

RESEARCH ARTICLE

# Individualized tourism recommendation based on self-attention

Guangjie Liu<sup>1</sup>, Xin Ma<sup>1\*</sup>, Jinlong Zhu<sup>1</sup>, Yu Zhang<sup>1</sup>, Danyang Yang<sup>1</sup>, Jianfeng Wang<sup>2</sup>, Yi Wang<sup>3</sup>

**1** College of Computer Science and Technology, Changchun Normal University, Changchun, Jilin, China, **2** School of Data and Computer Science, Sun Yat-Sen University, Guangzhou, China, **3** CRRC Changchun Railway Vehicles CO.,LTD, Changchun, Jilin, China

\* [mxxyundi\\_mj@163.com](mailto:mxxyundi_mj@163.com)



## OPEN ACCESS

**Citation:** Liu G, Ma X, Zhu J, Zhang Y, Yang D, Wang J, et al. (2022) Individualized tourism recommendation based on self-attention. PLoS ONE 17(8): e0272319. <https://doi.org/10.1371/journal.pone.0272319>

**Editor:** Sathishkumar V E, Hanyang University, KOREA, REPUBLIC OF

**Received:** December 9, 2021

**Accepted:** July 17, 2022

**Published:** August 25, 2022

**Copyright:** © 2022 Liu et al. This is an open access article distributed under the terms of the [Creative Commons Attribution License](https://creativecommons.org/licenses/by/4.0/), which permits unrestricted use, distribution, and reproduction in any medium, provided the original author and source are credited.

**Data Availability Statement:** All relevant data are within the manuscript and its [Supporting information](#) files. 1. Tourism dataset is a dataset built on the basis of the “2018 Cloud Mobile Cup Scenic Spot Word-of-Mouth Score Prediction” and the “Tourist scenic spots Data in the ModelWhale Community”, they are both publicly available. “2018 Cloud Mobile Cup Scenic Spot Word-of-Mouth Score Prediction” dataset comes from the National Tourism Big Data Challenge organized by Yunnan University and Yunnan Provincial Society of Applied Statistics, and the official competition platform is DataFountain. After we communicated

## Abstract

Although the era of big data has brought convenience to daily life, it has also caused many problems. In the field of scenic tourism, it is increasingly difficult for people to choose the scenic spot that meets their needs from mass information. To provide high-quality services to users, a recommended tourism model is introduced in this paper. On the one hand, the tourism system utilises the users’ historical interactions with different scenic spots to infer their short- and long-term favorites. Among them, the users’ short-term demands are modelled through self-attention mechanism, and the proportion of short- and long-term favorites is calculated using the Euclidean distance. On the other hand, the system models the relationship between multiple scenic spots to strengthen the item relationship and further form the most relevant tourist recommendations.

## Introduction

The system of recommendation is used in all aspects of life. It not only saves time for users when searching for information, it also brings better benefits to service providers [1, 2]. However, there have been few studies on scenic recommendation systems. Li et al. [3] fit the implicit preferences of users through demographic attribute information, obtaining their preferences for different population attributes through a hierarchical sampling of statistical models, and generating recommendation lists from the mined user preferences. Based on an analysis of the information evaluations made by other users, Alexander et al. [4] also recommend the current best attractions for users to attend based on user preferences and the current status of their areas.

Among the many recommendation techniques [5], item-based collaborative filtering (ICF) comes out to be the most widely applied [6, 7] owing to its low data dependency and easy algorithm implementation [8, 9]. The key to ICF is to discover similarities between items, then suggested similar items to user depending on information regarding the historicals they have [10–12]. The similarity is usually determined by the history of user interaction. Despite the prevalence and effectiveness of ICF methods, they are inadequate because of the fact that they allow only coarse-grained, collaborative similarity relationships lacking concrete semantics. He et al.

with the platform, the platform requested to mark the source, and we have added it to the “Materials and Methods” Section as required. The specific location in the paper on page 8 right-hand line number 254. “Tourist scenic spots Data in the ModelWhale Community” uses the Attribution-NonCommercial Use 4.0 International, which authorizes us to legally use the dataset. The agreement requires the attribution to provide a URL or hyperlink containing the requested information source, which we have added to the “Materials and Methods” Section as requested. The specific location in the paper on page 8 right-hand line number 258.

**Funding:** This work is supported by the Science and Technology Research and Development Project of Jilin Province (20180201086SF) and the Joint fund of Science & Technology Department of Liaoning Province and State Key Laboratory of Robotics, China (Grant No. 2020-KF-22-08). All authors are the recipients of the funding awards.

**Competing interests:** The authors have declared that no competing interests exist.

[13] introduced relational collaborative filtering (RCF) which integrates multiple item relationships to form recommendations, but does not consider the existing rich sequential patterns in users’ historical interactions.

In a user-item historical interaction, the associated timestamps are recorded with the passage of time. All such data can reflect the strong correlation and even causality that exists in the user behaviors [14–18]. Koren et al. [16] argued that simulating the time dynamics is key when designing recommendation systems or general customer preference models, after which they proposed models that can track time-changing behavior over the entire data lifecycle. Wu et al. [19] modelled the temporal evolution of ratings using a recurrent neural network (RNN), which were not designed for the recommendation domain. In 2016, YouTube [6] proposed to apply deep learning to video recommendation and achieved extremely good results. Since then, the spread of deep learning techniques has blossomed, resulting in a variety of papers, academic exchanges, and industrial applications in the field of recommender system. He et al. [20] also used a neural network structure to model user-item interaction data, while using multilayer perceptual machines to learn user-item interaction equations. Convolutional neural network (CNN) can find the features information in a large amount of data and can be generalized to similar problems with similar type [21–25]. Recurrent neural networks perform well in modelling of temporal dynamics [19, 26–28]. Zhang et al. [29] combined the user’s history on websites to make simple recommendations to the user; however, the rich sequential patterns in user interaction and the multiple relationships that exist between items are also extremely important. To better learn the sequence representation and multiple relationships present in the items, a neural sequence recommendation model for scenic spots, i.e., a Self-Attention based individualized Tourism Recommendation (ATTR), is proposed in this paper. The system models the sequence of user interactions through a self-attention and maintains item relationships through an embedding operation. These operations provide accurate analysis of the user’s interests to effectively predict the most suitable item for the user. Finally, prove the usefulness of the model experimentally. The contributions of this work are listed below:

