

Performance of Combined Recommendation Modeling Fused With Tagawareness In Mixed Traffic Scenarios

Xiangyuan Li

Nanjing University of Posts and Telecommunications

Fei Ding (✉ dingfei@njupt.edu.cn)

Nanjing University of Posts and Telecommunications <https://orcid.org/0000-0002-8935-8169>

Suju Ren

Nanjing University of Posts and Telecommunications

Jianmin Bao

Nanjing University of Posts and Telecommunications

Ruoyu Su

Nanjing University of Posts and Telecommunications

En Tong

Posts and Telecommunications Institute of Technology

Research Article

Keywords: mixed traffic, intelligent connected vehicles, recommended system, collaborative filtering algorithm, tag-aware

Posted Date: December 20th, 2021

DOI: <https://doi.org/10.21203/rs.3.rs-1120711/v1>

License: © ⓘ This work is licensed under a Creative Commons Attribution 4.0 International License.

[Read Full License](#)

Performance of combined recommendation modeling fused with tag-awareness in mixed traffic scenarios

Xiangyuan Li^{1,2,3}, Fei Ding^{1,2,3*}, Suju Ren^{1,2}, Jianmin Bao^{1,2}, Ruoyu Su^{1,2} and En Tong³

Abstract

Due to the heterogeneous characteristics of vehicles and user terminals, information in mixed traffic scenarios can be interacted based on the Web protocol of different terminals. The recommendation system can dig users' travel preferences by analyzing historical travel information of different traffic participants, to publish accurate travel information and services for the terminals of traffic participants. The diversification of existing road network users and networking modes, as well as the dynamic changes of user interest distribution caused by high-speed movement of vehicles, traditional collaborative filtering algorithms have limitations in terms of effectiveness. This paper proposes a novel Hybrid Tag-aware Recommender Model (HTRM). The model embedding layer first employs the Word2vec model to represent the tags and ratings of projects and users, respectively. The feature layer then introduces the auto-encoder to extract self-similar features of the item, and a long short-term memory (LSTM) network is used to extract user behavior characteristics to provide higher-quality recommendations. The gating layer combines the features of users and projects and then makes score recommendations based on the Fully Connected Neural Network (FCNN). Finally, Web data sets of different service preferences of traffic participants during the trip are used to evaluate the model recommendation performance in different scenarios. The experimental results show that the HTRM model is reasonable in design and can achieve high recommendation accuracy.

Keywords: mixed traffic; intelligent connected vehicles; recommended system; collaborative filtering algorithm; tag-aware

Introduction

With the application and development of Internet communication technology, the public's demand for intelligent transportation information has increased sharply. Various travel information service platform models have emerged to meet this demand, such as city information query, map point of interest search, dynamic route guidance, etc. Travel service decision-making is divided into external and internal factors. External factors mainly refer to travel expenses and weather conditions, and internal factors mainly refer to a traveler's personal travel habits and experiences. The service content of the existing travel service platform is too simple, and the form is relatively single. Current service requirements of most traffic information service systems for travelers¹ only ask travelers to enter queries manually. The development of the Internet of Vehicles and data mining can enhance traffic services by considering travelers as the service objects to conduct the analysis,

modeling, and semantics of travelers' behavior and preferences. Personalized service requirements can then be recommended, and information will be pushed before or during the trip to provide real-time traffic information services for travelers, thereby enhancing the public travel experience and convenience [1-2].

In recent years, the Intelligent Connected Transportation System (ICTS) has emerged as the evolutive direction of new infrastructure for road traffic and integrated control of vehicles, roads, and the cloud. Intelligent Connected Vehicles (ICVs), Recreational Vehicles (RVs), and various traffic participants will continue to share road resources for a long time in the future due to the limitations of system construction and development. These road users interact with each other via Web services and provide collaborative services [3]. ICVs employ Cellular Vehicle-to-Everything (C-V2X) technology to build in-vehicle networks. RVs and their users rely on in-vehicle Internet services and the mobile terminal platform to join the above-mentioned in-vehicle network and obtain services. Due to the differences in service objects and methods, as well as the imbalance between the supply and demand of traffic information resources, the demand for personalized traffic information services by travelers has become increasingly critical. Therefore, our recommendation algorithm has significant application potential in the ICTS field. Through the accurate

*Correspondence: dingfei@njupt.edu.cn

¹Jiangsu Key Laboratory of Broadband Wireless Communication and Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

²School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China

³China Mobile-Nanjing University of Posts and Telecommunications 5G Joint Innovation Center, Nanjing 210003, China

Full list of author information is available at the end of the article

transmission of information, the ICTS can not only optimize the performance of heterogeneous in-vehicle networks but also improve the efficiency of the system's precise services and provide personalized transportation services.

As a typical method in the Web recommendation system, the collaborative filtering algorithm uses the similarity between users and items to predict the items that current users are potentially interested in [4]. Collaborative filtering algorithms are divided into user-based and item-based models [5]. The user-based collaborative filtering algorithm uses the interest of nearest neighbor users to predict the current user's interest in the same item, while the item-based collaborative filtering algorithm calculates the inter-item of similarity to recommend to users. However, as traditional collaborative filtering algorithms focus on constructing a user-scoring matrix, they are prone to data sparseness and cold start problems when facing new businesses, content, and services, which affects the quality of recommendations [6-8].

