study that applies user reviews to recommender systems. However, it only considered qualitative information and the review's sentiment. Therefore, it provides for higher recommendation performance when considering both qualitative and quantitative data simultaneously. García-Cumbreras et al. [33] performed sentiment analysis to user-written reviews and classified users as intuitionists and pessimists. The performance of CF was higher when users are classified as intuitionists and pessimists than traditional CF. It is significant in that the user-written reviews were classified. However, review contents were not reflected in recommender systems, and there is a chance to reduce the loss of information. Zhang et al. [34] proposed an urCF (User Review enhanced Collaborative Filtering) recommendation methodology that reflected the review. It used the reviews about 32 movies in the movie review ontology of Zhou and Chaovalit [35]. The reviews' features were derived using FF-IRF (Feature Frequency-Inverse Review Frequency), similar to TF-IDF (Term Frequency-Inverse Document Frequency). The user's sentiment polarity is reflected in each review's features, then the similarity between users is calculated, and CF algorithms are proposed based on them. The results showed that the proposed methodology applied to Yahoo Movies data improved the prediction accuracy from 6.18% to 8.24% over traditional CF methods. The prediction performance results were excellent, but it disregarded the content of reviews. Jeon and Ahn [36] considered user-written reviews to improve the performance of CF. They verified the effectiveness of the proposed methodology applied to smartphone app review data and quantified reviews through text mining. Their results show that reflecting review between users' similarity in CF was better than traditional CF on performance. Hyun et al. [37] proposed a recommendation algorithm that combined user-written reviews and ratings to reflect on the CF. They established a sentiment dictionary using movie review data. The sentiment index of reviews was derived

from the sentiment dictionary, and new ratings generated by combining sentiment index with ratings are reflected in the CF. They proposed a new methodology that combines reviews and ratings. However, they only reflected the positive and negative sentiment of the reviews and did not consider the content or helpfulness of the review.

The existing recommender system studies using review data follow the same paradigm, in which the historical reviews are aggregated into a long document. Then, they focused on extracting the sentiment features or performed topic modeling analysis on the reviews text. However, the review text may include unhelpful content for users to make decisions, such as advertisements and fake reviews. Thus, having disregarded the content or helpfulness of the review, the recommendation performance decreases [38]. This study proposes an RHRM framework that provides personalized recommendation services by producing user profiles through filtering helpful information reviews. We try to filter high-quality reviews that help users make their decisions.

2.3. Review Text Classification with Deep Learning Approaches

Review text is one of the easiest and most effective ways for users to express a sentiment, such as the purpose and reason of purchase on an e-commerce website. Therefore, it is significant to explore the sentiment of these review texts [39]. Many researchers apply deep learning techniques that demonstrate the excellent performance in other domains to sentimental textual analysis [39]. Most studies of text classification now focus on the construction and optimization of neural networks [40]. Stojanovski et al. [41] proposed a CNN-based system for sentiment analysis, which is 8% higher than traditional sentiment analysis and sentiment identification of Twitter messages. Song et al. [42] proposed a positional convolutional neural network (P-CNN) that can enhance feature extraction by capturing positional features at three different language levels: word level, phrase level, and sentence level. Abdi et al. [43] proposed a deep learning-based method (RNSA) that applies Recurrent Neural Network (RNN), Long Short-Term Memory (LSTM) for sentiment analysis at the sentence level. This approach enhanced classification performance by more than 5% in review text sentiment classification through applying multi feature fusion methods. Rao et al. [44] proposed a new neural network model (SR-LSTM) with two hidden layers to capture long-term context texts and utilized semantic relationships between sentences in document-level sentiment classification. Experiment results show SR-LSTM outperforms the state-of-the-art models on three document-level review datasets. Both single neural networks, such as CNN and RNN, have been shown to have specific weaknesses. Therefore, building a hybrid network using the advantages of CNN and RNN has become a critical study direction. Hassan and Mahmood [45] proposed a neural language model combining CNN and Bidirectional Recurrent Neural Network (BRNN) for text classification. The bidirectional layers are a substitute for pooling layers in CNN to reduce information loss during the pooling operation and to capture the long-term dependencies of text sequences. Experiments with two sentiment analysis datasets show that the proposed model has better competitiveness than the state-of-art best. Hassan and Mahmood [46] proposed a new framework that exploits LSTM and CNN models to reduce detailed, local information loss and capture long-term dependencies. Experiments with this method demonstrated excellent performance with 93.3% accuracy and 48.8% accuracy on the Stanford Large Movie Review and Stanford Large Movie Review datasets. Liu and Guo [47] proposed an architecture that combines bidirectional long short-term memory with convolution layer (AC-BiLSTM). The CNN extracted semantic representations from word embedding vectors, and BiLSTM captured semantic context features. The model applied the attention mechanism to provide different attention to contextual feature information. Results show that the AC-BiLSTM model indicates excellent performance compared to the state-of-art text classification models. Batbaatar et al. [48] proposed a novel Semantic-Emotion Neural Network (SENN) architecture that utilizes BiLSTM and CNN combination model. The BiLSTM was used to capture contextual information and semantic relationships from word-level text vectors and use CNN to extract emotional features and

relationships between words from the text. Zheng and Zheng [49] proposed a hybrid Bidirectional Recurrent Convolutional Neural Network Attention-based (BRCAN) model to address the limitations of the traditional text classification model. Bi-LSTM captures the long-term contextual information when learning word representations. CNN is used to capture the critical feature of words in text classification through contextual information. The attention mechanism gives higher weight to critical keywords when classifying text. The result shows that the proposed model achieves an F1 value of 97.86% in the Sogou text classification dataset.

However, analyzing the sentiment of the review text is similar to the sequential model approach. The CNN approach is challenging for capturing long-term context information and requires multiple CNN layers modeled to capture long-term dependencies. Because the RNN approach is highly complex and challenging to extract dependencies between long-distance contexts accurately, the CNN approach is suitable for capturing long-term context information. However, the RNN approach generally outperforms CNN-based methods in the short text corpus. The combination hybrid networks of CNN and RNN can address the limitations of CNN and RNN. However, this combined network approach ignores the contribution of high-level features at different scales from the original context feature. In addition, the model must use a different convolutional kernel to extract different high-level features, which can increase complexity. Therefore, this study applies a scalable multichannel CNN-BiLSTM hybrid model with an attention mechanism for classifying the helpfulness of reviews. The applied hybrid models can obtain high-level semantic representations and original context information through a multichannel filter kernel. Therefore, it can significantly contribute to the effective implementation of the RHRM framework proposed in this study. Section 3.1 introduces specific information.

3. RHRM Framework

In this section, we specifically describe the RHRM framework shown in Figure 1. Our framework consists of three phases: a review semantics extractor, a user profile producer, and a user/item recommendation generator. The first phase classifies the helpfulness of the review. It uses a CNN-BiLSTM hybrid model to generate review semantic representation and conduces review helpfulness classification [50,51]. The second phase produces the user profile that contains only user/item interactions that correspond to written helpful reviews by the user. The final phase utilized the most popular CF techniques to model users' preferences based on their interactions profile. We introduce the details of each phase as follows.