- This paper proposes a new model for sequential recommendation tasks. The model combines the analysis of users’ long- and short-term favorites with modelling the relationship between items to better infer the following behavior of the target user.
- This paper analyses the interaction data between users and scenic items to get users’ long- and short-term favorites. In this method, short-term preferences are modelled using a self-attention mechanism, and item embedding is enhanced by preserving the relationship structure between scenic items.
- Finally, to validate the methodology introduced in this paper by two datasets, demonstrating that the frame attains the most advanced performance. The remainder of this paper is organized as follows: Section 2 explains the knowledge required for the model and the complete model formation process. Section 3 reports on the experimental and performance analyses of this approach. Finally, Section 4 presents a review of this work.

## Materials and methods

### Attention mechanism

In the process of reading and communicating, our attention is not allocated to every word in a balanced manner. To make computers more adaptable to human communication, they must learn to selectively forget and associate the context, which is a mechanism known as an attention mechanism [30–32]. The attention mechanism has developed as a hot research topic in

neural networks and good results has been gained in areas such as computer vision [33], image captioning [34], and machine translation [35], where the original idea of this mechanism lies in the efficient computation of attention distribution. Bahdanau et al. [35] were among the first to use attention as a mechanism to search for relevant parts from the input sequence for the current target item. Sanghyun et al. [36] also proposed adding an attention mechanism to both the feature channel and feature space dimensions of a CNN. The attention mechanism allows the RNN not to be limited by the input sequence length, allowing the CNN to acquire the information that requires more attention. Many more studies in the field of recommendation have been conducted on attentional mechanisms. Li et al. [37] proposed session-based recommendations to generate recommendation results from short sessions. Zhou et al. [38] addressed the problem that a user's interest in different items should be modelled in a way that does work by weighting the user's historical behavior sequences with attention.

In this paper, a new concept of self-attention is addressed and compared to standard attention, self-attention concerns the interaction of two sequences, where the attentional weight of one sequence depends on the other. Self-attention is predicated on attention mechanism. With the successful application of attention mechanism by the Google team [39], self-attention mechanism has also rapidly occupied hotspots in various fields due to several advantages such as its fast training speed, and has successfully solved numerous problems.

### Item-based collaborative filtering

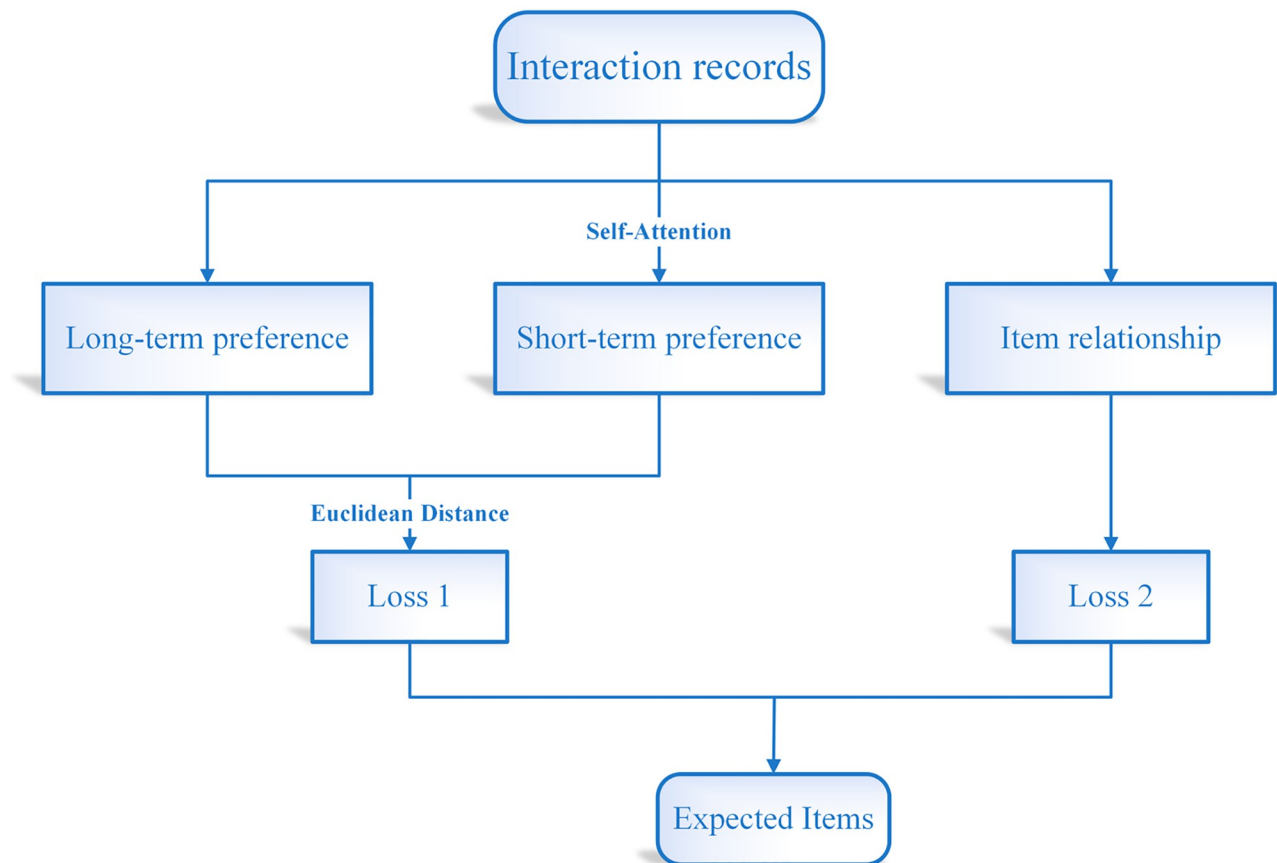
ICF is aimed at mining historical user behavior and uses it as a basis for making recommendations [8, 9]. User prediction scores for items are derived from the similarity between items, where the relationship between the items is called collaborative similarity. The most popular among ICFs is FISM [9], which offers a way to better express user information by portraying the user as an expression of the item that the user has liked. Both neural network enhancement methods [8] and involving local potential energy spaces [40, 41] have been the subject of extensive research in this area. Although these improvements have improved the performance of ICF, the coarse granularity of item relationships and the absence of semantic meaning remain a problem, making it difficult to produce better recommendations. The main differences in this work in comparison to existing approaches are the use of self-attention mechanisms and how the item relationship data can be modelled to introduce relational structures between item embeddings.