In a mixed transportation system, ICVs have the ability to perceive the surrounding environment and communicate with other vehicles of the same type. There are also temporal and spatial correlations between Web services in different business scenarios. Therefore, it is necessary to recognize service types and achieve high-quality recommendations through tag perception. J. Ye et al. [9] extracted the business tags that users most frequently employed and designed a recommendation method across resources based on the correlation between tag data in different fields. The Density-Based Spatial Clustering of Applications with Noise (DBSCAN) algorithm was then used for service recommendation based on tag vector clustering. A secondary recommendation strategy was also proposed based on the Term Frequency-inverse Document Frequency (TF-IDF) algorithm to achieve more personalized resource recommendations. A. Shepitsen et al. [10] aggregated redundant tags through clustering, making redundant tags easier to detect. Zhu Zhicheng et al. [11] regarded different attributes of the service as multiple tags, generated user portraits based on this, and proposed a recommendation method based on the Hidden Markov Model (HMM), using multi-tag data to improve recommendation performance. Wen Junhao et al. [12] employed a user-label matrix and a resource-label matrix to propose a collaborative filtering recommendation method based on tag preference probability to implement an algorithm based on user preference prediction. This research

overcomes the semantics existing in traditional tag-based recommendation algorithms. The ambiguity problem improves the accuracy of the recommendation result. However, due to the limitations of the user-tag two-dimensional matrix, although it can recommend users with high satisfaction services, the recommendation results obtained by such methods are usually not diverse. Ma Wenkai et al. [13] used tag data to propose a recommendation method based on the Markov chain model. This method disassembled the user-tag-service three-dimensional relationship into user-tag and tag-service relationships and used the Markov chain model to calculate the matching degree between the user and the service. Jiang Shuhao et al. [14] designed a personalized recommendation method that balanced optimization accuracy and diversity in response to differences in personal preferences. They combined user historical preferences and item ratings to generate a recommendation list and finally matched corresponding services based on user diversity. Meng Xiangfu et al. [15] constructed a geo-social relationship model for the similarity evaluation of interest points and introduced spectral clustering to divide the diversity of interest points. They then selected and personalized the interest points based on the probability factor model to obtain points of interest recommendation results that met the preferences of different users. Although the above-mentioned recommendation algorithms have achieved many beneficial results, most ignore the internal connection between user preferences and time and cannot provide users with timely and accurate information over time. Z. Cui et al. [16] proposed a recommendation model based on time correlation coefficients, clustering similar users based on the K-means algorithm and providing higher-quality recommendations by analyzing user behaviors. Y. Hu et al. [17] integrated the time label into similarity measurement and Quality of Service (QoS) prediction, which alleviated the problem of data sparseness to a certain extent, and met the service demand of personalized recommendation that changes over time. Qian Wenyi et al. [18] proposed a taxi pick-up point recommendation algorithm based on spatio-temporal context collaborative filtering, which mapped the pick-up point information to a spatial grid and calculated the similar passenger behavior neighbors assemble by introducing a time attenuation factor. Thus, the best pick-up point recommendation for the target taxi was realized.

Research into personalized recommendations represented by the neural network has made significant progress in recent years. This type of method uses the vector expression of the associated data of Web services, converts tags into sparse feature vectors, and employs neural networks to extract potential features. Zuo et al. [19] used a stacked sparse autoencoder to obtain an abstract representation of a tag-based user profile and combined it with user-based collaborative filtering for recommendations. Xu et al. [20] proposed the deep-semantic similarity based personalized recommendation with negative sampling (DSPR-NS) model, which used Multilayer Perceptron (MLP) to map user and service tags to the feature vector space, maximizing the semantic similarity between users and related services. In the above two methods, tags are represented by the Bag of Words (BOW) model. Due to the limitation of BOW, the problem of label redundancy and ambiguity cannot be solved. At the same time, the model ignores rich label semantic information in the initial state. Xu et al. [21] proposed an improved hybrid deep learning-based personalized recommendation with negative sampling (HDLPR-NS) model based on DSPR-NS and introduced an auto-encoder (AE) to further accelerate the efficiency of model self-learning. Huang et al. [22] used the Route-Path approximation method to generate an approximate optimal solution for time-dependent vehicle routing problem with path flexibility (TDVRP-PF) under random traffic conditions and combined the path selection and time selection factors in the road network to ensure the real-time performance of the recommendation. With the rapid development of Natural Language Processing (NLP), the industry has attempted to associate tag information with NLP to obtain richer semantic information [23]. For the scoring prediction model, Liang et al. [24] used deep neural networks and recurrent neural networks to extract the potential features of users and services and employed forward neural networks to simulate matrix decomposition to evaluate the feasibility of comprehensive prediction scoring by introducing vectors from different feature spaces. The above method only uses the label information of the user and the service and ignores the scoring information. However, for users who frequently use service ratings, the rating information can effectively mine user preferences. Tag recommendation predicts the relevant tags of the target user based on tags that other users have provided for the same user. Jiang et al. [25] analyzed taxi driver's preference information, such as

passenger seeking and passenger-carrying strategies, and explored driver preference by constructing a service strategy. Ren Xingyi et al. [26] quantified the user's preference for candidate regions from the perspectives of geography, interest, and society and combined the probability matrix decomposition model to make regional recommendations for users.

