Hindawi
Security and Communication Networks
Volume 2024, Article ID 9897316, 1 page
https://doi.org/10.1155/2024/9897316

Retraction

Retracted: Research on Tourism Route Recommendation Strategy Based on Convolutional Neural Network and Collaborative Filtering Algorithm

Security and Communication Networks

Received 8 January 2024; Accepted 8 January 2024; Published 9 January 2024

This article has been retracted by Hindawi following an investigation undertaken by the publisher [1]. This investigation has uncovered evidence of one or more of the following indicators of systematic manipulation of the publication process:

(1) Discrepancies in scope
(2) Discrepancies in the description of the research reported
(3) Discrepancies between the availability of data and the research described
(4) Inappropriate citations
(5) Incoherent, meaningless and/or irrelevant content included in the article
(6) Manipulated or compromised peer review

The presence of these indicators undermines our confidence in the integrity of the article's content and we cannot, therefore, vouch for its reliability. Please note that this notice is intended solely to alert readers that the content of this article is unreliable. We have not investigated whether authors were aware of or involved in the systematic manipulation of the publication process.

In addition, our investigation has also shown that one or more of the following human-subject reporting requirements has not been met in this article: ethical approval by an Institutional Review Board (IRB) committee or equivalent, patient/participant consent to participate, and/or agreement to publish patient/participant details (where relevant).

Wiley and Hindawi regrets that the usual quality checks did not identify these issues before publication and have since put additional measures in place to safeguard research integrity.

We wish to credit our own Research Integrity and Research Publishing teams and anonymous and named external researchers and research integrity experts for contributing to this investigation.

The corresponding author, as the representative of all authors, has been given the opportunity to register their agreement or disagreement to this retraction. We have kept a record of any response received.

References

[1] S. He, "Research on Tourism Route Recommendation Strategy Based on Convolutional Neural Network and Collaborative Filtering Algorithm," *Security and Communication Networks*, vol. 2022, Article ID 4659567, 9 pages, 2022.

Hindawi
Security and Communication Networks
Volume 2022, Article ID 4659567, 9 pages
https://doi.org/10.1155/2022/4659567

Research Article

Research on Tourism Route Recommendation Strategy Based on Convolutional Neural Network and Collaborative Filtering Algorithm

Shan He

Department of Foreign Languages and Tourism, Henan Institute of Economics and Trade, Zhengzhou, Henan 450000, China

Correspondence should be addressed to Shan He; happy9262022@163.com

Received 2 March 2022; Revised 3 April 2022; Accepted 12 April 2022; Published 7 May 2022

Academic Editor: Fang Liu

With improving people's living standards, tourism has become essential leisure and entertainment. At present, it has begun to shift from a quantity-oriented tourism method to a quality-oriented tourism method. It is difficult for passengers to choose the route that suits them from the numerous routes. Given the above problems, this study proposes a travel route recommendation algorithm that combines a convolutional neural network and collaborative filtering. The algorithm uses a convolutional neural network to extract the latent features in the customer and travel itinerary data and then uses the matrix factorization method based on collaborative filtering to perform score prediction. The experimental results show that the algorithm can meet the travel requirements of different customers. At the same time, the recommendation accuracy of the tourist route is improved, and technology and method are provided for realizing the personalized recommendation service of the tourist route.

1. Introduction

At present, the introduction of various tourism products has made the amount of tourism information extremely large, and it is difficult for users to quickly locate the products they are interested in from a large amount of tourism information [1]. In order to compete for tourists and increase revenue, travel companies need to continuously meet the needs of tourists and formulate travel routes that meet the interests of tourists [2]. Tourism recommendation system is an essential means to solve the problem of tourism information overload. It can actively push tourism routes that meet tourists' interests and help them make decisions quickly [3].

Tourist route recommendation algorithms mainly include four categories: content-based recommendation, collaborative filtering-based recommendation [4], knowledge-based recommendation, and social media-based recommendation [5]. Content-based travel route recommendation recommends similar routes to tourists based on the travel products they choose [6]. Recommend optional tours for tourists for a limited time. Tourist route recommendation based on collaborative filtering recommends the routes selected by tourists with similar interests according to the tourists' route preferences [7]. Knowledge-based travel route recommendation introduces tourism domain knowledge into the route recommendation system to improve the accuracy of route recommendation [8]. Travel route recommendation based on social media introduces the tourist relationship in social media into the route recommendation process [9].

The current route recommendation is generally based on the location of tourists, using geographic information systems and mobile devices to recommend surrounding routes or places [10]. But since tourist travel is often limited by climatic time, and users' interests also change, time is crucial for route selection [11]. Literature [12] recommends routes based on spending the least amount but reaching more destinations within a certain period. Literature [13] designed a personalized similarity model and used the user's heterogeneous travel information to make recommendations at a specific location at a particular moment. Literature [14] uses a time-dependent network to solve the problem of

random changes in the number of tours and attractions and selects the next attraction through conditional probability. Literature [15] proposed a time-sensitive travel route recommendation method based on dynamic transition graphs. Literature [16] designed a two-step greedy heuristic algorithm to predict the next destination. It considers space-time conflicts and solves the problem of sparse data. Literature [17] proposed a multiconstraint K-greedy algorithm based on the opening hours of scenic spots, ticket GIS coordinates, scenic spot evaluation information, etc., which can recommend better tourist routes for tourists. However, in the actual route selection process, whether a user will select a route is often affected by the theme of the route and the user's interest [18].