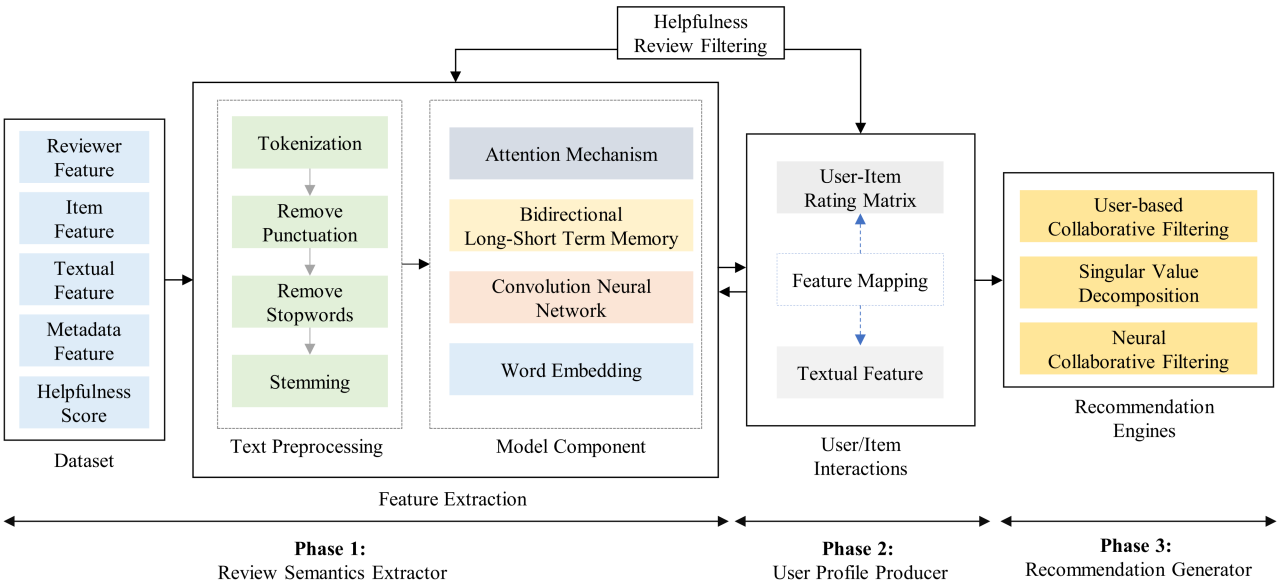


Figure 1. Proposed RHRM framework.

3.1. Phase 1: Review Semantics Extractor

The first phase constructs a CNN-BiLSTM hybrid model to classify review helpfulness information. The architecture overview of the CNN–BiLSTM hybrid model is shown in Figure 2. This study builds a CNN–BiLSTM hybrid model with excellent classification performance in NLP studies to classify review helpfulness [52,53]. CNN can reduce the input features for prediction, and the correlation between each word and a final classification is not the same for all input words [54,55]. The BiLSTM is utilized for encoding long-distance word dependencies effectively [52,53]. Due to each of these advantages, various types of hybrid CNN-BiLSTM models have been proposed [40,50,52,53,56]. The CNN-BiLSTM hybrid model applied in this study was motivated by Rai et al. [51] and Liu et al. [50]. Existing models mainly used a combination of a single CNN network and a single BiLSTM network. The model was either used as a regression model to predict numeric values or applied to multiple classification problems. Following the common single model combination strategy, we applied multiple filter kernels and added a new attention mechanism layer to extract the review text's semantic representation elaborately [47,57]. After generating a review-level semantic representation, the model classifies the helpfulness information for each review.

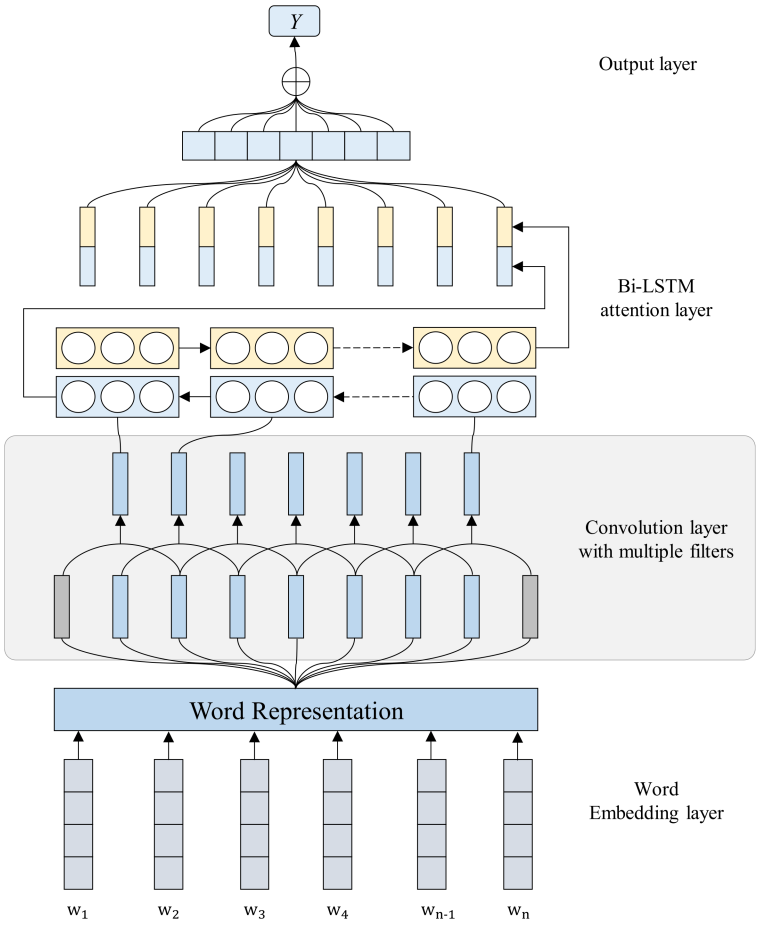


Figure 2. The architecture of CNN–BiLSTM hybrid model with the attention mechanism.

In this study, we gave $\mathcal{R} = \{r_1, r_2, \cdots, r_n\}$ as a dataset for constructing a CNN-BiLSTM hybrid model with the attention mechanism. Each review contains five attributions [P, U, C, M, H], where P indicates the item features, U indicates reviewer features, C indicates textual features, and M indicates metadata features (e.g., ratings and timestamp). H indicates the helpfulness score that is measured as the ratio of helpful votes to the total votes, where $H \in [0,1]$. Let F as a $n \times m$ review feature matrix, where n is the number of reviews in the dataset and m is the total number of

features. Z is an embedding vector of the predicted value for all reviews, where Z_i represents whether a review is helpful or not. Finally, we define a helpfulness threshold value Θ_1 and Θ_2. Therefore, Z_i is calculated as follows:

$$Z_i = \begin{cases} 1, & \text{if } H_i > \Theta_1 \\ 0, & \text{if } H_i < \Theta_2 \end{cases} \quad (1)$$

This study constructs a CNN–BiLSTM hybrid model that minimizes the prediction error of Z given F. The trained model is utilized to predict the helpfulness score of new review with unknown or unidentified helpful scores.

The CNN–BiLSTM hybrid model consists of three layers. The first layer is word embedding. Let $R_{u,i} = \{w_1, w_2, \cdots, w_n\}$ be a review text, which indicates that the user u has written the review to item i, where n is the length in the review. Many existing text-mining models were mainly applying the one-hot encoding method to convert each word into a vector. However, such a method has a data sparsity issue where the matrix dimensions are too large, and most of the vector values are filled with zero. In this study, each word included in the review was converted into a vector type through the word embedding layer [57]. Thus, this study has applied word embedding $f: w_n \rightarrow R^D$ for each word in the review, and then each word is represented as a dense vector. Then, the review text is represented by a matrix $E \in R^{n \times d}$, where d is the dimension of the word embedding vector.

The second layer is a multichannel convolutional layer. It extracts the word-level semantic representation from the review text through different sizes filters. Then, it adopted a filter K_j with a sliding window to performing a convolution operation. The convolution operation process can be defined as shown in Equation (2).

$$c_j = \phi(E * K_j + b_j), \quad (2)$$

where $*$ indicates convolution operator, $K_j \in R^{k \times m}$ indicates the parameter of the filter kernel, and $k \times m$ denotes kernel size. b_j is represented bias, and θ is the activation function *ReLU*, which is defined as Equation (3).

$$relu(x) = max(0, x) \quad (3)$$

We add the max-pooling layer to the output of the convolution operation to retain the main semantics and suppress noise. The max-pooling operation is defined as Equation (4).

$$O_j = \max([c_1, c_2, \cdots, c_{(l-t+1)}]) \quad (4)$$

This study applied multiple filters of different sizes to extract the various semantic feature included in the review. Finally, the output of the convolutional layer is as Equation (5).