Road network users and networking modes in a mixed transportation system are diverse. The high-speed movement of vehicles creates dynamic changes in the distribution of user interests, which restricts the traditional collaborative filtering algorithm. Fusion label data can efficiently identify services and solve the problem of data sparseness faced by recommendation algorithms. Service information can then be accurately released to the traffic data processing platform via combined modeling of the service needs and preferences of connected vehicles and travelers. The traditional recommendation system ignores the inherent connection between user preferences and time and introduces the recommendation modeling of user behavior tag data, which is conducive to providing users with timely and accurate information.

Based on the application characteristics of intelligent networked mixed traffic scenarios, we propose a Hybrid Tag-aware Recommender Model (HTRM), which uses word embedding models to express the tags and scores of users (traffic travelers) and projects (travel services), respectively. The complex processing of text features is simplified to vector calculation. An autoencoder is used to extract text features of item tags, and Long Short-Term Memory (LSTM) loop mechanism and gating mechanism are used to extract accurate features of user behavior tags. The Fully Connected Neural Network (FCNN) is then employed to score and predict the fusion of user and item features. Finally, the performance of the proposed model and recommendation algorithm is verified by using open Web data sets of different service preferences in the traffic travel process.

The contributions of this article are summarized as follows:

1. The combined modeling of heterogeneous data, such as tags and ratings of users and items and user behaviors, makes full use of the integrated feature vectors after fusion to better characterize users and items and overcome the cold start problem;
2. Word2vec is employed to convert feature labels into feature vectors and express the semantic similarity between text features through vector distance;

3. The proposed recommendation algorithms are tested and compared on public data sets. The experimental results show that, compared with the traditional recommendation model, the model proposed in this paper has better recommendation performance.

The rest of this paper is organized as follows. We present the design and method of the HTRM model in “HTRM model and method” section. Experimentation and comparison results are provided in “Experimental results and analysis” section. Finally, we offer concluding remarks in “Conclusion” section.

HTRM Model and Method

The ICTS scenario includes Intelligent Connected Vehicles, Recreational Vehicles, and various traffic participants, which employ Web services based on different terminal platforms for information interaction and collaborative services. ICVs perceive the surrounding environment information through the PC5 link and conduct remote information interaction through the Uu interface. The RV in-vehicle terminal, intelligent terminals carried by pedestrians, and personal terminals for user-oriented services are all remotely connected to the ICTS through the Internet. The Vehicle to Cloud (V2C) interface encapsulates each Internet access to ensure the standardization of Web services and the openness of the platform.

In the mixed traffic scenario, we regard traffic participants as users of the recommendation model and their corresponding travel service types as items. Since the tags and ratings of users and items in the recommendation system, as well as user behaviors, are heterogeneous data, this paper proposes a combined recommendation modeling method that integrates tag perception, as shown in Figure 1. The presentation layer uses the Word2vec model to represent item tags, item ratings, user behavior tags, and user ratings. The feature extraction layer introduces an autoencoder to extract the self-similar features of the item and obtains user behavior features through the Long Short-Term Memory network to provide more high-quality information. The feature fusion layer uses the gating layer to learn joint features of items and user information. FCNN is used to predict services in the prediction layer, according to the final learning results of the past items and users. This works to provide traffic travelers with the appropriate personalized travel service recommendations, such as suggestions for in-vehicle services, travel destinations, and

entertainment activities.

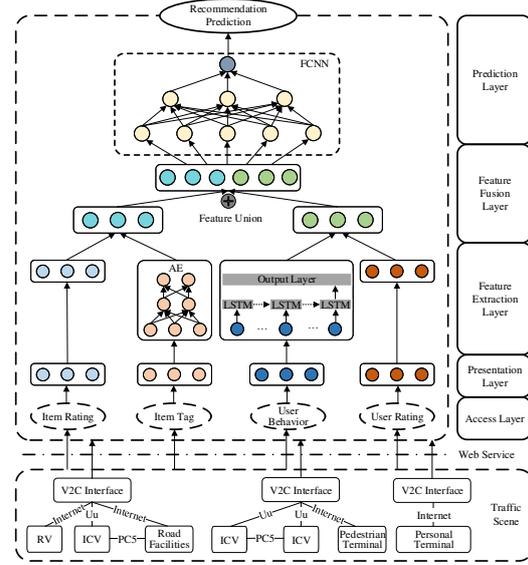


Figure 1. Recommender service framework in mixed traffic.

It is assumed that there are N users (traffic participants) and M items (travel service types) in the recommendation system under mixed traffic. The sparse scoring matrix is represented by $R \in R^{N \times M}$, where $R_{n,m}$ represents the rating of user n of the item m . In addition, u^R represents the user rating embedding learned from the user rating, and u^C represents the user text feature learned from the user tag behavior. In the same way, v^R and v^C respectively represent the item embedding and item text features learned from item scores and item tags.

Item tag perception model

The item tags in the recommendation system are derived from the text data of the user's historical services (item). Traditional service feature extraction usually employs the WordNet model [27] to calculate the similarity between the tag sets of different users. As the WordNet model employs a manually defined dictionary, the number of words is limited and cannot match scene changes. This paper introduces the Word2vec model [7], which maps words to K-dimensional distribution vectors to represent tags with word embeddings. The word embedding layer uses the Word2vec technology to convert the word sequence into a dense vector matrix, and the matrix weights are initialized randomly. The embedding of a word can be expressed as a table look-up process, which accelerates the convergence by using the word embedding table trained by the user in advance.