With the rapid development of smart mobile devices and the Internet, the problem of tourist information overload has emerged. Let tourists spend a lot of time and energy screening travel information, significantly reducing customer travel medical examination. The emergence of a travel recommendation system can assist tourists to quickly and accurately select resources that match their needs from the overloaded information. This study proposes a travel route recommendation algorithm integrating convolutional neural networks and collaborative filtering. The algorithm first converts the user comment data into words and extracts the hidden features in the user and travel route data through a convolutional neural network. Then, the collaborative filtering recommendation based on users and travel routes is completed. The algorithm can recommend scientific and reasonable travel routes for users to meet their travel needs in different situations.

2. The Algorithm Proposed in This Study

2.1. Problem Description. The collaborative filtering algorithm mines the hidden features of tourists or tourist routes according to the data of tourists' interest and preference for tourism. Then its correlations are calculated and score predictions are made. Finally, a recommendation is generated. In general, data is often very sparse. The current collaborative filtering is difficult to extract the deeply hidden features in the data.

For example, suppose there are w tourists and t travel routes. The sample data is in the form of a 4-tuple (userx, itemy , c_{xy}, r_{xy}), where r_{xy} represents the rating of tourists x on the tourist route y, which is an integer value from 1 to 5. c_{xy} represents the comment of tourist x on the tourist route y, which is a piece of text with different lengths. R is an $w * t$ rating matrix and C is an $w * t$ review matrix, where $r_{xy} \in R^{w\times t}, c_{xy} \in C^{w\times t}$, as shown in Table 1.

In Table 1, each row represents the ratings and reviews published by a tourist. Each column represents the ratings and reviews received by a tourist route. ? means no rating comment information. The matrices R and C can be obtained from Table 1.

Therefore, the problem to be solved in this study is to integrate the convolutional neural network into the collaborative filtering algorithm based on the rating matrix $R^{w\times t}$ and the comment matrix $C^{w\times t}$, Mining hidden features of tourists and tourist routes and predicting tourists' ratings for unrated routes.

Table 1: Rating and comment data.

User	Item			
	Item_1	Item_2	...	Item_w
User_1	r_{00}, c_{00}	?	?	?
User_2	?	r_{22}, c_{22}	?	?
...	?	?	r_{xy}, c_{xy}	?
User_w	?	?	?	?

2.2. Collaborative Filtering Model Design. The collaborative filtering algorithm model that integrates the convolutional neural network preprocesses the data, including removing invalid data and retaining comment and rating data. Then, each travel itinerary merges all its review data separately for each tourist and average the rating data. Then, the combined review data is converted into a vector-based on the text vectorization technology. The vectorized tourist and tourism data are sent to the convolutional neural network for training. Based on the convolutional neural network, the latent features of all comments of the tourist user x are extracted as the latent feature representation p_x of the tourist. Similarly, extract a particular travel route item, the implicit features of all reviews, as the feature representation q_y of the item. Then, based on the matrix decomposition method in collaborative filtering, the inner product of the transpose of p_x and q_y is performed to obtain the predicted score $r'_{xy} = p_x q_x^T$. Finally, the Adam optimization method is used to train the model to minimize the error between the predicted score r'_{xy} and the real score r_{xy}.

The model structure is shown in Figure 1, including the input layer, convolution layer, pooling layer, fully connected layer, and output layer. The input layer receives the vectorized review text as the input feature x, and the average score as the input feature y. The convolution layer performs convolution calculation on the input features by sliding the convolution kernel and obtaining the feature map. The pooling layer uses the maximum pooling method to reduce the dimension of the feature map calculated by the convolution layer and retains the more essential features in the network. The fully connected layer maps the learned latent features to the sample space then passes to the output layer to generate the output.

The model is divided into upper, middle, and lower parts (see Figure 1): tourist submodel, matrix decomposition model, and tourist route submodel, respectively. The input data format of the tourist sub-sub-model is $(vec_x^{user} \bar{r}_x^{\text{item}})$, which is regarded as a multiclassification problem. A 5-class prediction is made on the review data vector of a tourist. The category is $\bar{r}_x^{\text{user}}$ and is extracted from the fully connected layer during the training process as the latent feature r_x. The input data format of the travel route submodel is $(vec_x^{\text{item}}, (\bar{r}_x^{\text{item}})$ which is also regarded as a multiclassification problem. That is, the comment vec y, of an item is used for 5-category prediction. The category is $\bar{r}_x^{\text{item}}$, and during the training process, the hidden feature q_x of the tourist route is