$$O = [o_1, o_2, \cdots, o_n] \quad (5)$$

The third layer is an attention network. Each vector in the convolution layer output denotes the time step of the BiLSTM model. BiLSTM consists of two components: forward LSTM and backward LSTM. The forward LSTM captures the review semantic in the path from left to right, and the backward LSTM captures the sequence feature from right to left. This study defines the outputs of the forward and backward LSTMs as $\overrightarrow{S}_t$ and $\overleftarrow{S}_t$, respectively. We applied Bi-LSTM for processing all terms in the path sequence to obtain two separate hidden state sequences. Let the defined input sequence $\{o_1, o_2, \cdots, o_n\}$,

the forward LSTM generate hidden states $\left\{\overrightarrow{S}_1, \overrightarrow{S}_2, \cdots, \overrightarrow{S}_t\right\}$, and the backward LSTM generate hidden states $\left\{\overleftarrow{S}_1, \overleftarrow{S}_2, \cdots, \overleftarrow{S}_t\right\}$.

$$\begin{aligned} \overrightarrow{S}_t &= \text{LSTM}(\overrightarrow{S}_{t-1}, O_t) \\ \overleftarrow{S}_t &= \text{LSTM}(\overleftarrow{S}_{t-1}, O_t) \\ m &= [\overrightarrow{S}_l; \overleftarrow{S}_1] \end{aligned} \tag{6}$$

The BiLSTM connects the last hidden state of the forward LSTM with the first hidden state of the backward LSTM to generate the final representation. The embedding vector m consists of both forward and backward information of the path to efficiently capture the orderings. Finally, to highlight the importance of different words to the classification of review helpfulness, we added the attention mechanism layer in the CNN–BiLSTM hybrid model to further extract review features and highlight the review-helpfulness-related information. This study belongs to the feed-forward attention mechanism, defined as Equation (7).

$$\begin{aligned} h_t &= \sigma(m_i) \\ a_t &= \frac{\exp(h_t)}{\sum_{i=1}^{n} \exp(h_t)} \\ Q &= \sum_{i=1}^{m} a_t \cdot m_i \end{aligned} \tag{7}$$

where m_t indicates the eigenvector output of the BiLSTM layer and σ is the attention learning activation function *tanh*. h_t is the weight of the calculated generated attention. a_t is the matching score indicating how well the model participates in the path when responding to a query relation. The weighted sum operation uses the *SoftMax* function for normalization to generate an attention probability. Q indicates a fusion feature of the representation multiplied by the probability of attention and the hidden state semantics encoding m_t. Then, assign attention weight using the sum of weights.

The objective of this model is to compute the probability of the helpfulness score based on the semantic feature extracted from the review and classify the results, which can be defined as Equation (8).

$$Y = \theta(W_s \cdot Q + b_s), \tag{8}$$

where θ indicates the *Sigmoid* activation function, W_s indicates the weight matrix, and b_s indicates the bias. Finally, the smectic input feature of review is classified as 0 or 1 and returned as output. A value of 0 output indicates that the review is unhelpful, and a value of 1 indicates a helpful review.

3.2. Phase 2: User Profile Producer

The second phase also utilizes helpfulness information classification results to update user profiles based on the user's helpful reviews about the item. We applied the CNN-BiLSTM hybrid model that we constructed in the first phase to classify review usefulness information. Here, the updated user profile contains only user/item interactions that correspond to written helpful reviews by the user. Given that $\Re = \{r_1, r_2, \cdots, r_m\}$ is a set of new reviews, each review can contain five attributions $[\mathcal{P}, \mathcal{U}, \mathcal{C}, \mathcal{M}]$, where $\mathcal{P}$ and $\mathcal{U}$ indicate item and reviewer features, respectively. In addition, $\mathcal{C}$ is a textual feature of the new reviews, and $\mathcal{M}$ is metadata features (e.g., ratings and timestamp). Let $\mathbb{R}_{ui}$ be a $N \times M$ review feature matrix, where N is the number of reviews in the dataset and M is the total number of features. Y is the embedding vector value in which

the CNN–BiLSTM hybrid model predicts all new reviews, where Y_{ui} represents review r_{ui} is helpful or not helpful.

$$\mathbb{R}_{ui} = \begin{cases} 1, & \text{if } r_{ui} \text{ (user } u, \text{ item } i) \text{ indicate helpful;} \\ 0, & \text{otherwise.} \end{cases} \tag{9}$$

where 1 for $\mathbb{R}_{ui}$ indicates that the user u has written a helpful review of item i. Similarly, 0 indicates that the review was unhelpful. Finally, we build a new user profile that contains only helpful reviews with the value 1 based on the classification results.

3.3. Phase 3: Recommendation Generator

To evaluate the performance of the proposed recommendation framework, we predict preference ratings by applying the UBCF, SVD, and NCF models, which are typically used in personalized recommendation services-related studies.

The first is the UBCF model. UBCF approach is the standard approach that is based on neighborhood models in recommender systems. The most common UBCF measures similarity between users, where $sim(u, v)$ represents user u and user v similarity [58,59]. The goal of this technique is to predict the user u preference rating $\hat{r}_{ui}$ for item i. Using the similarity measure, we identify the items rated by user u, most similar to i. The predicted rating is taken as a weighted sum of the ratings for neighborhood users, defined as follows:

$$\hat{r}_{ui} = \overline{r}_u + \frac{\sum\limits_{v \in N_i^k(u)} sim(u, v) \cdot (r_{vi} - \overline{r}_v)}{\sum\limits_{v \in N_i^k(u)} sim(u, v)} \tag{10}$$

The second is the SVD model. The latent factor approach has gained its popularity due to its high accuracy and scalability. This study focuses on methods that SVD of the user–item interaction matrix induces. The most common approach to estimating interaction components is the matrix factorization framework [1,12]. A common approach widely used in research relates each latent factor vector of a user to a latent factor vector for the item. Typically, this approach is applied to explicit feedback datasets while addressing overfitting issues through a regularized model. The SVD model is defined as follows:

$$\min_{U.V} \|Y - M \odot (UV)\|_F^2 + \lambda(\|U\|_F^2 + \|V\|_F^2), \tag{11}$$

where U and V indicates the number of latent factor users and items, respectively, and λ is used for regularizing the model. Y is the available ratings set, and M is the binary mask.

The third is the NCF model. The traditional latent factor model utilized a simple vector dot item to estimate the relationship latent vector. Therefore, such an approach cannot produce excellent results. To solve the latent factor technique's limitations, the NCF model captures the interaction between the user's latent vector and the item's latent vector through a multi-layer perceptron [60,61]. The user's latent vector and the item's latent vector are inputs to multi-layer perceptron to predict user preferences. The output layer is used to predict user preference, and the model performs learning by minimizing the loss between the prediction and actual ratings. The NCF predictive model is defined as follows:

$$\hat{r}_{ui} = f(U^T \cdot s_u^{user}, V^T \cdot s_i^{item} | U, V, \theta), \tag{12}$$

where s_u^{user} and s_i^{item} denote that the input layer consists of two feature vectors. U and V denote the latent factors for the user and item, respectively, and θ denotes the model's parameter.

4. Experiments

4.1. Dataset Overview

We used Amazon Book (http://jmcauley.ucsd.edu/data/amazon/, accessed on 1 May 2021) publicly accessible datasets to evaluate the proposed performance of the RHRM framework [62,63]. The original datasets were collected from May 1996 to July 2014 and contain 8,872,495 reviews from 817,789 users on 562,073 items. Table 2 displays an example of attribution information from the Amazon Book Dataset. Each review contains (1) the ID and name of the reviewer, (2) the ID of the reviewed item, (3) the helpfulness information that including the number of helpful votes and the number of unhelpful votes, (4) rating information, (5) summary reviews and detailed reviews on the item, and (6) reviews published time.

Table 2. An example Amazon Book dataset review composition.

Attribute Name	Value
Reviewer ID	A2S166WSCFIFP5
Item ID	000100039X
Reviewer name	Adam
Number of helpful votes	10
Total number of votes	25
Review text	I evidently misread the writeup, I thought it was a hardback. It was a cheap paperback. I got it as a present so I couldn't send it back, but I'm very disappointed for the cost!
Rating	3
Summary headline	Not Bad!
Review time	2012-10-10

To conduct experiments effectively, we have built the CNN–BiLSTM hybrid model using the dataset (DS1) collected from May 1996 to December 2011, which contains 2,757,812 reviews from 281,661 users on 223,452 items. In addition, to evaluate the proposed recommendation framework performance, we use the dataset (DS2) collected from January 2012 to July 2014, which contains 6,114,683 reviews from 536,128 users on 338,621 items. The descriptive statistics of the two datasets are summarized in Table 3.