First, the users' short-term intention is obtained by modelling their historical interaction behavior by the self-attention, and the influence of long- and short-term intentions on the user is analysed based on the Euclidean distance. Second, item embedding is enhanced by preserving the relational structure between scenic spots.

### Methodology

This section first explains how to get users' long- and short-term intentions, and then build upon this by introducing the modelling of the item relationships. Fig 1 shows the framework of the model. The left half in this figure models the user preferences, in which the short-term interests are modelled by self-attention model, the right half of the figure models the item relationships.

This paper assumes that  $U$  is the user collections and  $I$  is the item collections, where  $|U| = M$  and  $|I| = N$ . Here,  $I^u = (I_1^u, \dots, I_{|I^u|}^u)$  represents the items in the chronological user interaction record, where  $I^u \in I$ . Define the relationship between item pair  $(i, j)$  as the set of  $r = \langle \text{relation type } t, \text{relation value } v \rangle$ . Table 1 presents the symbolic representation of this model.



**Fig 1. Frame diagram for this paper.** The left part of the figure models the user's long- and short-term intentions, of which the short-term intentions are derived from the self-attention mechanism. The right half of the figure models the item relations.

<https://doi.org/10.1371/journal.pone.0272319.g001>

**User preference modelling.** A better understanding of short-term preferences can be gained by analysing the user's recent behavior. In this paper, their interaction recordings were modelled through self-attention to obtain short-term intentions. The self-attention is special case where the query, value, and key are identical in attention, and all consist of the interaction data about user and item. The mechanism of attention is essentially a weighted summation of

**Table 1. Symbolic representation of the model in this paper.**

Notation	Narrate
$U$	user collections
$I$	item collections
$I^u$	historical interaction sequence for a user $u : (I_1^u, \dots, I_{ u }^u)$
$I_r(i, j)$	item relation function, when $I_r(i, j) = 1$ means that relation $r$ exists for item $i$ and item $j$
$\mathcal{T}$	relation type collections
$\mathcal{V}$	relation value collections
$q^i \in R^{N \times d}$	embedding of item $i \in I$
$x_t \in R^{N \times d}$	embedding of relation type $t \in \mathcal{T}$
$z_v \in R^{N \times d}$	embedding of relation value $v \in \mathcal{V}$

<https://doi.org/10.1371/journal.pone.0272319.t001>

the values of the elements, and the query and key are used to calculate the weighting factors for the corresponding values.

Assume that the user's recent preference are acquired from the nearest  $L$  (e.g., 5, 10) item interaction, and item can be expressed as a  $d$ -dimensional embedding vector. Set all item embeddings denoted as  $X \in R^{N \times d}$ . Stack the most recent  $L$  items sequentially to obtain the matrix, as in Eq (1):

$$\mathbf{X}_t^u = \begin{bmatrix} X_{(t-L+1)1} & X_{(t-L+1)2} & \cdots & X_{(t-L+1)d} \\ \vdots & \vdots & \vdots & \\ X_{(t-1)1} & X_{(t-1)2} & \cdots & X_{(t-1)d} \\ X_{t1} & X_{t2} & \cdots & X_{td} \end{bmatrix}. \quad (1)$$

The latest  $L$  items are a subset of  $I^u$ . In this part, user  $u$ ' query, key, and value of at step  $t$  are equal to  $X_t^u$ . First, the query and key are projected into the same space via a nonlinear activation function  $ReLU$  with shared parameters, as in Eqs (2) and (3):

$$Q' = ReLU(X_t^u W_Q). \quad (2)$$

$$K' = ReLU(X_t^u W_K). \quad (3)$$

here,  $W_Q \in R^{d \times d}$  is the weight matrix of the query and  $W_K \in R^{d \times d}$  is the weight matrix of the key. Then, the product of  $Q'$  and  $K'$  are calculated. To avoid overly large results, the result is divided by a scale  $\sqrt{d}$ , and the affinity matrix is computed as in Eq (4) [39]:

$$s_t^u = softmax\left(\frac{Q'K'^T}{\sqrt{d}}\right). \quad (4)$$

$s_t^u$  is an  $L \times L$  matrix that shows the similarity between the  $L$  terms, and  $d$  is initialized to a larger value (e.g. 100). For avoiding high-patch points between equal query and key vectors, the masked affinity matrix diagonal operation is adopted before softmax is applied.