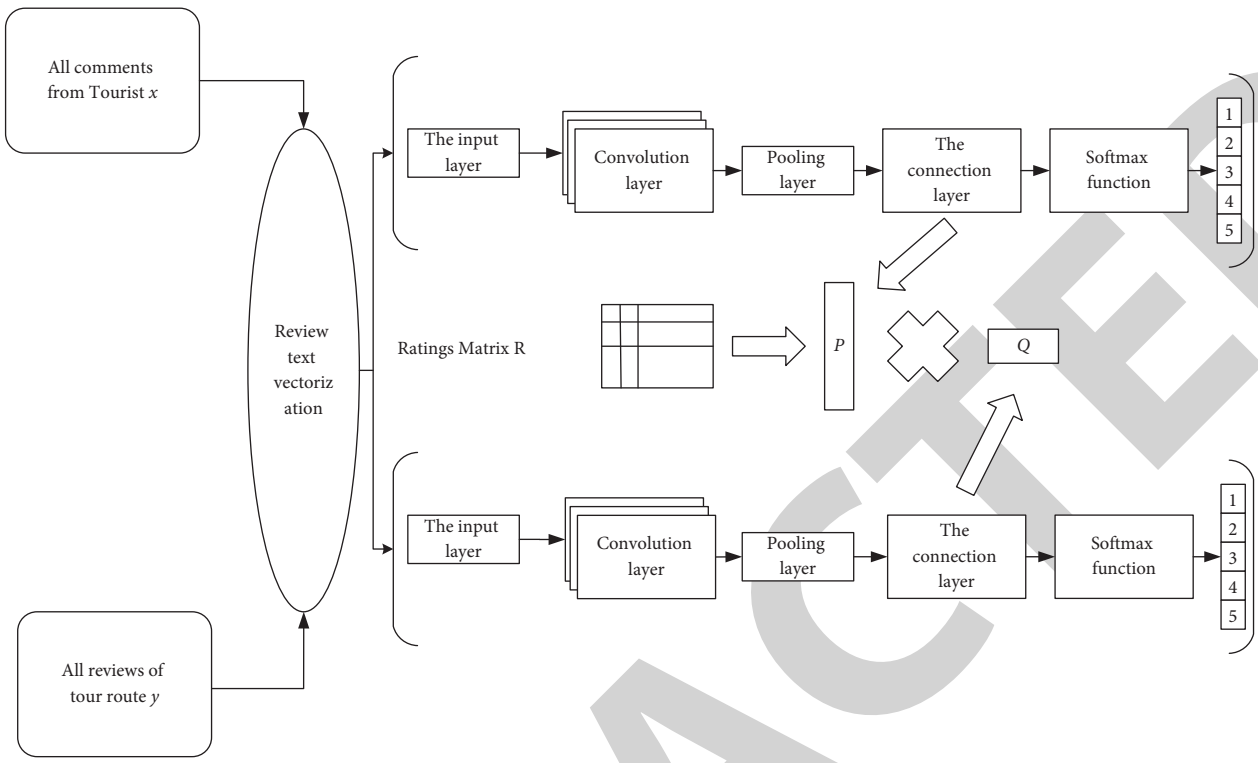

Figure 1: Structure of LSTM network.

extracted from the fully connected layer. The middle part is the matrix decomposition submodel, which takes the latent features p_x of tourists learned in the submodel of tourists and the latent features q_x of items learned in the submodel of tourist routes as inner products. Get the predicted score $r'_{xy} = p_x q_y^T$. Based on the Adam optimization method training, the whole model and the submodel minimize tourists. The prediction error of the travel route submodel and matrix factorization model calculates the error between the predicted score r_{xy}' and the real score r_{xy}.

Score prediction is realized based on nonnegative matrix factorization (NMF). The matrix $R \in R_+^{w\times t}$ is decomposed into two submatrices $W \in R_+^{w\times z}$ and $T \in R_+^{z\times t}$ by NMF. All $R, w,$ and t, terms are on negative terms because the following relationship can be derived: $R \approx \mathrm{W} \times \mathrm{T}$. In the context of recommender systems, $R_+^{w\times t}$ represents an $m * t$ nonnegative matrix. W can be understood as a tourist route. T is the latent factor. Z is the number of nondecomposable factors of R. In real recommendation problems, there are no negative terms (travel routes, tourists, ratings). The nonnegative constraints on W and T improve the interpretability of the collaborative filtering system.

The NMF is initialized based on the tourist route latent factor. The goal of the model is to find two lower-ranked nonnegative matrices $W_{Iq} \in R_+^{w\times z}$ and $T \in R_+^{z\times t}$ to predict ratings, calculated as $R = W_{Iq}T$. Set up a constraint between R and $W_{Iq}N$ as the objective function.

$$\left\|R - W_{Iq}T\right\|^2 = \sum_{px}\left(R_{px} - \left(W_{Iq}T\right)_{px}\right)^2. \tag{1}$$

In the formula, W_{Iq} represents the latent factor matrix initialized based on the contextual features of the tourist route.

Overfitting may occur in the prediction process, so the F-norm regular term is added to the objective function, and the objective function is rewritten as

$$\begin{gathered}\frac{1}{2}\rho_w \sum_t^w \left(W_{Iq}\right)_{p2} + \frac{1}{2}\rho_t \sum_t^T T_{x2}, \\ R - W_{Iq}T^2 = \sum_{px}\left(R_{px} - \left(W_{Iq}T\right)_{px}\right)^2 +,\end{gathered} \tag{2}$$

where ρ_w is the regular term of feature vector $(W_{Iq})_p$ and ρ_t is the regular term of T_x.

The eigenmatrices W_{Iq} and T are obtained by minimizing the objective function. The mathematical model is defined as

$$(W, T) = \min\left\|R - W_{Iq}T\right\|^2, \tag{3}$$

where $W > 0$ and $T > 0$.

Initialize the matrix T to a random matrix of size $z \times t$: $T = \mathrm{random}^{z\times t}$, where z is the minimum factor of the rating matrix. Calculate the minimum value of the objective function by iteratively updating W and T. The mathematical formula of iteration is

$$W_{pz} \leftarrow W_{pz}\frac{\left(T^N R\right)_{pz}}{\left(T^N T W_{Iq}\right)_{pz}},\ T_{zx} \leftarrow T_{zx}\frac{\left(W_{Iq}^N R\right)_{zx}}{\left(W_{Iq}^N W_{Iq} T\right)_{zx}}. \tag{4}$$

Inputs: the scoring matrix R of the dataset, the maximum number of iterations Maxiter, the threshold.
Output: feature matrices W and T.
Preprocess auxiliary information of tourist routes to obtain sentence matrix.
Use CNN to extract the feature vector of the sentence matrix.
Initialization: $T = \text{random}_{z\times t}, W = I_q = W_{Iq}$.
for z from 1 to Maxiter do /* Minimize the objective function $\min\|R - WT\|^2, W \geq 0, T \geq 0*/$
for p from 1 to W do
$W_{pz} \leftarrow W_{pz}(T^N R)_{pz}/(T^N T W_{Iq})_{pz}$.
endfor
for x from 1 to T do
$T_{zx} \leftarrow T_{zx}(W_{Iq}^N R)_{pz}/(W_{Iq}^N W_{Iq} T)_{pz}$.
endfor
Rerr = $RMSE(R - WT)^2$// Calculate the objective function
if Rerr < threshold
Break
Endif.
Endfor.
Return.