Table 3. Descriptive statistics of the two datasets.

Dataset	Period	User	Item	Rating and Reviews
DS1	May 1996–December 2011	281,661	223,452	2,757,812
DS2	January 2012–July 2014	536,128	338,621	6,114,683

Among these reviews in DS1, only total voting by at least 10 users as helpful or unhelpful are regarded as a training dataset for helpfulness classification [17,64]. Following the exiting study's common strategy, we measured helpfulness score as the ratio of helpful votes to the total votes. The distribution of the measured helpfulness score is depicted in Figure 3. To better classify helpful or unhelpful reviews, we preferred only highly helpful reviews ($\theta_1 > 0.9$) and unhelpful reviews ($\theta_2 < 0.2$) as the training dataset. Figure 4 shows examples of helpful reviews and unhelpful reviews. With this filtered dataset, we train binary models for review helpfulness classification. The DS2 volume is large but highly sparse. Therefore, we filtered the dataset to contain only users with at least 20 interactions [60].

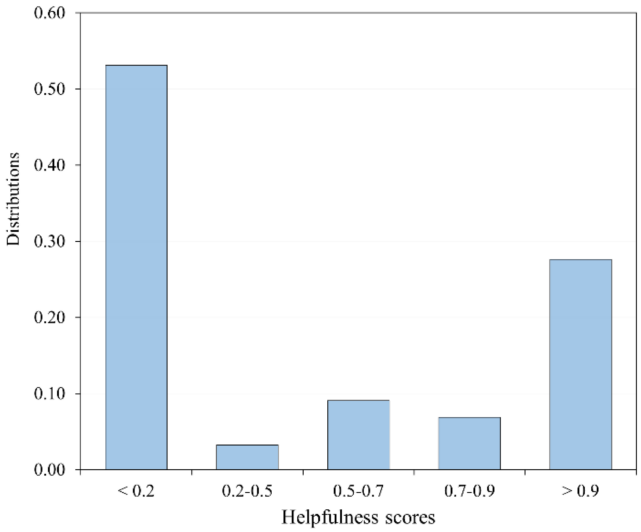
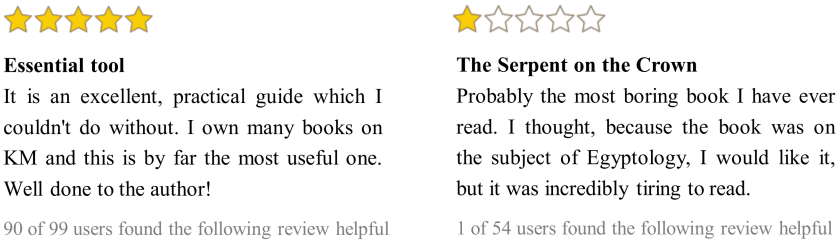


Figure 3. Distributions of helpfulness scores.



(a) Helpful Review Example (b) Unhelpful Review Example

Figure 4. Examples of helpful reviews and unhelpful reviews.

4.2. Evaluation Protocols

To evaluate CNN–BiLSTM hybrid model classification performance in this study, we experimented with DS1 and adopted Accuracy, Precision, Recall, and F1-score as metrics. Furthermore, to evaluate the prediction performance of the proposed recommendation framework, we experimented with DS2 and adopted Mean Absolute Error (MAE) and Root Mean Square Error (RMSE) metrics. We set 80% of each dataset as a training dataset and measure the performance with the remaining dataset [12,58,59].

First, to evaluate the classification performance of the CNN–BiLSTM hybrid model, we adopted Accuracy, Precision, Recall, and F1-score as metrics using the confusion matrix shown in Table 4. Accuracy is the most used evaluation metric when measuring classification performance and represents the number of accurate classifications ratio of helpful and unhelpful reviews in the total classification results. Precision represents the contained ratio of actual helpful reviews to the classified helpful review by the model. The recall represents the contained ratio of the classified helpful review by the model to actual helpful reviews. The F1 score represents a balance weight average between precision and recall. A higher F1 score means a higher classification ability of the recommender system. The Accuracy, Precision, Recall, and F1-Score are defined in Equations (13)–(16).

$$Accuracy = \frac{TP + TN}{TP + FN + TN + FP} \tag{13}$$

$$Precision = \frac{TP}{TP + FP} \tag{14}$$

$$Recall = \frac{TP}{TP + FN} \tag{15}$$

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (16)$$

Table 4. Confusion matrix example for evaluating the performance of helpfulness classification.

Actual Class \ Predicted Class	Helpful	Unhelpful
Helpful	TP	FN
Unhelpful	FP	TN

The MAE and RMSE are statistical accuracy metric that evaluate prediction performance by comparing the difference between predicted and actual ratings, as defined in Equations (17) and (18) [7,10]. The MAE gives the same weight regardless of the magnitude of the error between the actual and predicted ratings. However, RMSE gives a relatively high value weight with a large error between the actual and predicted ratings. When the value is low, the corresponding recommendation prediction is more accurate.

$$MAE = \frac{1}{N}\sum_{i=1}^{N}|y_{ui} - \hat{y}_{ui}| \quad (17)$$

$$RMSE = \sqrt{\frac{1}{N}\sum_{i=1}^{N}(y_{ui} - \hat{y}_{ui})^2} \quad (18)$$

where N is the total test dataset, $\hat{y}_{ui}$ is the predicted rating, and y_{ui} is the actual rating by the user u for the item i.

4.3. Parameter Settings

For performing review text data preprocessing, we applied the NLTK (Natural Language Toolkit) package to remove stopwords, special characters, symbols, numbers, etc., included in the review [65,66]. For training CNN–BiLSTM hybrid model, we set an embedding dimension of 300, filter windows of 3, 4, and 5, a filter size of 100, and the number of hidden units in Bi-LSTM of 64. In addition, to solve the overfitting problem, the dropout rate was set to 0.5, batch size was set to 50, and Epoch was set to 100. As the optimization algorithm, Adam, which is widely used in previous studies, was applied, and the learning rate was set to 0.05 [67]. We set the review length of average and maximum length and the vocabulary size of several sizes [68]. Then, we set the optimal review length and vocabulary size based on classification performance. We applied the same parameters to baseline algorithms and compared the classification performance. For the CF algorithm, Pearson Correlation Coefficients measures similarity between users, and the neighbor size is set from 1 to 100. Furthermore, we set the latent factor sizes of SVD and NCF techniques to 8, 16, 32, 64, and 128 [60]. Experiments in this study were conducted using TensorFlow, Keras, and Surprise packages. All experiments were conducted in a computer environment with CPU Intel Core i9-9900KF, 64G of memory, and a GeForce RTX 2080 Ti.

4.4. Experimental Result

4.4.1. Review Helpfulness Classification Performance Comparison

In this section, we first study the effect of changes in vocabulary size and review length on the classification performance of the CNN–BiLSTM hybrid model. To retain the main semantics and suppress noise, first, we performed several experiments that set several vocabulary sizes from 20,000 to 104,702. Then, we select the maximum review length and the average review length for each vocabulary size to conduct experiments. Finally, we set the optimized vocabulary size and review length to train the CNN–BiLSTM hybrid model efficiently. We conducted the experiment five times and reported the mean and the standard deviation of the classification performance. Table 5 indicates the mean

and the standard deviation of classification performances of the CNN–BiLSTM hybrid models with several vocabulary sizes and review lengths. We found that the performance worsened when the vocabulary sizes were set too high. Thus, optimal vocabulary sizes should be used to improve the recommendation performance. In addition, we found that maximum review lengths should be used to improve the recommendation performance. As a result, the optimal vocabulary size was 80,000, and the maximum length of the review should be chosen. We set the optimal vocabulary size and review length to compare the classification performance with the baseline model.

Table 5. Mean and standard deviation for classification performance on the different vocabulary sizes and reviews length for the CNN–BiLSTM hybrid model.