Then, let the value and  $X_t^u$  be the same. Unlike other cases in which linear transformations are typically used to map values, the use of identity mapping in this model is beneficial. In other application areas such as word embedding, values are usually pre-trained feature embeddings, whereas in this paper, values are composed of parameters that need to be learned. The difficulty of viewing the actual parameters can be made by adding a linear or non-linear transformation. Queries, keys, and values are not sensitive to transformations in the same way because queries and keys are used as subsidiary factors.

Finally, the resulting matrix is multiplied by the value to produce a representation of the weight summation, as in Eq (5):

$$a_t^u = s_t^u X_t^u. \quad (5)$$

The short-term interests of users are represented by this output. To learn individual attention representations, the user's short-term intention is represented by the minimal embedding in the  $L$  self-attention denotation, as indicated in Eq (6):

$$m_t^u = \min a_t^u. \quad (6)$$

The above formulas also operate with the *sum*, *max* and *mean*, the validity of which is compared in a later section. The time signal is not included in the above model and needs to be

added to this model to preserve the sequence pattern. In the next work, we propose to provide time signals for the query and key through positional embedding. Next, sinusoidal signals of different frequencies are added to the input using a geometric time-scale sequence. The two sine signals form the time embedding ( $TE$ ), as shown in Eqs (7) and (8):

$$TE(t, 2i) = \sin(t/10000^{2i/d}). \quad (7)$$

$$TE(t, 2i + 1) = \cos(t/10000^{2i/d}). \quad (8)$$

Here,  $t$  denotes the time step,  $i$  denotes the dimension. Before the query and key are nonlinearly transformed,  $TE$  is added.

After modelling short-term intentions, considering the combination of the users' general and long-term preferences will yield better overall recommendations for users. As with the latent factor approach, a latent factor is assigned to each user and each item. Set  $U' \in R^{M \times d}$  and  $V \in R^{N \times d}$  as the potential factors for both the user and item. Affinity between  $u$  and  $i$  is measured by the Euclidean distance, as shown in Eq (9) [42]:

$$\|U'_u - V_i\|_2^2. \quad (9)$$

If user  $u$  likes item  $i$ , then this distance should be small, and if user  $u$  does not like item  $i$ , then this distance should be large.

Predict the items (denoted by  $I_{t+1}^u$ ) which user  $u$  probably interacts with at time step  $t + 1$  by modelling their short- and long-term preferences at the previous  $t$  steps. For consistency, Euclidean distance is used to predict the weight of short- and long-term preferences, which is used as a recommendation score, as indicated in Eq (10):

$$y_{t+1}^u = \omega \|U'_u - V_{I_{t+1}^u}\|_2^2 + (1 - \omega) \|m_t^u X_{t+1}^u\|_2^2. \quad (10)$$

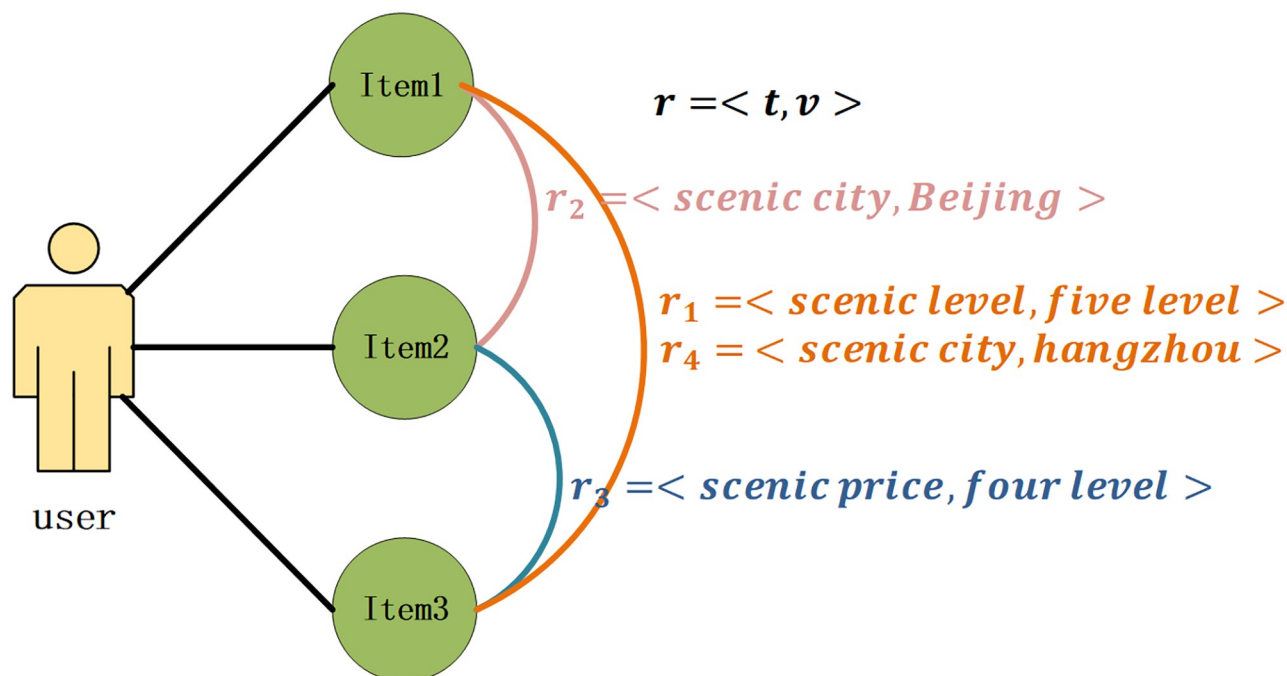
In the formula, the first term is the product of the control factor  $\omega$  and the user  $u$  long-term intention score for the next item  $I_{t+1}^u$ , and the next term is the product of the control factor  $\omega$  and the user  $u$  short-term interest score for item  $I_{t+1}^u$ . Here,  $V$  and  $X$  are distinct parameters, whereas both  $V_{I_{t+1}^u}$  and  $X_{t+1}^u$  represent the  $t+1$ th item of the embedding vector.