Algorithm 1: Recommendation algorithm based on nonnegative maximal matrix.

The predicted rating of tourist p for tourist route x can be calculated as

$$R'_{px} = W_p T_x. \tag{5}$$

In the formula, R'_{px} is the predicted score.

Algorithm 1 summarizes the algorithm of the recommendation system in this study. The algorithm iteratively updates the two latent factor matrices W and T until convergence is achieved.

2.2.1. Text Vectorization. Text vectorization uses word embedding to map text to w-dimensional vectors. First, for each tourist, all the review data are merged separately for each travel itinerary and average their scores. The raw data is processed into visitor review rating data and item review rating data. The user data format is as follows.

$(\text{user}_x, c_{x0} \oplus c_{x1} \oplus \cdots \oplus c_{xt}, 1/t \sum_{j=1}^{t} r_{xy}$, the travel route data format is: $\text{item}_x, c_{0y} \oplus c_{1y} \oplus \cdots \oplus c_{wy}, 1/w \sum_{x-1}^{w} r_{xy})$, where $\oplus$ means that the text is concatenated with spaces. Arrange the above formula as formulas (6) and (7).

$$\left(\text{user}_x, d_x^{\text{user}}, \bar{r}_x^{\text{user}}\right), \tag{6}$$

$$\left(\text{item}_y, d_y^{\text{item}}, \bar{r}_y^{\text{item}}\right), \tag{7}$$

where d_x^{user} represents all the reviews of tourist user_x on $\text{item}_1 \sim \text{item}_t$ and $\bar{r}_x^{\text{user}}$ represents the mean of all ratings of tourist user_x on $\text{item}_1 \sim \text{item}_t$. Then, based on the text vectorization technique, the merged review data is represented by a vector as vec_x^{user}. It is defined as formulas (8) and (9).

$$vec_y^{\text{user}} = Doc2VecC\left(d_x^{\text{user}}\right), \tag{8}$$

$$vec_y^{\text{item}} = Doc2VecC\left(d_y^{\text{item}}\right), \tag{9}$$

where the Doc2VecC function returns a w-dimensional vector, which represents each document d_x^{user} as a simple average of word embeddings, and ensure that the generated vector representation can capture the semantic information of the document during training learning. After text vectorization, the sample data of tourist x can be expressed as

$$\left(\text{user}_x, vec_x^{\text{user}}, \bar{r}_x^{\text{user}}\right). \tag{10}$$

The data of travel route y is represented as

$$\left(\text{user}_x, vec_x^{\text{user}}, \bar{r}_x^{\text{user}}\right). \tag{11}$$

2.2.2. Convolutional Layer. The convolutional layer is the core of the convolutional neural network, which is used to perform convolution calculations on the input sample features and extract important features. In the model in this study, the convolution layer receives the text vectors vec_x^{user} and vec_x^{item} of user $user_x$ to tourist route $item_y$ as input features. Then perform operation reshape on it and convert it to the size of $k * k$, where $k * k = m$. Suppose there are t neurons in the convolution layer, and each neuron n passes through the sliding convolution kernel F_n. Perform convolution calculations on the input samples and obtain new feature maps. The convolution kernel is also called a filter. The convolution computation in each convolution kernel slides across the width and height of the input data. Calculate the inner product of the entire convolution kernel and the input data anywhere. Assuming that the size of the convolution kernel is s, that is, $F_n \in R^{s\times s}$, the nth feature map can be calculated by the following formula:

$$w_n = f\left(vec_x^{\text{user}} * F_n + h_x\right). \tag{12}$$

Among them, $*$ represents the convolution calculation, h_x represents the bias term, which is a real number, and f is a nonlinear activation function. It can introduce nonlinear

factors into the model and solve the problem that linear models are difficult to represent as features. Commonly used activation functions are Sigmoid function, Tanh function, and ReLU function. The calculation formulas of the three activation functions are as follows:

$$\begin{aligned} \mathrm{Re}LU(i) &= \max(0, i), \\ Tanh(i) &= \frac{e^{i} - e^{-i}}{e^{i} + e^{-i}}, \\ \mathrm{Sigmoid}(i) &= \frac{1}{1 + e^{-i}}. \end{aligned} \tag{13}$$