Vocabulary Size	Review Length (max/mean)	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
20,000	2703	81.88 ± 0.44	81.44 ± 0.39	82.25 ± 0.41	81.84 ± 0.40
	110	81.02 ± 0.40	80.13 ± 0.38	82.96 ± 0.36	81.52 ± 0.37
40,000	2795	84.42 ± 0.39	83.57 ± 0.32	85.43 ± 0.34	84.48 ± 0.33
	113	83.88 ± 0.32	82.92 ± 0.38	81.65 ± 0.35	82.28 ± 0.36
60,000	2815	82.26 ± 0.24	83.25 ± 0.29	81.95 ± 0.31	82.59 ± 0.30
	114	82.10 ± 0.41	82.71 ± 0.38	81.65 ± 0.35	82.17 ± 0.36
80,000	2817	86.14 ± 0.35	85.54 ± 0.33	88.73 ± 0.34	87.10 ± 0.33
	115	82.18 ± 0.31	81.73 ± 0.29	84.39 ± 0.39	83.03 ± 0.35
104,702 (Maximum)	2837	83.19 ± 0.25	84.56 ± 0.34	80.15 ± 0.31	82.29 ± 0.32
	115	82.26 ± 0.41	82.95 ± 0.39	84.32 ± 0.33	83.62 ± 0.36

We set the optimal number of words and review length and compared the CNN–BiLSTM hybrid model with the baseline to evaluate the classification performance. We experimented five times and reported that the mean and the standard deviation of the classification performance are shown in Figure 5. The CNN–BiLSTM hybrid model outperforms other baseline models with an accuracy of 86.71% and F1-Score of 86.43%. Although the CNN single model represents an excellent classification effect, the other deep learning models are better than CNN. Compared to CNN, BiLSTM single models, the CNN–BiLSTM hybrid model shows the advantages of combined networks in semantic representation extraction. Because CNN models for word vectors are conducive to reprocessing CNN feature extraction by BiLSTM, we can find that adding an attention mechanism to the combination model can effectively enhance classification performance. The attention mechanism helps the model learning essential features by assigning different weights and learning differences between different features.

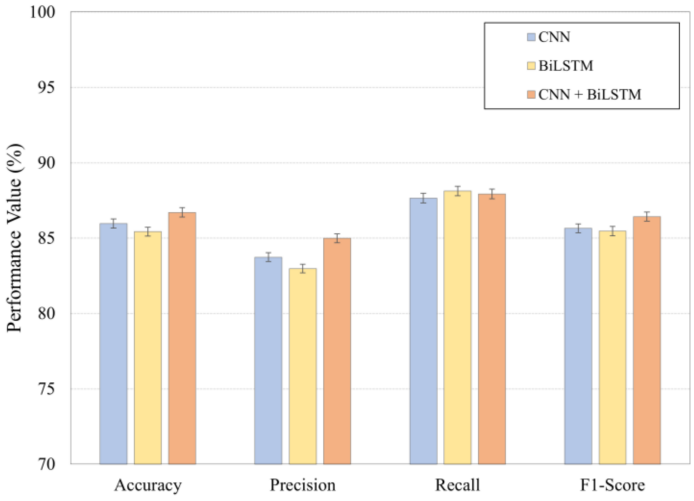


Figure 5. Performance comparison of classification techniques.

4.4.2. Prediction Performance Comparison Based on Helpful Review Filtering

This session identifies the effectiveness of the framework proposed in this study. Firstly, we have classified whether the new reviews written by users were helpful through the CNN-BiLSTM hybrid model. Then, we have produced a new user profile by filtering only helpful reviews. Comparing the existing recommendation methodology with the proposed RHRM framework in this study with the prediction performance through UBCF, SVD, and NCF techniques, respectively, are shown in Figures 6–8, where "Existing" represents a traditional recommendation methodology that produces user profiles, including all reviews. "Proposal" represents a proposed recommendation framework that produces user profiles, which includes only helpful reviews. We have set the neighbor sizes from 1 to 100 to evaluate the prediction performance of changing the neighbor size in the UBCF technique. We set the latent factor sizes of SVD and NCF techniques to 8, 16, 32, 64, and 128 and compared the prediction performance. MAE and RMSE metrics are used to measure the prediction performance for the error between predict rating and actual ratings.

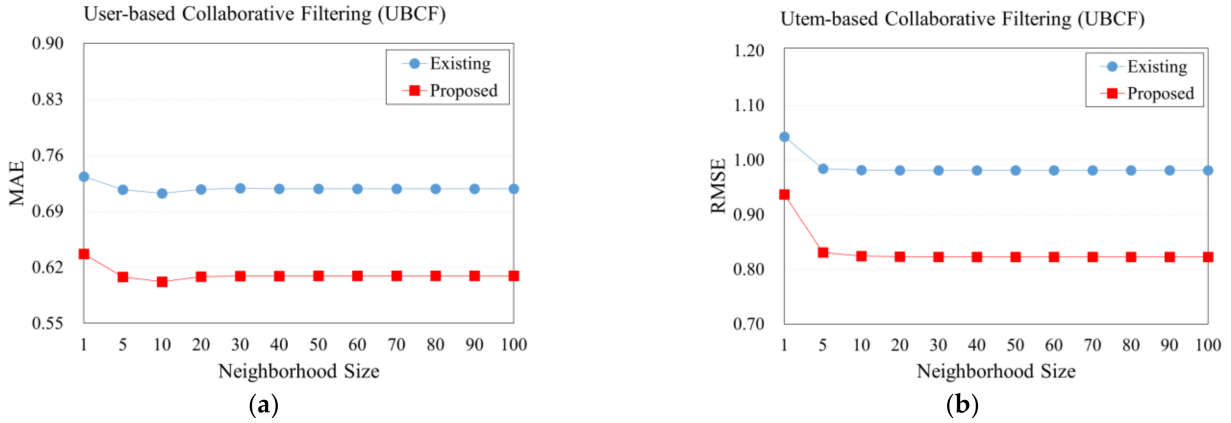


Figure 6. Prediction performance of MAE (**a**) and RMSE (**b**) of UBCF model.

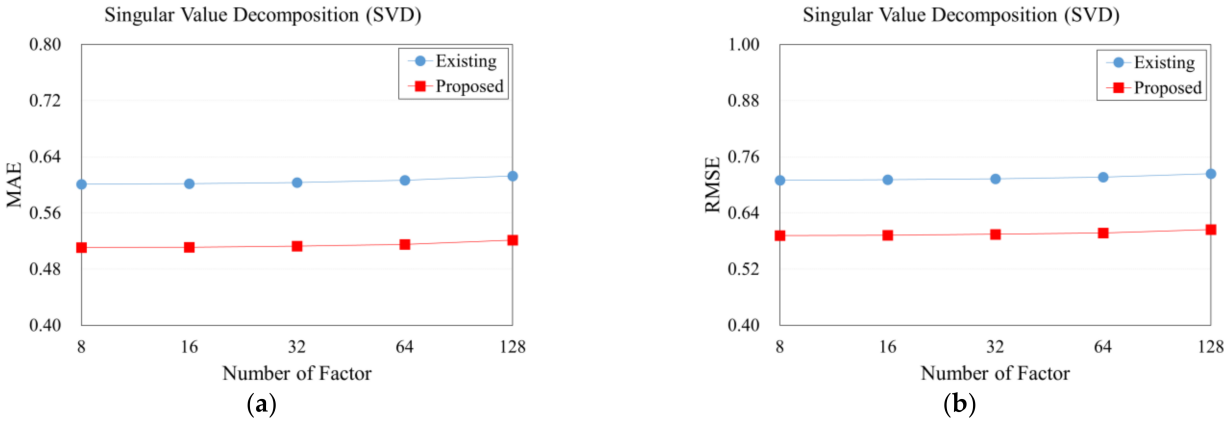


Figure 7. Prediction performance of MAE (**a**) and RMSE (**b**) of SVD model.