Word sequences can have different lengths, and their representation is usually encoded as a vector with the maximum length. For word sequences less than the maximum length, zeros are padded at the

beginning or the end of the sequence. Considering a word sequence $X = [x_1, x_2, \dots, x_m]$, where x_m represents the word sequence of the item m , the model looks up the embedding of each word in the sequence from the pre-trained word embedding table and then connects these dense word vectors to obtain the dense matrix, as shown in Equation (1).

$$X = \begin{bmatrix} \dots & x_{m-1} & x_m & x_{m+1} & \dots \\ | & | & | & | & \\ | & | & | & | & \\ | & | & | & | & \\ \dots & \dots & \dots & \dots & \dots \end{bmatrix} \quad (1)$$

In order to capture the semantic information present in item tags, HTRM uses pre-trained word embedding vectors to represent item information.

The output vector of the embedding layer is sent to the encoder for advanced feature extraction. Supposed that the input vector of the encoder is initialized as $x_i^{(1)}$, the output of the first hidden layer $x_i^{(2)}$ is expressed as:

$$x_i^{(2)} = \tanh(W_1 x_i^{(1)} + b_1) \quad (2)$$

where W_1 is the weight matrix, and b_1 is the bias vector.

The final output result of the item label, that is, the output of layer j , can be expressed as:

$$x_i^{(j+1)} = \tanh(W_j x_i^{(j)} + b_j) \quad (3)$$

where $x_i^{(j)}$ is the output of layer $j - 1$.

User behavior label perception mode

User behavior labels are used as the input layer of the Recurrent Neural Network (RNN), and the state vector of the hidden layer is used as the advanced features of user behavior labels. The HTRM model represents user behavior labels as a time series containing timestamps. Before importing the model, the latest L record data is sorted according to the tag timestamp, and each item sequence is coded through the embedding layer. The RNN model can dynamically read sequences of indefinite length, by cropping the longer sequence, and filling the shorter sequence with 0 to ensure the sequence length is the same. For example, for a given user $U_i = [u_i^1, u_i^2, \dots, u_i^L]$, the user can be represented by a matrix of size $L \times D$. Therefore, the characteristics of each user N can be represented by a three-dimensional array $N \times L \times D$.

In the process of using RNN for sequence prediction, the hidden layer is located between the input layer and the output layer to save historical information. In the HTRM model, the state of the hidden layer represents the high-level latent features

of the user's label behavior as \hat{u} . To avoid the problem of vanishing gradient caused by RNN, the LSTM network is used as the feature extraction network of user behavior labels.

Gating model

In order to make full use of the label and scoring information, it can be jointly represented by fusion methods, such as concatenation, dot product, and regularization. The HTRM model introduces two neural gating layers to fuse the ratings embeddings and text features of users and items respectively. Among them, the user gating layer integrates user rating embedding and user behavior label text features, and the item gating layer integrates item embedding ratings and item label text features.

The neural gating layer uses a matrix to replace the weight vector in the traditional gating layer. For a given user n , the neural gating layer can be expressed as:

$$G^{(u)} = \sigma(W_{g1}^U u_n + W_{g2}^U u_n + b_g^{(u)}) \quad (4)$$

where W_{g1}^U and W_{g2}^U are user gating layer parameters, $b_g^{(u)}$ is the bias, and σ is the activation function. In order to ensure $G^{(u)}$ is in the range of $[0,1]$, a sigmoid activation function is used.

Similarly, the neural gating layer for a given item m can be expressed as:

$$G^{(v)} = \sigma(W_{g1}^V u_m + W_{g2}^V u_m + b_g^{(v)}) \quad (5)$$

where W_{g1}^V and W_{g2}^V are the project gating layer parameters, and $b_g^{(v)}$ is the bias term. The joint representation of user and item vectors is:

$$u_n^f = G^{(u)} u_n^R + (1 - G^{(u)}) u_n^C \quad (6)$$

$$v_m^f = G^{(i)} v_m^R + (1 - G^{(i)}) v_m^C \quad (7)$$

Recommended prediction model

The HTRM model fully mines the deeper semantic information in the scoring and text features through the gating layer and provides accurate information for the subsequent scoring prediction based on the fully connected network, thereby improving the accuracy of the recommendation.

Supposed that the fusion result of user and item features output by the gating layer is expressed as:

$$x = [u_n^f, v_m^f] \quad (8)$$

A fully connected neural network is added as a prediction layer to improve the robustness and nonlinear fitting ability of the model. The network adopts a pyramid structure, and the number of

neurons decreases as the number of layers deepens. The output of the prediction layer can be expressed as:

$$F = \sigma_L(W_L(\sigma_{L-1}(W_{L-1}L(\sigma_1(W_1x+b_1)+b_{L-1}))+b)) \quad (9)$$

where L represents the number of hidden layers; σ_l , b_l , and W_l represent the activation function, bias term and parameter matrix of the layer l , respectively.

The prediction result can be expressed as:

$$\hat{y}_{ui} = \sigma(h^T F) \quad (10)$$

where \hat{y}_{ui} represents the predicted score of user u on item i , and h represents the hidden unit.

The loss function is defined as the squared error between the actual and predicted scores, which is expressed as:

$$J(\theta) = \sum_{y_{ui} \in R} (\hat{y}_{ui} - y_{ui})^2 \quad (11)$$

where \hat{y}_{ui} represents the predicted score, y_{ui} represents the real score, and the training goal is to minimize the loss function through gradient descent.

In summary, the HTRM model algorithm proposed in this paper is shown in Algorithm 1.