2.2.3. Pooling Layer. The pooling layer is mainly used to reduce the dimension of the feature map calculated by the convolution layer. Retain the more essential features in the network while controlling the "overfitting" phenomenon to a certain extent. Suppose the feature graph $W_n = \{w_1, w_2, \ldots, w_{k-s+1}\}$ is obtained in the n-th convolution layer. The maximum pooling takes the maximum value in W_n, and u_n represents the pooling result of the n-th convolutional layer, which is given by

$$u_n = \max(W_n) = \max\{w_1, w_2, \ldots, w_{k-s+1}\}. \tag{14}$$

2.2.4. Fully Connected and Output Layer. The fully connected layer receives the output of the pooling layer. Suppose there are w neurons in the fully connected layer. After the fully connected layer ReLU activation function, a fixed-size vector p_x is obtained, which is the implicit feature of the tourist user $_x$. Similarly, the implicit feature vector q_y of the tourist route item $_y$ can also be obtained. The calculation formula is as follows:

$$\begin{aligned} q_y &= \mathrm{Re}LU\left(m_{\mathrm{item}_y} P_{\mathrm{item}_y} + h_{\mathrm{item}_y}\right), \\ p_x &= \mathrm{Re}LU\left(m_{\mathrm{user}_x} P_{\mathrm{user}_x} + h_{\mathrm{user}_x}\right), \end{aligned} \tag{15}$$

where $p_x, q_y \in R^w, P_{user_x}$ and P_{item_y} represent the pooling layer outputs of the tourist and tourist route submodels, respectively. m_{user_x} and m_{item_y} are the weights of the fully connected layers of the tourist and tourist route submodels, respectively, and h_{user_x} and h_{item_y} are the corresponding biases.

2.2.5. Matrix Factorization. Matrix decomposition uses matrix multiplication to decompose the user-item rating matrix into two low-dimensional user feature matrices and item feature matrices. In the model-based collaborative filtering algorithm, the model of tourists and tourist routes is constructed. From historical data such as ratings of tourists and tourist routes, learn the feature matrix of users and the feature matrix of items. Then the two feature matrices are multiplied to get the predicted score matrix. Finally, fit the predicted score matrix and the actual score matrix.

This study constructs a tourist submodel and an item submodel based on the convolutional neural network. From the reviews posted by tourists and all the reviews received by the tourist routes, the latent features p_x of tourists and the latent features q_y of tourist routes is learned. Then calculate the inner product of the implicit feature matrix of tourists and the implicit feature matrix of the tourist route. The predicted score matrix $r'_{xy} \in R$ is obtained, and the calculation formula is as follows:

$$r'_{xy} = p_x q_y^N. \tag{16}$$

Fit the predicted score and the actual score by the Adam optimization method. For tourist routes with ratings, make their predicted ratings as close to the actual ratings as possible. From this, it can be assumed that an unrated tour will have an actual rating close to the predicted rating. The calculation formula is shown in the below formula.

$$r_{xy} \approx r'_{xy} = p_x q_y^N. \tag{17}$$

3. Experimental Results and Analysis

3.1. Dataset. The travel data comes from a travel company owned by an airline. It includes 25,714 travel records of 4,731 tourists on 1,439 tourist routes, and each tourist contains at least three routes that they have participated in. Each record includes tour group number, tourist name, gender, ID number, line departure time, line price, and detailed introduction to attractions. It is an implicit feedback dataset. As long as a tourist has participated in a certain route, it is considered that the tourist likes this route. The dataset is split into a training set (train), a validation set (validation), and a test set (test) in a ratio of 7 : 2 : 1. The training set is used to train the model, the validation set is used for parameter selection, and the test set is used to evaluate the model.

3.2. Evaluation Indicators. This study uses Precision, Recall, F1 value, and Normalized Discounted Cumulative Gain (NDCG) as the algorithm evaluation criteria.

Precision: The recommendation accuracy refers to the ratio of the number of travel routes that tourists like to the total number of recommended travel routes among the recommended travel routes.

$$\mathrm{Precision} = \frac{\sum_{p \in P} |R(p) \cap N(p)|}{\sum_{p \in P} |R(p)|}. \tag{18}$$

In the formula, $R(p)$ represents the list of recommended items for tourist p according to the behavior of tourist p on the training set, and $N(p)$ represents the list of items that tourist p likes on the test set.

Recall rate (Recall): The recall rate of the recommendation system represents the ratio between the number of travel routes that tourists like among the recommended travel routes and the number of travel routes that tourists like. The calculation process of the recall rate is as follows:

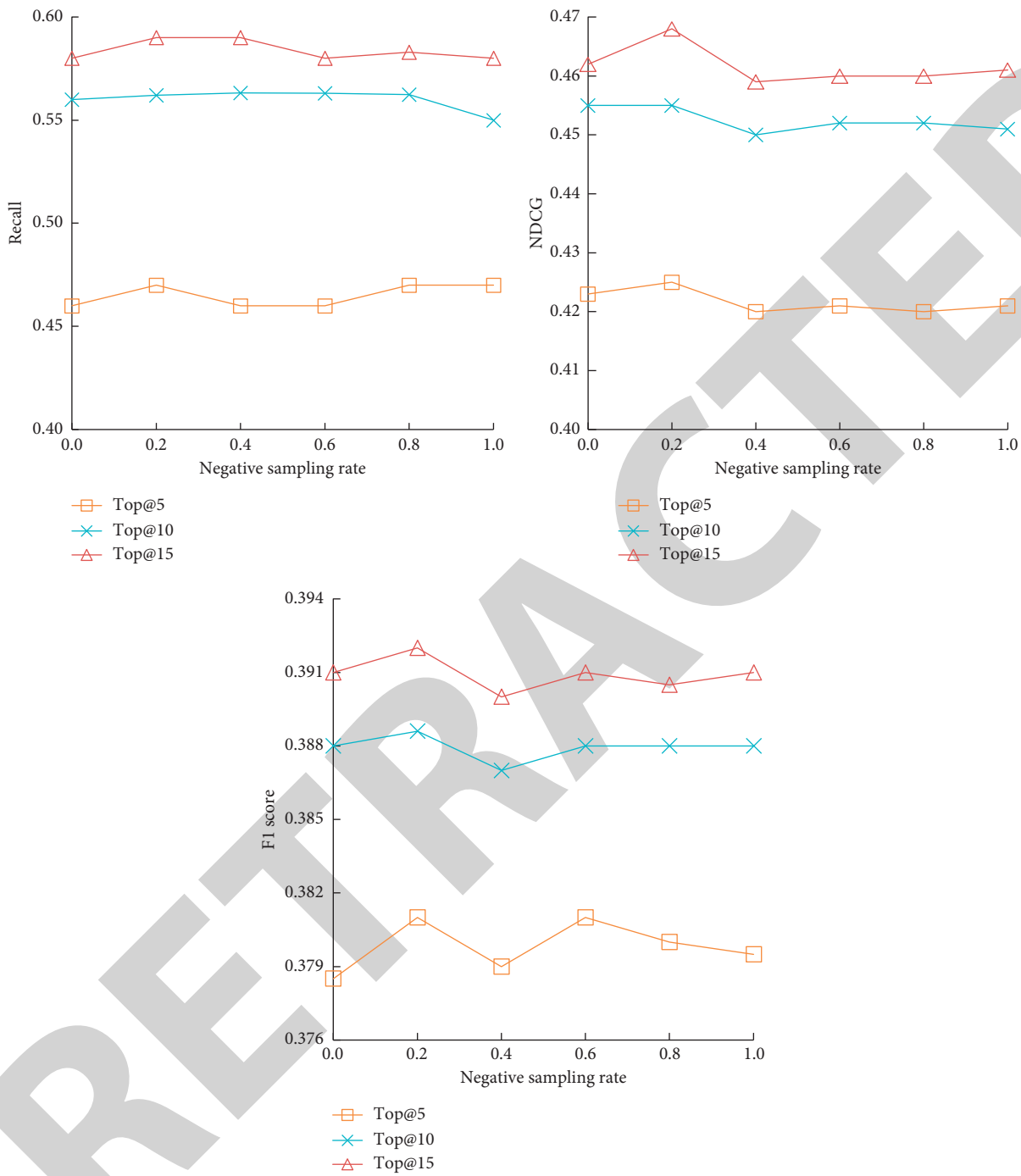

Figure 2: Effect of negative sampling rate.

$$\text{Recall} = \frac{\sum_{p \in P} |R(p) \cap N(p)|}{\sum_{p \in P} |N(p)|}. \tag{19}$$

F1 score: It is a comprehensive index of precision and recall. The larger the value of F1, the better the algorithm's performance. The F1 value is defined as follows:

$$F1 = \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \tag{20}$$

NDCG@K is used to sort the predicted probability value of each travel route in the recommended list. If the predicted probability value of the top travel route in the recommended list is more significant, the greater the value of NDCG, the better the recommendation effect of the model.