This work aims to predict not just one item, but the next few items of user  $u$ . This requires our model to capture the jumping behavior. Make  $T^+$  indicate the next  $T$  items that interact with the user in the groundtruth. It is also necessary to collate items which user do not interact with and denote them with  $T^-$ .  $T^+$  and  $T^-$  from set  $I$ . The goal of doing this procedure is to easily learning the model variables using pairwise sorting, using the losses as in Eq (11):

$$L_1 = \sum_{(u,i) \in T^+} \sum_{(u,i) \in T^-} [y_i^u + \gamma - y_i^u]_+ + \lambda \|\theta\|_2^2. \quad (11)$$

here,  $\theta = \{X, V, U, W_Q, W_K\}$  stands for the parameters of the model, and  $\gamma$  represents the margin that divides  $T^+$  and  $T^-$ .

**Item relational data modeling.** The following is the modelling of the item relationships. Fig 2 shows an example of multiple relationships among items from the user interaction data. The item relationship  $r$  as a function of relation type  $t$  and relation value  $v$ ,  $r = \langle t, v \rangle$ . For example, there is a relationship  $r_2$  between Item1 and Item 2, and Item 2 and Item 3 also exist relationship  $r_3$ . There may be more than one relationship between two items, as shown in Fig 2, there are two relationships between Item 1 and Item 3. Knowledge graphs, as an emerging type of auxiliary information, have gradually captured the eye of society over the past few years. The knowledge graph store real-world entities and the relationships between entities



**Fig 2. Diagram illustrating the multiple relationships between items in the user interaction history.** Each relation  $r$  is described by type  $t$  and value  $v$ . There is a relationship  $r_2$  between Item 1 and Item 2, the relationship type is the scenic city, and the relationship value is Beijing. Multiple relations may also exist between two items.

<https://doi.org/10.1371/journal.pone.0272319.g002>

whose nodes indicate entities or concepts and whose edges indicate all kinds of contextual relationships between entities or concepts. A knowledge graph consists of several ternary groups  $(\mathcal{h}, r, \mathcal{t})$ , where  $\mathcal{h}$  and  $\mathcal{t}$  stand the head and tail nodes of a relationship, and  $r$  represents the relationship. An effective way to derive signals in relational data is to embed knowledge graphs into a space of continuous vectors. However, the direct use of knowledge graph embedding techniques has certain problems in the recommendation domain:

1. Item relationships are defined as a two-level structure: relationship type and relationship value. To represent this relationship correctly, the relationship between the two levels of model fidelity must be considered. Therefore, single embeddings cannot be assigned to item relationships. To resolve this problem, it use the two levels of layered parts as relational embedding, which can be expressed using Eq 12:

$$r = x_t + z_v. \quad (12)$$

2. Unlike traditional knowledge graphs that are represented by directed graphs, item relationships are invertible (i.e. relation  $r$  is valid for  $(\mathcal{h}, \mathcal{t})$  and  $(\mathcal{t}, \mathcal{h})$ ), form an undirected graph structure. The most widely used graph, TransE [18], maps the relationship between two entities to an embedded action between them, whereby  $\mathcal{h} + r \approx \mathcal{t}$  when  $(\mathcal{h}, r, \mathcal{t})$  holds, where the embedding of the head entity is denoted by  $\mathcal{h}$ , the embedding of the relation is denoted by  $r$ , and the embedding of the tail entity is denoted by  $\mathcal{t}$ . Comprised of the above, TransE frames the triplet's scoring function as  $f(\mathcal{h}, r, \mathcal{t}) = \|\mathcal{h} + r - \mathcal{t}\|_2$ , where  $\|\cdot\|_2$  denotes the  $L_2$  criterion of the tensor. Owing to the non-directional structure, both  $\mathcal{h} + r \approx \mathcal{t}$  and



$\mathbf{e} + r \approx \mathbf{e}$  are obtained. An objective function optimized in this way may yield an insignificant solution with  $r \approx 0$  and  $\mathbf{e} \approx \mathbf{e}$ . To solve this problem, the origin is found to be the subtractive operation of TransE, which applies only to directed structures. We require a model that solves the exchange rule (i.e.  $f(\mathbf{e}, r, \mathbf{e}) = f(\mathbf{e}, r, \mathbf{e})$ ). DistMult [43] is another advanced approach for knowledge graph embedding, and expresses the rating function as  $f(\mathbf{e}, r, \mathbf{e}) = \mathbf{e}^T M_r \mathbf{e}$ , where  $M_r$  is a matrix denotation of  $r$ . DistMult clearly fulfils these requirements. From the above, we define items  $i$  and  $j$  with relation  $r$  as  $(i, r, j)$ , and their function as Eq 13:

$$f(i, r, j) = q_i^T \cdot \text{diag}(r) \cdot q_j. \quad (13)$$

where  $\text{diag}(r)$  represents diagonal matrix with the same value of diagonal elements and  $r$ . In this part, it is necessary to maximize  $f(i, r, j)$  with positive instances and minimize it with negative instances. The target function is refined by comparing the points of the interacted triplets  $(i, r, j)$  to the uninteracted triple  $(i, r, j^-)$ , as shown in Eq 14:

$$L_2 = - \sum_{(i, r, j, j^-) \in D_R} \ln \sigma(f(i, r, j) - f(i, r, j^-)). \quad (14)$$

here  $D_R$  is defined as Eq 15:

$$D_R = \{(i, r, j, j^-) | i, j, j^- \in I \wedge I_r(i, j) = 1 \wedge I_r(i, j^-) = 0\}. \quad (15)$$