Algorithm 1 HTRM recommendation algorithm based on hybrid neural network

Input User ratings, item ratings, item tags, and user tag behavior

Output Project prediction score

- 1: for epoch = 1 to epochs do
 - 2: for n = 1 to N do
 - 3: Encode the project ID
 - 4: Feature extraction of item tags through autoencoder
 - 5: Represent the project matrix through the project gating layer
 - 6: for m = 1 to M do
 - 7: Encode the user ID
 - 8: Feature extraction of user label behavior through LSTM
 - 9: Represent the user matrix through the user gating layer
 - 10: Joint representation of user and project gating layer
 - 11: Predict item scores through fully connected layer
 - 12: Minimize the loss function
-

Experimental Results and Analysis

Data set and experimental environment

Different traffic participants, such as Intelligent Connected Vehicles, Recreational Vehicles, and pedestrians, exist in mixed traffic and interact with the cloud platform through the terminal platform. The experiment employs traffic travelers' preferences for music, Web bookmarks, movies, and other services during travel as examples. Three public data sets, Last.Fm, Delicious, and MovieLens 20m, are then introduced to simulate the Web Service performed by traffic participants in a mixed transportation system and evaluate and compare the recommended performance of the HTRM model. The specific information of the above-mentioned data set used for simulation analysis is provided in Table 1. For each data set, the training set, validation set, and test set account for 70%, 20%, and 10% of the total data set, respectively.

Table 1. Experimental data set information.

Data set	Number of user	Number of service	Number of tag	Number of making tag
Last.Fm	1867	69226	53388	437593
Delicious	1867	69226	53388	437593
MovieLens	2113	10197	13222	47957

The experiment was run on a Windows 10 64-bit operating system, with an Intel(R) Core(TM) i7-8750H CPU 2.20 GHz processor and 8G memory, using Python language and based on the PyCharm and Jupyter integrated environment.

Experimental design and evaluation indicators

We employed the Keras deep learning framework for neural network construction, using NVIDIA 1080 Ti GPU for model acceleration. During the experimental training, Adam optimizer was used, the batch size was 128, and the learning rate was initialized to 0.001. The model parameters were initialized with Gaussian distribution, and the bias term was set to a zero vector. The word embedding adopted the Word2vec model, the autoencoder used two hidden layers, and the LSTM model adopted a single-layer network structure containing 50 hidden units. For the three public data sets of Last.Fm, Delicious, and MovieLens 20m, the HTRM model was run three times, and the average value on the test set was selected as the experimental result. The evaluation indicators were Mean Squared Error (MSE) and Mean Absolute Error (MAE).

$$MSE = \frac{1}{m} \sum_{u=1}^m \frac{1}{n} \sum_{i=1}^n (\hat{y}_{ui} - y_{ui})^2 \quad (12)$$

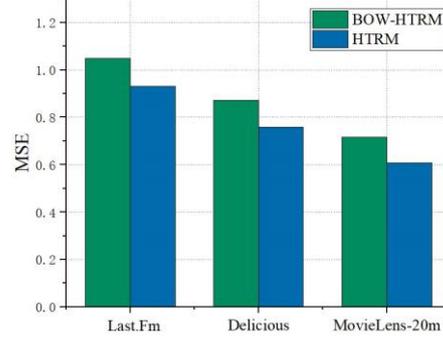
$$\text{MAE} = \frac{1}{m} \sum_{u=1}^m \frac{1}{n} \sum_{i=1}^n | \hat{y}_{ui} - y_{ui} | \quad (13)$$

where m and n are the number of users and the number of projects, respectively.

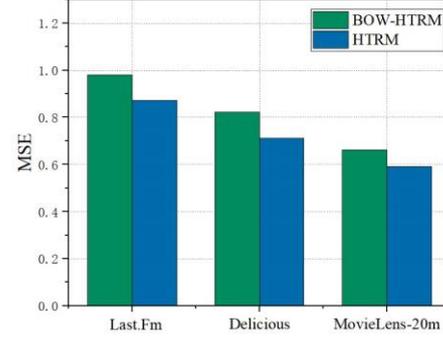
Experimental results and analysis

In order to evaluate the recommendation algorithm performance of the proposed HTRM model, it was compared against the alternative recommendation algorithms tag-based collaborative filtering (TCF), autoencoder-based collaborative filtering (ACF), clustering-based collaborative filtering (CCF), and deep-semantic similarity based personalized recommendation (DSPR) [21]. Among them, TCF is a tag-based collaborative tag filtering method; ACF is a tag-aware recommendation method based on a deep neural network, which uses an autoencoder to extract deep features from the tag latent space layer by layer; CCF is a method of aggregating redundant tags into the same cluster for recommendation; DSPR is the latest method in the tag-aware recommendation system, which maps the configuration files based on tag users and items to the high-order tag latent space for deep semantic similarity calculation.

The choice of the word embedding model has different effects on the performance of a recommendation algorithm. In the embedding layer of the HTRM model, the recommendation performance comparison results of the Word2vec model and the BOM model are shown in Figure 2. Compared with the BOW-HTRM recommendation method, the proposed HTRM model adopts the Word2vec model and is tested and verified by Last.Fm, Delicious, and MovieLens 20m data sets, obtaining corresponding recommendation performance improvements of 11.2%, 13.4, and 10.6%, respectively. This is largely attributed to the enhanced feature extraction performance of the Word2vec model, which avoids label redundancy and semantic model problems. In addition, compared with the Word2vec model, the vector dimensionality generated by the recommendation algorithm using the BOW model is higher, resulting in a greater computational resource overhead.