The results of the experiment show that the prediction performance of the proposed recommendation framework has improved regardless of the neighbor size and number of latent factors. When we have applied both MAE and RMSE metrics for the UBCF technique, both metrics showed excellent prediction performance regardless of the neighbor size. It showed the best prediction performance when neighbor size is 10. The SVD and NCF technique indicate excellent prediction performance when the number of latent factors is 8 and 32, respectively. Therefore, we have compared the proposed framework to the existing methodology: when using the MAE metric, the prediction performance improved 14.95% (UBCF), 14.99% (SVD), and 22.08% (NCF), respectively. Similarly, using the RMSE

metric, the prediction performance improved 15.38% (UBCF), 16.58% (SVD), and 21.59% (NCF). Experiments show that producing user profiles using only helpful reviews results in a better prediction performance than the existing methodology. Therefore, reflecting review helpfulness information on personalized recommendation services can improve the performance of recommender systems, which we have further conducted two-sample *t*-tests as shown in Table 6 to confirm that all improvements were statistically significant for $p < 0.01$.

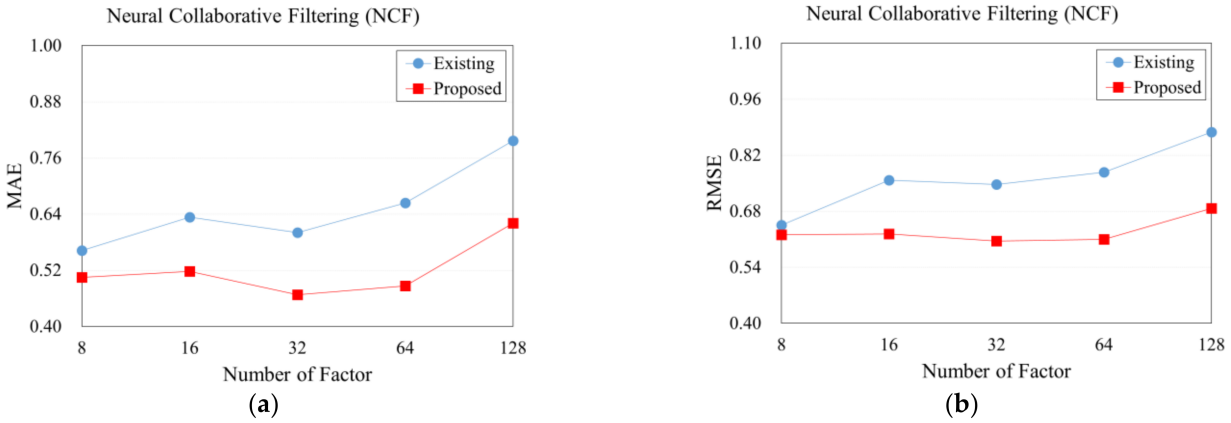


Figure 8. Prediction performance of MAE (**a**) and RMSE (**b**) of NCF model.

Table 6. Two samples t-tests between existing and proposed framework in several recommendation method types.

Method	Metrics Factor		Mean	Standard Deviation	*t*-Statistics	*p*-Value
UBCF	MAE	Existing	0.716	0.435	32.007	0.000
		Proposal	0.617	0.355		
	RMSE	Existing	0.870	0.498	39.388	0.000
		Proposal	0.732	0.402		
SVD	MAE	Existing	0.601	0.374	33.589	0.000
		Proposal	0.511	0.311		
	RMSE	Existing	0.710	0.426	38.775	0.000
		Proposal	0.592	0.351		
NCF	MAE	Existing	0.600	0.422	41.192	0.000
		Proposal	0.468	0.383		
	RMSE	Existing	0.747	0.506	37.542	0.000
		Proposal	0.605	0.444		

5. Conclusions

5.1. Discussion

We propose a novel RHRM recommendation framework that filters only helpful reviews and reflects them in the personalized recommendation service. To achieve our study objective, we built CNN-BiLSTM hybrid models that demonstrate excellent classification performance in NLP studies to filter helpful reviews. We have also evaluated the performance of the proposed recommendation framework in this study by utilizing UBCF, SVD, and NCF techniques that are widespread in the use of recommender systems studies. To evaluate the recommendation performance, we used large numbers of Amazon publicly accessible datasets [62,63]. The experimental results show that the RHRM framework outperforms the prediction performance of the existing recommendation framework without regard to review helpfulness. Experimental results also suggest that the review's helpfulness information can significantly impact user preference ratings. In other words, users' high quality of reviews can provide higher reliability than preference rating information given by users [14]. Furthermore, we have identified that the CNN-BiLSTM hybrid model

used in this study outperforms other deep learning models such as CNN and Bi-LSTM single model. This demonstrates the advantages of the CNN-BiLSTM hybrid model in semantic feature extraction. We have also identified that the classification performance of the CNN-BiLSTM hybrid model depends on the vocabulary size and the review length used in model training. We found that when the word size was 80,000, and the review length was maximum, and the model indicates excellent classification performance through various experiments. Because using all vocabulary as training data includes noise features that are insignificant to the analysis, this result in increased computational costs, time, and reduced classification performance [17].

5.2. Theoretical Contributions and Practical Implications

In this study, we have enhanced the recommendation performance by analyzing review helpfulness information through deep learning techniques and reflecting them in recommender systems. The theoretical implications of this study are as follows: First, the existing studies on personalized recommendation services used all reviews included in the item to extract the sentiment features and reflect them in the recommender system. However, user-written reviews include advertisements, falsehoods, and unknown content reviews [69]. In other words, if reviews are irrelevant to items and unhelpful to users, they can reduce recommendation performance. Therefore, we proposed a recommendation framework to classification review helpfulness information and reflect them in recommender systems. In this study, we have improved the recommender systems' performance by using the review helpfulness information. This result can contribute to the extended scope of the personalized recommendation service-related studies. Second, to evaluate the recommender system's performance considering the review helpfulness information, we compared the results considering the review helpfulness as well as not considering the review helpfulness. The experimental results showed that the recommendation performance was higher when considering review helpfulness information. Therefore, besides features, price, and users' sentiment, the review helpfulness information is essential in purchasing decision making. Furthermore, objective information such as the number of review helpfulness votes influences users' preference more than subjective user-written reviews.

The practical implications of this study are as follows: First, we proposed a recommendation framework that classified the review helpfulness information and reflected them in the personalized recommendation services. We conducted several experiments and found that considering helpful reviews can enhance recommendation performance over traditional methods. Most e-commerce websites provide a module for writing reviews of items purchased by users. Nonetheless, few e-commerce websites have reflected the helpfulness information in reviews. Therefore, it needs to provide services that could evaluate the review helpfulness information. For example, if the review helpfulness information is evaluated with a high score, it can increase the review information value of the item by providing users with mileage or coupons. Second, most e-commerce websites have focused on item reviews and encouraged users to write reviews on items. We found that the quality of the review is more critical than the number of reviews when providing personalized recommendation services. Therefore, rather than increasing the number of reviews, it requires a strategy that encourages users to write high-quality reviews. Finally, the proposed recommendation framework in this study can apply to the various domains of e-commerce websites that provide review usefulness information. This enables the website to build more sophisticated recommendation services, providing decision support in many aspects, including marketing and user management. Therefore, e-commerce websites can increase the convenience and satisfaction of the users and expect sales growth.

5.3. Limitations and Future Study

We have classified the review helpfulness information through the CNN-BiLSTM hybrid model. We then conducted an experiment based on the proposed recommendation

framework to evaluate the recommendation performance. The limitations of this study are as follows: First, we only used Amazon publicly accessible book datasets. We built a CNN-BiLSTM hybrid model using all the datasets without classifying the book category. However, users may have different preferences depending on the book category. In future studies, it is necessary to classify book categories and measure additional recommendation effects. Additionally, applying the proposed recommendation framework to other domains must be evaluated using datasets from multiple domains. Second, to classify the review helpfulness information, we applied the CNN-BiLSTM hybrid model that showed excellent performance in NLP studies. Recently, BERT, ELECTRA, and GPT-3 models have shown excellent performance in NLP studies. Therefore, future study needs to compare the performance of multiple deep learning models. Third, we proposed a recommendation framework that classifies reviews helpfulness information and then builds users' profiles with helpful reviews to provide recommendation services. In other words, we proposed a framework that only used the review helpfulness information. However, considering item features, purchase history, and other information would further improve recommendation performance. Finally, the early-written review received more helpful votes than the later written review. As this may create a sequential bias problem, future studies should consider the dates of the written review.