**Model learning.** During the recommendation phase, the item's recommendation score is calculated, and the candidate items are sorted in ascending order. Then the user is recommended the highest ranked items. To efficiently learn the recommendation parameters and retain the relational structure between project embeddings, the sequential recommendation section and the relational modelling section are learned end-to-end using a multitasking framework. The overall target function of this work is given by Eq 16:

$$L = L_1 + L_2. \quad (16)$$

Fig 3 shows the structure of the paper. It contains the long- and short-term intentions of the user, and it contains the relationships between items. Both were added together to form the eventual recommendation list. The short-term intentions of users are inferred through self-focused networks, as well as by building the entire system within the framework of measurement learning.

## Results and discussion

In this part, two real datasets were used to experiment and assess the proposed sequence recommendation model. The aim of this work is by answering the following issues:

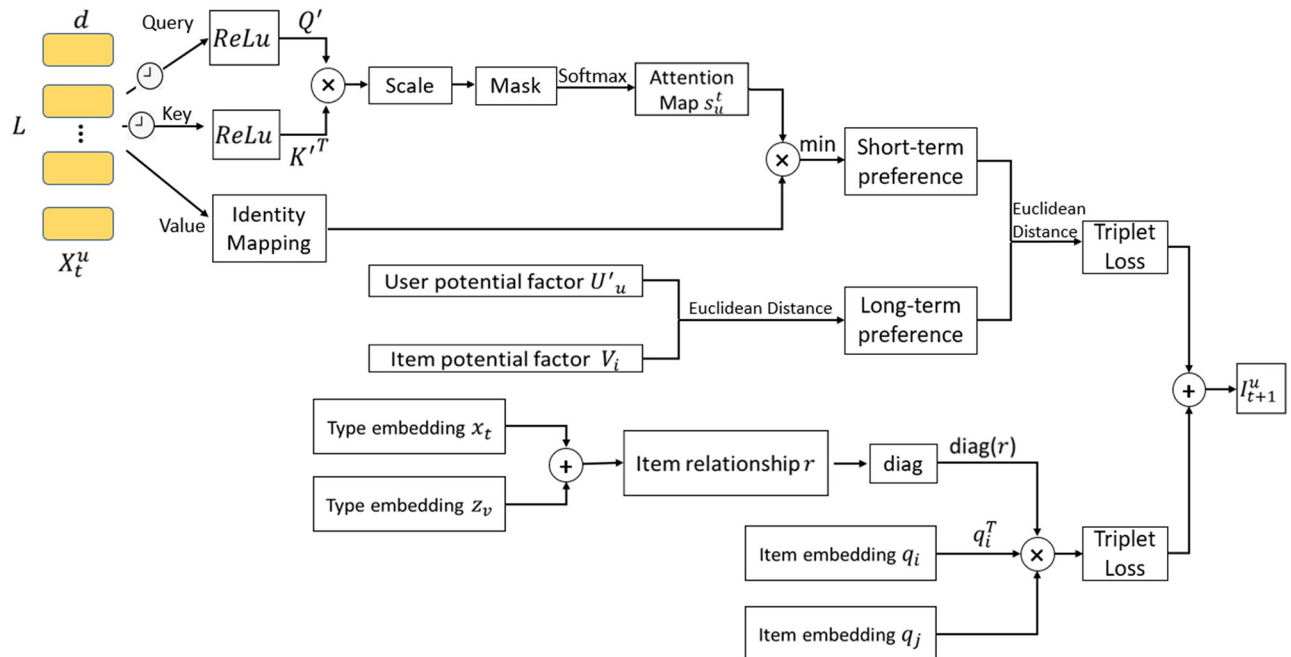
**RQ1:** Does the self-attention based model introduced in this paper perform the advanced performance?

**RQ2:** What are the implications of the critical hyper-parameters?

## Dataset descriptions

In this paper, two datasets were used for the experiments: Tourism Dataset, and the hetrec2011 dataset on movie recommendations.





**Fig 3. Structural diagram of this paper.** The upper half of the diagram is composed of modelling the users' long- and short-term intentions and the lower half is the modelling of the item relationships. The self-attention mechanism is adopted to analyse the users' short-term intentions, and the Euclidean distance is applied to simulate the influence of long- and short-term intentions.

<https://doi.org/10.1371/journal.pone.0272319.g003>

#### 1. Tourism Website(<https://github.com/DATASU10/DATASET>)

Tourism dataset is a dataset built on the basis of the “2018 Cloud Mobile Cup Scenic Spot Word-of-Mouth Score Prediction”(Comes from the National Tourism Big Data Challenge organized by Yunnan University and Yunnan Provincial Society of Applied Statistics, and the official competition platform is DataFountain) and the “Tourist scenic spots Data in the ModelWhale Community” (<https://www.heywhale.com/mw/dataset/6108b262911b330017451cc7/file>), and they are both publicly available. Tourism dataset contains 70,544 data records from 850 users for 678 scenic spots and is available at <https://github.com/DATASU10/DATASET>. The dataset contains two parts: the interaction records of users and scenic spots (including userID, scenic spotID, ratings, and timestamps), and the relationship data of 678 scenic spots (including scenic spotID, scenic spot city, scenic spot level, and scenic spot ticket price). The interaction records between users and scenic spots are composed of the “2018 Cloud Mobile Cup Scenic Spot Word-of-Mouth Score Prediction”, and the relationship data of scenic spots are composed of the “Tourist scenic spots data in the ModelWhale Community”.