(a) MSE of BOW-HTRM and HTRM



(b) MAE of BOW-HTRM and HTRM

Figure 2. Comparison diagram of recommended performance of different word embedding models.

In order to evaluate the influence of the number of neurons in the embedding layer on the performance of the recommendation algorithm, the number of neurons is set to 8, 16, 24, 32, and 64, respectively. Figure 3 shows the performance comparison results of different models used in the Last.Fm dataset for recommendations. It can be seen that as the number of neurons in the embedding layer increases, the MSE value tends to decrease; that is, the recommendation performance is improved, but as the number of neurons continues to increase, the MSE value increases again, and the recommendation performance becomes worse. Among them, the recommendation performance is optimal when the number of neurons is configured to be 24, and the recommendation performance is the worst when the number of neurons is 8. The recommended performance of 24 neurons is 11.1% and 8.9% higher than that using 8 and 64 neurons, respectively. The recommendation performance under different data sets shows a trend of first rising and then falling. The main reason for this trend is that the number of neurons is set too small, which will cause information leakage. When the number of neurons is between 8 and 24, the information leakage problem is continuously alleviated, which continuously improves the recommendation performance. However, when the number of neurons is set too high, a large amount

of redundant information will be generated in the embedding layer, which will cause the performance of the recommendation algorithm to deteriorate constantly.

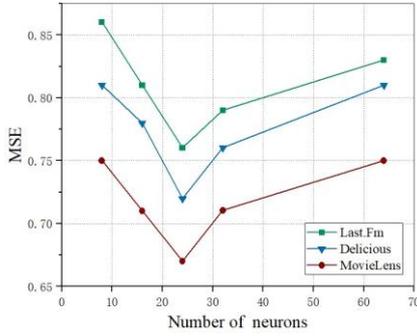


Figure 3. Comparison of recommended performance with different numbers of neurons.

User behavior is time-dependent, and the LSTM model is suitable for feature extraction of user behavior related to time series. Figure 4 illustrates the comparison results of different user behavior label feature extraction methods using the Last.Fm data set. Among them, NN represents a traditional fully connected neural network, and NN@2 represents a two-layer neural network for the feature extraction of user tag behavior. Compared to the feature extraction method of user behavior tags using the traditional fully connected neural network method, the experimental results show that better recommendation performance can be obtained through the LSTM network. Specifically, the MSE index performance of LSTM@1 compared to NN@1 is optimized by 17.9%, and LSTM@3 is optimized by 19.8% compared to NN@3.

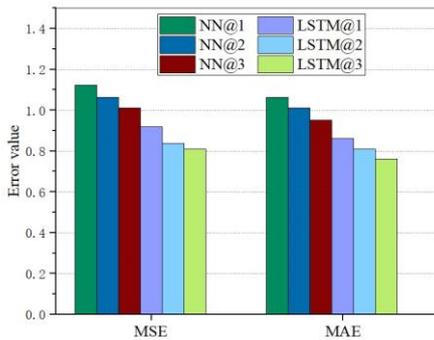


Figure 4. Performance comparison diagram of user behavior label feature extraction.

In the HTRM model, the gating layer is responsible for the fusion of label features and scoring features. Unlike other feature fusion technologies, the HTRM model uses two gating operations to perform feature fusion on item tags and item ratings, as well as user behavior tags and user ratings. To verify the effectiveness of the feature

fusion method of gating operations, the Last.Fm data set is used to compare the feature fusion performance of gating operations and multiplication operations, concatenation operations, and addition operations. The results are provided in Figure 5, showing that the performance after feature combination is better, which verifies the effectiveness of gating operations in the HTRM model. Specifically, under the MAE index, the performance of gating operations is increased by 40.5%, 21.6%, and 12.7%, respectively, compared with multiplication, concatenation, and addition operations, indicating that the feature interaction layer can integrate feature extraction information into the new tag and scoring features more effectively, which can improve the recommendation performance.

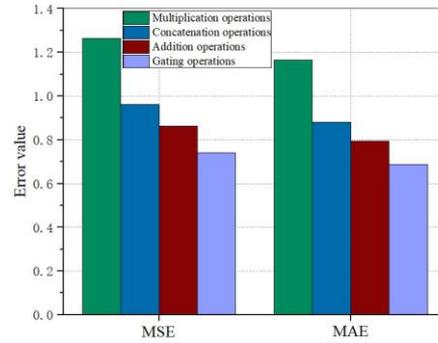
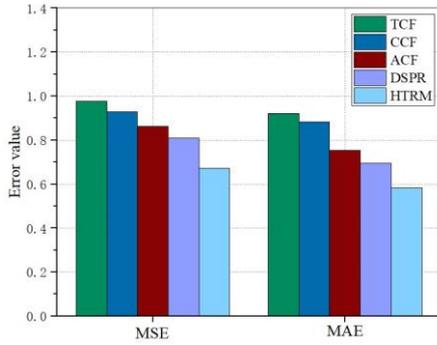
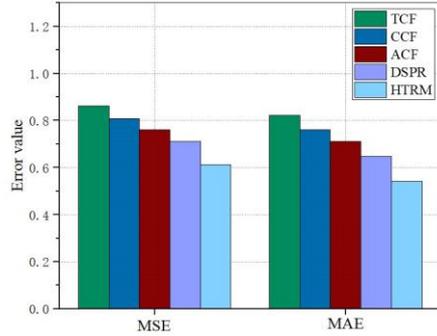


Figure 5. Performance comparison diagram of different feature fusion methods.