Author Contributions: Conceptualization, J.K. and Q.L.; methodology, Q.L. and X.L.; data curation, Q.L. and X.L.; writing—original draft preparation, Q.L. and B.L.; writing—review and editing, Q.L. and J.K.; supervision, J.K. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by the BK21 FOUR Program (5199990913932) was funded by the Ministry of Education (MOE, Korea) and the National Research Foundation of Korea (NRF).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data are available on http://jmcauley.ucsd.edu/data/amazon/ (accessed on 1 May 2021).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Lu, J.; Wu, D.; Mao, M.; Wang, W.; Zhang, G. Recommender system application developments: A survey. *Decis. Support Syst.* **2015**, *74*, 12–32. [CrossRef]
2. Bobadilla, J.; Ortega, F.; Hernando, A.; Gutiérrez, A. Recommender systems survey. *Knowl.-Based Syst.* **2013**, *46*, 109–132. [CrossRef]
3. Das, A.S.; Datar, M.; Garg, A.; Rajaram, S. Google news personalization: Scalable online collaborative filtering. In Proceedings of the 16th International Conference on World Wide Web, Banff, AB, Canada, 8–12 May 2007; pp. 271–280.
4. Linden, G.; Smith, B.; York, J. Amazon.com recommendations: Item-to-item collaborative filtering. *IEEE Internet Comput.* **2003**, *7*, 76–80. [CrossRef]
5. Bennett, J.; Lanning, S. The netflix prize. In Proceedings of the KDD Cup and Workshop, San Jose, CA, USA, 12 August 2007; p. 35.
6. Lee, D.; Hosanagar, K. How do recommender systems affect sales diversity? A cross-category investigation via randomized field experiment. *Inf. Syst. Res.* **2019**, *30*, 239–259. [CrossRef]
7. Kim, J.; Choi, I.; Li, Q. Customer satisfaction of recommender system: Examining accuracy and diversity in several types of recommendation approaches. *Sustainability* **2021**, *13*, 6165. [CrossRef]
8. Kim, H.K.; Ryu, Y.U.; Cho, Y.; Kim, J.K. Customer-driven content recommendation over a network of customers. *IEEE Trans. Syst. Man Cybern.-Part A Syst. Hum.* **2011**, *42*, 48–56. [CrossRef]
9. Choi, K.; Yoo, D.; Kim, G.; Suh, Y. A hybrid online-product recommendation system: Combining implicit rating-based collaborative filtering and sequential pattern analysis. *Electr. Commer. Res. Appl.* **2012**, *11*, 309–317. [CrossRef]
10. Park, D.H.; Kim, H.K.; Choi, I.Y.; Kim, J.K. A literature review and classification of recommender systems research. *Expert Syst. Appl.* **2012**, *39*, 10059–10072. [CrossRef]
11. Kim, H.K.; Kim, J.K.; Ryu, Y.U. Personalized recommendation over a customer network for ubiquitous shopping. *IEEE Trans. Serv. Comput.* **2009**, *2*, 140–151. [CrossRef]

12. Ricci, F.; Rokach, L.; Shapira, B. Introduction to recommender systems handbook. In *Recommender Systems Handbook*; Springer: Berlin/Heidelberg, Germany, 2011; pp. 1–35.
13. Li, X.; Wang, M.; Liang, T.-P. A multi-theoretical kernel-based approach to social network-based recommendation. *Decis. Support Syst.* **2014**, *65*, 95–104. [CrossRef]
14. Qiu, L.; Gao, S.; Cheng, W.; Guo, J. Aspect-based latent factor model by integrating ratings and reviews for recommender system. *Knowl. -Based Syst.* **2016**, *110*, 233–243. [CrossRef]
15. Moore, S.G. Attitude predictability and helpfulness in online reviews: The role of explained actions and reactions. *J. Consum. Res.* **2015**, *42*, 30–44. [CrossRef]
16. Srifi, M.; Oussous, A.; Ait Lahcen, A.; Mouline, S. Recommender systems based on collaborative filtering using review texts—A survey. *Information* **2020**, *11*, 317. [CrossRef]
17. Ge, S.; Qi, T.; Wu, C.; Wu, F.; Xie, X.; Huang, Y. Helpfulness-aware review based neural recommendation. *CCF Trans. Pervasive Comput. Interact.* **2019**, *1*, 285–295. [CrossRef]
18. Hu, Y.-H.; Chen, Y.-L.; Chou, H.-L. Opinion mining from online hotel reviews–a text summarization approach. *Inf. Process. Manag.* **2017**, *53*, 436–449. [CrossRef]
19. Kaushik, K.; Mishra, R.; Rana, N.P.; Dwivedi, Y.K. Exploring reviews and review sequences on e-commerce platform: A study of helpful reviews on Amazon. in. *J. Retail. Consum. Serv.* **2018**, *45*, 21–32. [CrossRef]
20. Castelli, M.; Manzoni, L.; Vanneschi, L.; Popovič, A. An expert system for extracting knowledge from customers' reviews: The case of amazon. com, inc. *Expert Syst. Appl.* **2017**, *84*, 117–126. [CrossRef]
21. Na, H.; Nam, K. Application of diversity of recommender system according to user preference change. *J. Intell. Inf. Syst.* **2020**, *26*, 67–86.
22. Paradarami, T.K.; Bastian, N.D.; Wightman, J.L. A hybrid recommender system using artificial neural networks. *Expert Syst. Appl.* **2017**, *83*, 300–313. [CrossRef]
23. Kim, H.K.; Oh, H.Y.; Gu, J.C.; Kim, J.K. Commenders: A recommendation procedure for online book communities. *Electron. Commer. Res. Appl.* **2011**, *10*, 501–509. [CrossRef]
24. Lee, Y.; Won, H.; Shim, J.; Ahn, H. A hybrid collaborative filtering-based product recommender system using search keywords. *J. Intell. Inf. Syst.* **2020**, *26*, 151–166.
25. Su, X.; Khoshgoftaar, T.M. A survey of collaborative filtering techniques. *Adv. Artif. Intell.* **2009**, *2009*, 1–19. [CrossRef]
26. Al-Bashiri, H.; Abdulgabber, M.A.; Romli, A.; Kahtan, H. An improved memory-based collaborative filtering method based on the TOPSIS technique. *PLoS ONE* **2018**, *13*, e0204434. [CrossRef]
27. Elahi, M.; Ricci, F.; Rubens, N. A survey of active learning in collaborative filtering recommender systems. *Comput. Sci. Rev.* **2016**, *20*, 29–50. [CrossRef]
28. Breese, J.S.; Heckerman, D.; Kadie, C. Empirical analysis of predictive algorithms for collaborative filtering. In Proceedings of the 14th Conference on Uncertainty in Artificial Intelligence, Morgan Kaufmann, San Francisco, CA, USA, 24–26 July 1998; pp. 43–52.
29. Isinkaye, F.O.; Folajimi, Y.O.; Ojokoh, B.A. Recommendation systems: Principles, methods and evaluation. *Egypt. Inform. J.* **2015**, *16*, 261–273. [CrossRef]
30. Bang, H.; Lee, H.; Lee, J.-H. TV Program recommender system using viewing time patterns. *J. Korean Inst. Intell. Syst.* **2015**, *25*, 431–436. [CrossRef]
31. Guy, I.; Mejer, A.; Nus, A.; Raiber, F. Extracting and ranking travel tips from user-generated reviews. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 987–996.
32. Leung, C.W.; Chan, S.C.; Chung, F.-l. Integrating collaborative filtering and sentiment analysis: A rating inference approach. In Proceedings of the ECAI 2006 Workshop on Recommender Systems, Riva del Garda, Italy, 28 August–1 September 2006; pp. 62–66.
33. García-Cumbreras, M.Á.; Montejo-Ráez, A.; Díaz-Galiano, M.C. Pessimists and optimists: Improving collaborative filtering through sentiment analysis. *Expert Syst. Appl.* **2013**, *40*, 6758–6765. [CrossRef]
34. Zhang, Z.; Zhang, D.; Lai, J. urCF: User review enhanced collaborative filtering. In Proceedings of the 20th Americas Conference on Information Systems, Savannah, GA, USA, 7–9 August 2014; pp. 1–14.
35. Zhou, L.; Chaovalit, P. Ontology-supported polarity mining. *J. Am. Soc. Inf. Sci. Technol.* **2008**, *59*, 98–110. [CrossRef]
36. Jeon, B.; Ahn, H. A collaborative filtering system combined with users' review mining: Application to the recommendation of smartphone apps. *J. Intell. Inf. Syst.* **2015**, *21*, 1–18.
37. Hyun, J.; Ryu, S.; Lee, S.-Y.T. How to improve the accuracy of recommendation systems: Combining ratings and review texts sentiment scores. *J. Intell. Inf. Syst.* **2019**, *25*, 219–239.
38. Cheng, Z.; Ding, Y.; Zhu, L.; Kankanhalli, M. Aspect-aware latent factor model: Rating prediction with ratings and reviews. In Proceedings of the 2018 World Wide Web Conference, Lyon, France, 23–27 April 2018; pp. 639–648.
39. Gan, C.; Feng, Q.; Zhang, Z. Scalable multi-channel dilated CNN–BiLSTM model with attention mechanism for Chinese textual sentiment analysis. *Future Gener. Comput. Syst.* **2021**, *118*, 297–309. [CrossRef]
40. Deng, J.; Cheng, L.; Wang, Z. Attention-based BiLSTM fused CNN with gating mechanism model for Chinese long text classification. *Comput. Speech Lang.* **2021**, *68*, 101182. [CrossRef]
41. Stojanovski, D.; Strezoski, G.; Madjarov, G.; Dimitrovski, I.; Chorbev, I. Deep neural network architecture for sentiment analysis and emotion identification of Twitter messages. *Multimed. Tools Appl.* **2018**, *77*, 32213–32242. [CrossRef]