#### 2. Hetrec2011(<https://grouplens.org/datasets/hetrec-2011/>)

Hetrec2011 is a public dataset available at <https://grouplens.org/datasets/hetrec-2011/>. It contains 199997 data records, which is a dataset for recording interactions between users and movies. Select data on the interaction record between users and movies (including userID, itemID, ratings, and timestamps) and data on the relationship between movies (itemID, country of movie, movie genre, and movie director). The main parts used are the four files userRatedmovies-timestamps.dat, movie\_genres.dat, movie\_director.dat and movie\_countries.dat.

Table 2. Dataset description.

Dataset	Tourism Website	Hetrec2011
#user	678	511
#item	850	8708
#interaction	70544	199997

<https://doi.org/10.1371/journal.pone.0272319.t002>

Datasets with explicit scores are converted into implicit feedback. The detailed statistics for the dataset are presented in Table 2.

## Evaluation metrics

For each user, this paper use nearest item for testing and conducts hyper-parameter tuning using the second nearest item. The hit rate (*HR*), mean reciprocal (*MRR*) and normalized discounted cumulative gain (*NDCG*) were taken to evaluate the capability of all models. The *HR* measures the correctness of the recommendation. The *HR* is reported to have a stop value of *k* (*k* = 5, 10), which is defined as Eq 17:

$$HR@k = \frac{1}{M} \sum_{u \in U} 1(R_{u,g_u} \leq k). \quad (17)$$

Here,  $g_u$  is the rank produced by the model for this groundtruth item.

The mean reciprocal rank indicates where the model ranks the item.  $MRR@k$  allocates better marks to items on the recommended list. The *MRR* is defined as Eq 18:

$$MRR@k = \frac{1}{M} \sum_{u \in U} \frac{1}{R_{u,g_u}}. \quad (18)$$

Here,  $R_{u,g_u}$  is the rank for the ground truth item.

$NDCG@k$  places highly relevant items at the top of the recommendation list, emphasizing the sequential nature of the items. The *NDCG* is defined as Eq 19:

$$NDCG@k = \sum_{i=1}^k \frac{2^{r_i} - 1}{\log_2(i + 1)} \quad (19)$$

Here,  $r_i$  is the user's preference value for the *i*-th item among the first *k* items.

## Compared models

The model introduced in this paper is compared with traditional approaches and more advanced models. Specifically, the introduced model is measured against the baseline below:

**AttRec** [29]: This is a sequence-aware recommendation model that uses a self-attention mechanism to model the interaction between the user and history, resulting in a final user representation.

**TiSASRec** [44]: The approach proposes a time interval self-attention mechanism to model the time interval in user interactions to better infer user preferences.

**LSSA** [45]: The method proposes a multilayer long- and short-term self-attention network for sequential recommendation that combines long-term and short-term favorites of users to capture their complex preferences.

**RCF** [13]: This approach proposes a new item-based collaborative filtering framework designed to integrate relationships across multiple items for better recommendations.

**FISM** [9]: This is the most advanced ICF model that describes users in terms of the average aggregation of interaction item embeddings.

**NAIS** [8]: This method enhances the FISM by displacing the average aggregation of the FISM with an attention-based summation.

**MF** [46]: It uses the inner product of the user and the interaction item to simulate the user preferences. This is a standard matrix factorization method.

Given that the adaptive gradient optimizer is adopted in this model, the learning rate is fixed at 0.05. To assure a balanced comparison of the model performance, the latent dimension  $d$  for the model introduced in this paper and for all models in which this variable is present is fixed at 100. The effect of  $d$  in this model is explained in the following section. The regularization rate  $\lambda$  is adjusted between 0.1, 0.01, 0.001, 0.0001. The dropout rate is adjusted between 0, 0.3, 0.5, 0.7. The weight factor  $\omega$  is adjusted between 0, 0.2, 0.4, 0.6, 0.8, 1.0. The sequence length of  $L$  is fixed at 5. The target length  $T$  is fixed at 3. The margin  $\gamma$  of the hinge loss is set to 0.5 for all datasets. The experimental part of this paper is realized in Python using TensorFlow.