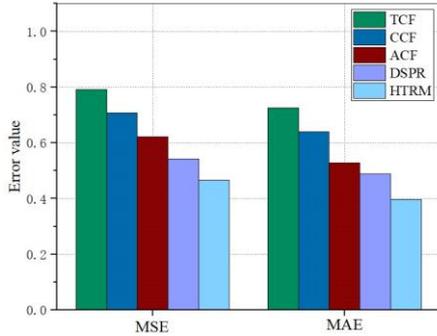
Figure 6 shows the performance comparison between the HTRM model and the other four types of tag-based recommendation models. It can be seen that the recommendation algorithm based on the HTRM model has the best performance, followed by the DSPR, ACF, CCF, and TCF models. Among them, compared with the TCF, CCF, ACF, and DSPR models, the MAE performance of the HTRM model recommendation algorithm in the Delicious dataset experiment is optimized by 34.1%, 28.9%, 23.9%, and 16.9%, respectively. Because the HTRM model uses the autoencoder and the LSTM network to extract the text features of the item tags and user behavior tags while considering the scoring information and tag features, it avoids the problem of insufficient data mining, thereby improving the performance of the recommendation algorithm.



(a) Last.Fm



(b) Delicious



(c) MovieLens

Figure 6. Performance comparison diagram of different recommended algorithms.

Conclusion

The application of the recommendation system in a mixed traffic scenario analyzes the historical travel data of different traffic participants to determine user travel service behavior preferences. The system combines traffic information to recommend suitable travel services to travelers in the form of an information push. The performance of traditional collaborative filtering algorithms is restricted by the diversification of road network users and networking modes and the dynamic changes of user interest distribution caused by the high-speed movement of vehicles. This paper proposes a combined recommendation model that integrates label perception under mixed traffic. It extracts the scoring embedding and label text features of travel service items and the user information of traffic

participants and combines the data with joint feature learning and service prediction to improve the performance of the recommendation algorithm. The proposed Web service recommendation algorithm is tested and validated using multiple public data sets regarding service preferences, such as music, web bookmarks, and movies, during the travel process. The experimental results show that the proposed model has structural advantages and an improved recommendation performance compared with the traditional recommendation model.

Acknowledgements

We would like to thank participants to IEEE SmartIoT conference for their observations.

Authors' contributions

X.L. and F.D. completed the methodology; S.R. and J.B. completed the design; X.L. and S.R. wrote the paper. F.D. revised the paper. R.S. and E.T. reviewed the paper.

Authors' information

Xiangyuan Li (X.L.) is a graduate student. Her main research interests include mobile communication big data technology and intelligent information processing.

Fei Ding (F.D.) is an associate professor. His main research interests include intelligent system and information processing, big data technology, etc.

Suju Ren (S.R.) is a graduate student. Her main research interests include recommendation systems and predictive algorithms.

Jianmin Bao (J.B.) is a professor. His main research interests include Internet of Things and big data.

Ruoyu Su (R.S.) is a lecturer. His main research interests include Internet of Things and networking system.

En Tong (E.T.) is a Professorate Senior Engineer. His main research interests include Internet of Things and information processing.

Funding

This work is supported by Industrial Technology Basic Public Service Platform Project of the Ministry of Industry and Information Technology (No. 2019-00892-3-1); Communication Soft Science Research Project of the Ministry of Industry and Information Technology (No. 2019-R-26); the Six Talent Peaks Project of Jiangsu Province (No. DZXX-008); the China Postdoctoral Science Foundation (No. 2019M661900); the Postdoctoral Research Funding Program of Jiangsu Province (No. 2019K026); NUPTSF (No. NY220028).

Availability of supporting data

We used open source data sets to test and verify, and the model parameters we used are clearly listed in the manuscript itself.

Declarations

Ethics approval and consent to participate

Not applicable.

Consent for publication

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Author details

¹Jiangsu Key Laboratory of Broadband Wireless Communication and Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China. ²School of Internet of Things, Nanjing University of Posts and Telecommunications, Nanjing 210003, China. ³China Mobile-Nanjing University of Posts and Telecommunications 5G Joint Innovation Center, Nanjing 210003, China.