42. Song, Y.; Hu, Q.V.; He, L. P-CNN: Enhancing text matching with positional convolutional neural network. *Knowl.-Based Syst.* **2019**, *169*, 67–79. [CrossRef]
43. Abdi, A.; Shamsuddin, S.M.; Hasan, S.; Piran, J. Deep learning-based sentiment classification of evaluative text based on Multi-feature fusion. *Inf. Process. Manag.* **2019**, *56*, 1245–1259. [CrossRef]
44. Rao, G.; Huang, W.; Feng, Z.; Cong, Q. LSTM with sentence representations for document-level sentiment classification. *Neurocomputing* **2018**, *308*, 49–57. [CrossRef]
45. Hassan, A.; Mahmood, A. Efficient deep learning model for text classification based on recurrent and convolutional layers. In Proceedings of the 2017 16th IEEE International Conference on Machine Learning and Applications (ICMLA), Cancun, Mexico, 18–21 December 2017; pp. 1108–1113.
46. Hassan, A.; Mahmood, A. Convolutional recurrent deep learning model for sentence classification. *IEEE Access* **2018**, *6*, 13949–13957. [CrossRef]
47. Liu, G.; Guo, J. Bidirectional LSTM with attention mechanism and convolutional layer for text classification. *Neurocomputing* **2019**, *337*, 325–338. [CrossRef]
48. Batbaatar, E.; Li, M.; Ryu, K.H. Semantic-emotion neural network for emotion recognition from text. *IEEE Access* **2019**, *7*, 111866–111878. [CrossRef]
49. Zheng, J.; Zheng, L. A hybrid bidirectional recurrent convolutional neural network attention-based model for text classification. *IEEE Access* **2019**, *7*, 106673–106685. [CrossRef]
50. Liu, Z.-x.; Zhang, D.-g.; Luo, G.-z.; Lian, M.; Liu, B. A new method of emotional analysis based on CNN–BiLSTM hybrid neural network. *Clust. Comput.* **2020**, *23*, 2901–2913. [CrossRef]
51. Rai, A.; Shrivastava, A.; Jana, K.C. A CNN-BiLSTM based deep learning model for mid-term solar radiation prediction. *Int. Trans. Electr. Energy Syst.* **2020**, *31*, e12664. [CrossRef]
52. Jang, B.; Kim, M.; Harerimana, G.; Kang, S.-u.; Kim, J.W. Bi-LSTM model to increase accuracy in text classification: Combining Word2vec CNN and attention mechanism. *Appl. Sci.* **2020**, *10*, 5841. [CrossRef]
53. Rhanoui, M.; Mikram, M.; Yousfi, S.; Barzali, S. A CNN-BiLSTM model for document-level sentiment analysis. *Mach. Learn. Knowl. Extr.* **2019**, *1*, 832–847. [CrossRef]
54. Cao, R.; Zhang, X.; Wang, H. A review semantics based model for rating prediction. *IEEE Access* **2019**, *8*, 4714–4723. [CrossRef]
55. Mitra, S.; Jenamani, M. Helpfulness of online consumer reviews: A multi-perspective approach. *Inf. Process. Manag.* **2021**, *58*, 102538. [CrossRef]
56. Chen, T.; Xu, R.; He, Y.; Wang, X. Improving sentiment analysis via sentence type classification using BiLSTM-CRF and CNN. *Expert Syst. Appl.* **2017**, *72*, 221–230. [CrossRef]
57. Kim, Y. Convolutional neural networks for sentence classification. In Proceedings of the EMNLP, Doha, Qatar, 25–29 October 2014.
58. Ekstrand, M.D.; Riedl, J.T.; Konstan, J.A. *Collaborative Filtering Recommender Systems*; Now Publishers Inc.: Norwell, MA, USA, 2011.
59. Herlocker, J.L.; Konstan, J.A.; Terveen, L.G.; Riedl, J.T. Evaluating collaborative filtering recommender systems. *ACM Trans. Inf. Syst. (TOIS)* **2004**, *22*, 5–53. [CrossRef]
60. He, X.; Liao, L.; Zhang, H.; Nie, L.; Hu, X.; Chua, T.-S. Neural collaborative filtering. In Proceedings of the 26th International Conference on World Wide Web, Perth, Australia, 3–7 April 2017; pp. 173–182.
61. Zhang, S.; Yao, L.; Sun, A.; Tay, Y. Deep learning based recommender system: A survey and new perspectives. *ACM Comput. Surv. (CSUR)* **2019**, *52*, 1–38. [CrossRef]
62. He, R.; McAuley, J. Ups and downs: Modeling the visual evolution of fashion trends with one-class collaborative filtering. In Proceedings of the 25th International Conference on World Wide Web, Montreal, QC, Canada, 11–15 April 2016; pp. 507–517.
63. McAuley, J.; Targett, C.; Shi, Q.; Van Den Hengel, A. Image-based recommendations on styles and substitutes. In Proceedings of the 38th International ACM SIGIR Conference on Research and Development in Information Retrieval, Santiago, Chile, 9–13 August 2015; pp. 43–52.
64. Liu, Y.; Huang, X.; An, A.; Yu, X. Modeling and predicting the helpfulness of online reviews. In Proceedings of the 2008 Eighth IEEE International Conference on Data Mining, Pisa, Italy, 15–19 December 2008; pp. 443–452.
65. Park, S.; Woo, J. Gender classification using sentiment analysis and deep learning in a health web forum. *Appl. Sci.* **2019**, *9*, 1249. [CrossRef]
66. Yoo, S.; Song, J.; Jeong, O. Social media contents based sentiment analysis and prediction system. *Expert Syst. Appl.* **2018**, *105*, 102–111. [CrossRef]
67. Kingma, D.P.; Ba, J. Adam: A method for stochastic optimization. *arXiv* **2014**, arXiv:1412.6980.
68. Yang, L.; Li, Y.; Wang, J.; Sherratt, R.S. Sentiment analysis for E-commerce product reviews in Chinese based on sentiment lexicon and deep learning. *IEEE Access* **2020**, *8*, 23522–23530. [CrossRef]
69. Saumya, S.; Singh, J.P. Detection of spam reviews: A sentiment analysis approach. *CSI Trans. ICT* **2018**, *6*, 137–148. [CrossRef]