$$NDCG@K = \frac{1}{|P|} \sum_{p \in P} NDCG_p@K, \tag{21}$$

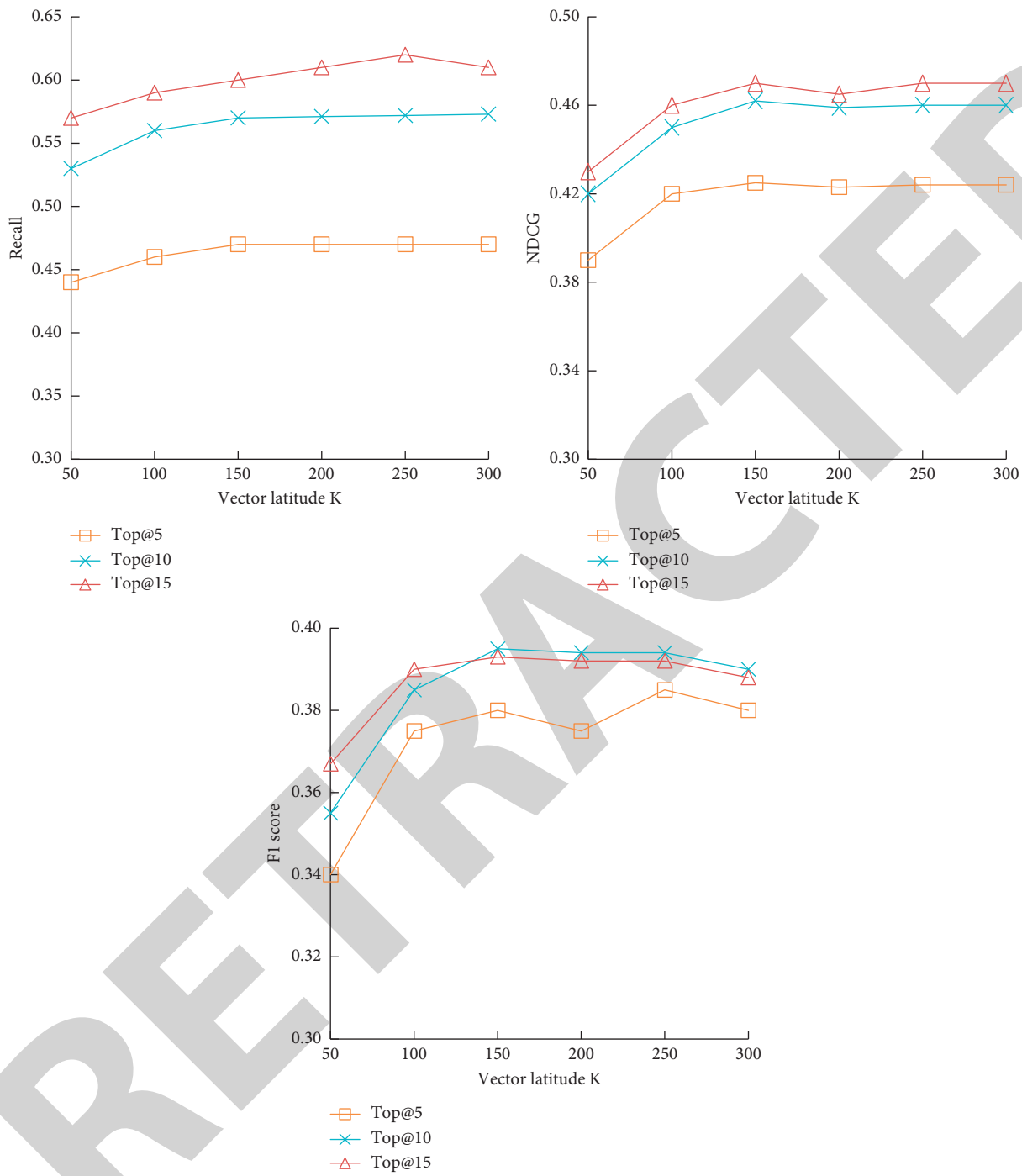

Figure 3: Effect of dimension size.

where P represents the number of tourists in the test set and $\sum_{p \in P} TDCA_p@K$ is the sum of the NDCG values of each tourist.

3.3. Parameter Training

3.3.1. Effects of Negative Sampling Rates. The negative sampling rate indicates how many disliked items are sampled for each traveler. At a fixed negative sampling rate, the number of disliked lines sampled for each tourist is different. The larger the negative sampling rate, the more unfavorable routes are sampled for the same tourist. There are many negative samples in the optimization process. If the value of the negative sampling rate is smaller, the number of lines that are not liked is extracted, and there will be fewer negative samples. Therefore, it is discussed in the experiment how many negative samples are drawn for each tourist, making the model perform the best. The grid search method is used in the experiment, and the negative sampling rate is

Table 2: Comparison of the proposed with other methods.

Method	Recall			NDCG			F1 score		
	Top@5	Top@10	Top@15	Top@5	Top@10	Top@15	Top@5	Top@10	Top@15
Proposed	0.508	0.564	0.596	0.469	0.497	0.506	0.405	0.418	0.419
Literature [19]	0.491	0.545	0.584	0.452	0.483	0.493	0.385	0.407	0.405
Literature [20]	0.472	0.531	0.582	0.446	0.465	0.475	0.382	0.394	0.407
Literature [21]	0.467	0.526	0.551	0.425	0.446	0.457	0.376	0.383	0.389

selected according to the algorithm's performance. With fixed regularization parameters and latent factors, the search range for negative sampling rates is {0.0, 0.2, 0.4, 0.6, 0.8, 1.0}, and the performance results are shown in Figure 2.

It can be seen from Figure 2 that as increases, the three evaluation indicators of Top@5, Top@10, and Top@15 first increase and then decrease, later tend to remain unchanged. The results confirm that the model performance first increases and then decreases as the number of negative samples increases. The larger the number of negative samples, the slower the model's training will be. Therefore, to balance model performance and efficiency, the negative sampling rate is taken as 0.2.

3.3.2. Influence of Vector Dimension. The vector dimension represents the length of the latent features of tourists and travel itineraries. The larger the vector dimension, the richer the information representation of tourists and tourist routes, but the more significant computation. The smaller the vector dimension, the less accurate, but the faster the learning speed. Therefore, in the case of balancing accuracy and calculation amount, a more appropriate dimension size is generally set through experiments. Set the vector dimension search range to {0, 20, 40, 60, 80, 100}, and the experimental results are shown in Figure 3. It can be seen from Figure 3 that with the increase of vector dimension, the NDCG and F1 indicators proliferate. When the vector dimension reaches more than 40, the growth rate slows down, and the algorithm performance tends to be stable. The growth of the Recall indicator is not as dramatic as that of NDCG and F1, but it still increases first and then decreases. After reaching 40, the growth rate slows down, so the vector dimension is taken as 40.

3.4. Experimental Results and Comparative Analysis. To further illustrate the effect and prediction performance of the algorithm in this study, the algorithm in this study is compared with other traditional recommendation algorithms. The performance comparisons of the four algorithms in Recall@K, NDCG@K, and F1 are shown in Table 2.

It can be seen from Table 1 that as N increases, Recall@K, NDCG@K, and F1@K all increase. Among them, the growth rate of Recall@K is particularly obvious. Such results validate the fact that in reality, the more routes are offered to tourists, and the more routes will be chosen to satisfy tourists' interests. However, there are many recommended routes, and the routes that tourists are most interested in may not always be at the top of the recommended list. Based on the two methods of network representation learning, the results of our algorithm and literature [19] are better than that of literature [20] and literature [21]. This is because the network representation learning method is used to learn route and passenger vector representation, making better data information. A good uniform representation is obtained. It is difficult for literature [20] and literature [21] to fuse and fully utilize this information. The proposed is better than literature [19]. It shows that using the word vector method to add information about the co-occurrence of tourists, common liking of route pairs, and a shared dislike of route pairs can get a better vector representation of routes and passengers. However, literature [19] only utilizes the feature information of scenic spots to learn the vector representation. The performance of literature [20] is better than that of literature [21] because literature [20] makes good use of extra data information. Literature [21] only uses the interaction matrix of tourists and routes.