References

1. Ho CL, Li PS, Wang YC, Chou PY, Pan YL, Chen ST (2021) Constructing a Customized Travel Scheduling Recommendation Service Based on Personal Preference and Special Requirements. In: International Conference on Smart Grid and Internet of Things. pp 414-425. https://doi.org/10.1007/978-3-030-69514-9_32
2. Chen X, Luo L, Hu ZJ, Pei XJ, Peng QQ, Han MY, Wei LJ, Shin H (2020) A Travel Recommendation Method Based on User Personalized Characteristics with Collaborative Fusion Matrix. In: 2020 5th International Conference on Mechanical, Control and Computer Engineering (ICMCCE). pp 1298-1301. <http://doi.org/10.1109/ICMCCE51767.2020.00284>
3. Shit RC (2020) Crowd intelligence for sustainable futuristic intelligent transportation system: a review. *Iet intelligent transport systems* 14(6):480-494
4. Ricci F, Rokach L (2015) Recommender systems: introduction and challenges. In: Shapira B (ed) *Recommender systems handbook*, vol 42. Springer, Boston, pp 1-34
5. Zhou X, Tan WA (2020) An Improved Collaborative Filtering Algorithm Based on Filling Missing Data. In: International Conference on Human Centered Computing. pp 220-226. http://doi.org/10.1007/978-3-030-70626-5_23
6. Gao N, Yang M (2015) An Improved Unifying Tags and Ratings Collaborative Filtering for Recommendation System. *Journal of Nanjing Normal University (Natural Science Edition)* 38(01):98-103
7. Zheng E, Kondo GY, Zilora S, Yu Q (2018) Tag-aware dynamic music recommendation. *Expert Systems with Applications* 106:244-251
8. Bao J M, Ren S J, Ding F (2021) HTRM: A Hybrid Neural Network Algorithm based on Tag-aware. 2021 IEEE International Conference on Smart Internet of Things (SmartIoT). pp 160-165. <http://doi.org/10.1109/SmartIoT52359.2021.00033>
9. Ye JX, Xiong HX (2019) Recommending Personalized Contents from Cross-Domain Resources Based on Tags. *Data Analysis and Knowledge Discovery* 3(2):21-32
10. Shepitsen A, Gemmell J, Mobasher B, Burke R (2008) Personalized recommendation in social tagging systems using hierarchical clustering. In: Proceedings of the 2008 ACM Conference on Recommender Systems. pp 259-266. <http://doi.org/10.1145/1454008.1454048>
11. Zhu ZC, Liu JW, Yan SH (2019) Research and Application of Multi-label Learning in Intelligent Recommendation. *Computer Science* 46(S2):189-193
12. Wen JH, Yuan PL, Cao J, Wang XB, Zhou W (2017) Research on Collaborative Filtering Recommendation Algorithm Based on Topic of Tags. *Computer Engineering* 43(01):247-252+258
13. Ma WK, Li G, Li ZY, Han ZY, Cao KY (2019) Top-N Personalized Recommendation Algorithm Based on Tag. *Computer Science* 46(S2):224-229
14. Jiang SH, Zhang LY, Zhang ZX (2018) Recommendation Algorithm for Optimizing Diversity Based on Personalization. *Journal of Tianjin University(Science and Technology)* 51(10):1042-1049
15. Meng XF, Zhang XY, Tang YH, Jia D, Qi YX, Mao Y (2019) A Diversified and Personalized

- Recommendation Approach Based on Geo-Social Relationships. *Chinese Journal of Computers* 42(11):2574-2590
16. Cui ZH, Xu XH, Xue F, Cai XJ, Cao Y, Zhang WS, Chen JJ (2020) Personalized Recommendation System Based on Collaborative Filtering for IoT Scenarios. *IEEE Transactions on Services Computing* 13(4):685-695
 17. Hu Y, Peng QM, Hu XH, Yang R (2015) Time Aware and Data Sparsity Tolerant Web Service Recommendation Based on Improved Collaborative Filtering. *IEEE Transactions on Services Computing* 8(5):782-794
 18. Qian WY, Jiang XH, Liao LC, Zhou FM (2015) Recommendation algorithm of taxi passenger-finding locations based on spatio-temporal context collaborative filtering. *Journal of Computer Applications* 35(06):1659-1662+1667
 19. Zuo Y, Zeng JL, Gong MG, Jiao LC (2016) Tag-aware recommender systems based on deep neural networks. *Neurocomputing* 204:51-60
 20. Xu ZH, Chen C, Lukasiewicz T, Miao YS, Meng XW (2016) Tag-Aware Personalized Recommendation Using a Deep-Semantic Similarity Model with Negative Sampling. In: *Proceedings of the 2016 ACM International on Conference on Information & Knowledge Management (CIKM 2016)*. pp 24-28. <http://doi.org/10.1145/2983323.2983874>
 21. Xu ZH, Lukasiewicz T, Chen C, Miao YS, Meng XW (2017) Tag-Aware Personalized Recommendation Using a Hybrid Deep Model. In: *Proceedings of the 26th International Joint Conference on Artificial Intelligence (IJCAI 2017)*. IJCAI:3196-3202
 22. Huang YX, Zhao L, Van Woensel T, Gross J-P (2017) Time-dependent vehicle routing problem with path flexibility. *Transportation Research Part B: Methodological* 95:169-195
 23. Yu H, Zhou B, Deng MY, Hu F (2018) Tag recommendation method in folksonomy based on user tagging status. *Journal of Intelligent Information Systems* 50:479-500
 24. Liang N, Zheng HT, Chen JY, Sangaiah AK, Zhao CZ (2018) TRSDL: Tag-Aware Recommender System Based on Deep Learning–Intelligent Computing Systems. *Applied Sciences* 8(5):799
 25. Jiang WW, Lian J, Shen M, Zhang L (2017) A multi-period analysis of taxi drivers' behaviors based on GPS trajectories. In: *2017 IEEE 20th International Conference on Intelligent Transportation Systems (ITSC)*. pp 1-6. <http://doi.org/10.1109/ITSC.2017.8317622>
 26. Ren XY, Song MN, Song JD (2017) Context-Aware Point-of-Interest Recommendation in Location-Based Social Networks. *Chinese Journal of Computers* 40(04):824-841
 27. Zhou WB, Tang L, Zeng CQ, Li T, Shwartz L, Grabarnik GY (2016) Resolution Recommendation for Event Tickets in Service Management. *IEEE Transactions on Network and Service Management* 13(4):954-967