4. Conclusion

With the rapid development of smart tourism under the Internet, more and more people are accustomed to choosing travel routes on major travel websites. At present, there is a lot of recommended information about travel routes on travel websites, and all of them are mainly display information. Choosing a travel route that meets the interests of tourists from many travel information has become an urgent problem for the platform to solve. This study proposes a travel itinerary recommendation algorithm combining convolutional neural networks and collaborative filtering. The algorithm convolutional neural network consists of an input layer, a convolutional layer, a pooling layer, a fully connected layer, and an output layer. The algorithm first vectorizes the information about tourists and tourist routes and obtains the implicit feature map through volume and neural network. Then use the collaborative filtering algorithm to complete the line recommendation. This study uses the evaluation criteria of Recall, Normalized Discounted Cumulative Gain (NDCG), and F1 score. The experimental results show that the proposed algorithm has higher recommendation accuracy and recall. At the same time, the search accuracy and convergence speed of the optimal solution are significantly improved. At present, users will choose a variety of transportation modes during travel. Urban traffic conditions are complex, and different modes of transportation have a great impact on the travel experience. In the following work, the influence of travel mode on travel route recommendation will be considered. How to apply

travel mode to tourism route recommendation is a problem that we need to focus on and solve.

Data Availability

The labelled dataset used to support the findings of this study is available from the corresponding author upon request.

Conflicts of Interest

The authors declare no competing interests.

Acknowledgments

This work was supported by the Henan Institute of Economics and Trade.

References

[1] S. C. Brown and D. Knox, "Why go to pop concerts? The motivations behind live music attendance," *Musicae Scientiae*, vol. 21, no. 3, pp. 233–249, 2017.

[2] S. Song and M. Yoo, "The role of social media during the pre-purchasing stage," *Journal of Hospitality and Tourism Technology*, vol. 7, 2016.

[3] J. W. Morris, "Music platforms and the optimization of culture," *Social Media + Society*, vol. 6, no. 3, Article ID 205630512094069, 2020.

[4] J. P. Simon, "New players in the music industry: lifeboats or killer whales? The role of streaming platforms," *Digital Policy, Regulation and Governance*, vol. 21, 2019.

[5] X. Xie, X. Xie, and C. Martínez-Climent, "Identifying the factors determining the entrepreneurial ecosystem of internet cultural industries in emerging economies," *International Entrepreneurship and Management Journal*, vol. 15, no. 2, pp. 503–522, 2019.

[6] C.-H. Hsiao, J.-J. Chang, and K.-Y. Tang, "Exploring the influential factors in continuance usage of mobile social Apps: s," *Telematics and Informatics*, vol. 33, no. 2, pp. 342–355, 2016.

[7] I. Kamehkhosh, G. Bonnin, and D. Jannach, "Effects of recommendations on the playlist creation behavior of users," *User Modeling and User-Adapted Interaction*, vol. 30, no. 2, pp. 285–322, 2020.

[8] N. Askin and M. Mauskapf, "What makes popular culture popular? Product features and optimal differentiation in music," *American Sociological Review*, vol. 82, no. 5, pp. 910–944, 2017.

[9] K. Volchek, A. Liu, H. Song, and D. Buhalis, "Forecasting tourist arrivals at attractions: search engine empowered methodologies," *Tourism Economics*, vol. 25, no. 3, pp. 425–447, 2019.

[10] R. Villegas, J. Yang, and Y. Zou, "Learning to Generate Long-Term Future via Hierarchical prediction," in *Proceedings of the international Conference on Machine Learning. PMLR*, pp. 3560–3569, Sydney, Australia, August, 2017.

[11] L. Yan, H.-T. Zhang, J. Goncalves, and Y. Xiao, "An interpretable mortality prediction model for COVID-19 patients," *Nature Machine Intelligence*, vol. 2, no. 5, pp. 283–288, 2020.

[12] Z. Wang and R. S. Srinivasan, "A review of artificial intelligence based building energy use prediction: c," *Renewable and Sustainable Energy Reviews*, vol. 75, pp. 796–808, 2017.

[13] K. C. Green and J. S. Armstrong, "Simple versus complex forecasting: the evidence," *Journal of Business Research*, vol. 68, no. 8, pp. 1678–1685, 2015.

[14] N. S. Arunraj and D. Ahrens, "A hybrid seasonal autoregressive integrated moving average and quantile regression for daily food sales forecasting," *International Journal of Production Economics*, vol. 170, pp. 321–335, 2015.

[15] S. Smyl, "A hybrid method of exponential smoothing and recurrent neural networks for time series forecasting," *International Journal of Forecasting*, vol. 36, no. 1, pp. 75–85, 2020.

[16] Q. Wen, J. Gao, X. Song, and L. H. S. Sun, "RobustSTL: a robust seasonal-trend decomposition algorithm for long time series," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 33, no. 01, pp. 5409–5416, 2019.

[17] X.-Y. Zhang, F. Yin, Y.-M. Zhang, and C.-L. Y. Liu, "Drawing and recognizing Chinese characters with recurrent neural network," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 40, no. 4, pp. 849–862, 2018.

[18] Z. Huang, W. Xu, and K. Yu, "Bidirectional LSTM-CRF models for sequence tagging," 2015.

[19] L. Hu, C. Li, C. Shi, and C. C. Yang, "Graph neural news recommendation with long-term and short-term interest modeling," *Information Processing & Management*, vol. 57, no. 2, Article ID 102142, 2020.

[20] T. Bai, J. R. Wen, and J. Zhang, "A neural collaborative filtering model with interaction-based neighborhood," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*, pp. 1979–1982, Singapore, November, 2017.

[21] D. Liang, J. Altosaar, and L. Charlin, "Factorization meets the item embedding: regularizing matrix factorization with item co-occurrence," in *Proceedings of the 10th ACM conference on recommender systems*, pp. 59–66, Boston, USA, September, 2016.