## Model comparison

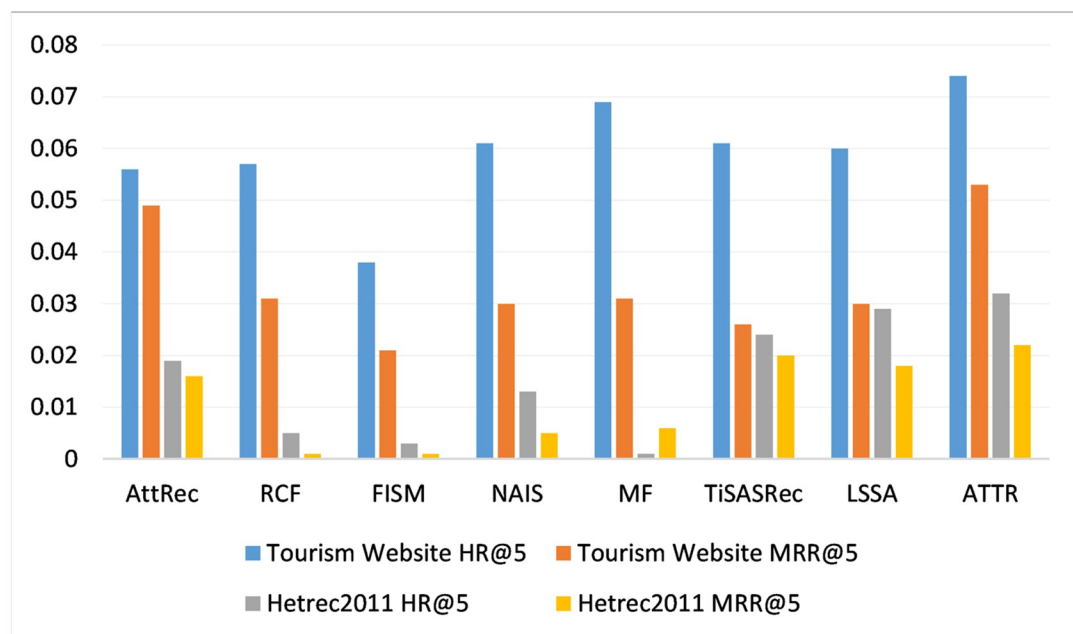
Table 3 shows the experimental outputs for the seven baselines and the model introduced in this paper on two datasets. The table shows that this model always achieves a good performance on both datasets. This also establishes the validity of all methods applied.

In contrast to the sequence-aware recommendation model, AttRec, TiSASRec and LSSA does not model the relationships between items, but only the historical interactions of users using the self-attention mechanism. The importance of modelling the item relationships can also be illustrated from this perspective. Compared to RCF, FISM, and NAIS, the latter only considers the collaborative similarity. In this work not only the long and short term preferences of users are modeled, but also the item relationships are analyzed, where the self-attention mechanism is used to model the short term preferences of users and the Euclidean distance is used to calculate the respective shares of long and short term preferences, in addition to considering sequential modeling, which is the main reason for the improvement. From this perspective, the results demonstrate the importance of the sequential modelling. Fig 4(a)

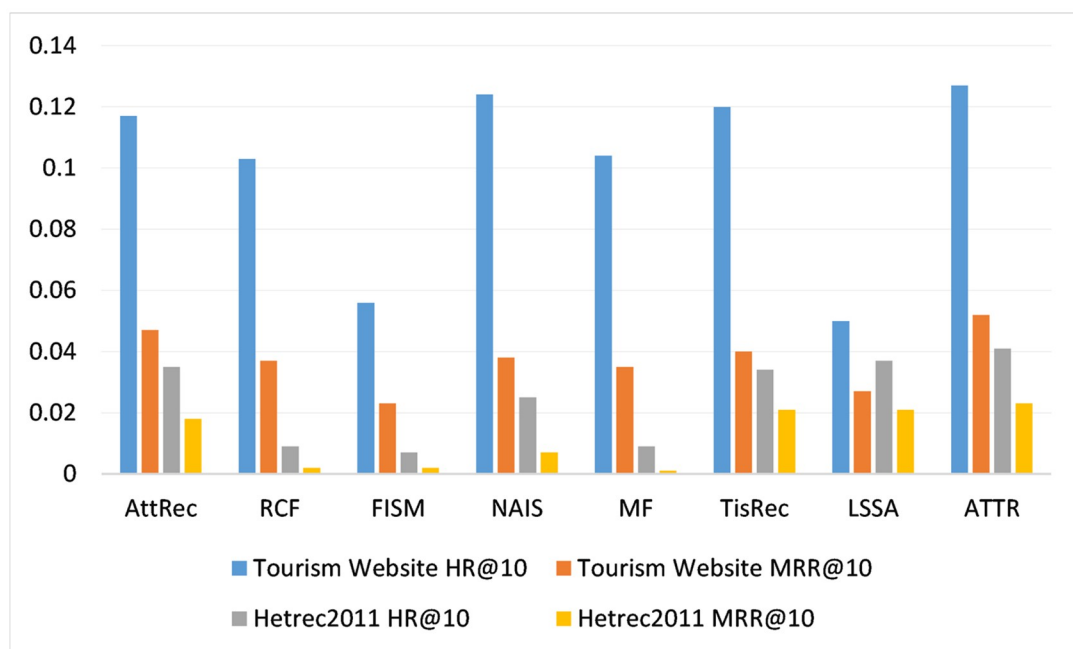
Table 3. Comparison of hit rate (HR), NDCG and MRR performance of all models on two datasets.

Models	Metirc	AttRec	RCF	TiSASRec	LSSA	FISM	NAIS	MF	ATTR
Tourism Website	HR@5	0.056	0.057	0.061	0.060	0.038	0.061	0.069	<b>0.074</b>
	NDCG@5	0.044	0.038	0.037	0.036	0.025	0.037	0.040	<b>0.054</b>
	MRR	0.049	0.031	0.026	0.030	0.021	0.043	0.031	<b>0.053</b>
	HR@10	0.110	0.103	0.120	0.050	0.056	0.124	0.104	<b>0.127</b>
	NDCG@10	0.045	0.052	0.053	0.032	0.031	0.058	0.051	<b>0.064</b>
	MRR	0.047	0.037	0.040	0.027	0.023	0.038	0.035	<b>0.052</b>
Hetrec2011	HR@5	0.019	0.005	0.024	0.029	0.003	0.001	0.001	<b>0.032</b>
	NDCG@5	0.012	0.002	0.015	0.013	0.002	0.007	0.003	<b>0.021</b>
	MRR	0.016	0.001	0.020	0.018	0.001	0.005	0.006	<b>0.022</b>
	HR@10	0.035	0.009	0.034	0.037	0.007	0.025	0.009	<b>0.041</b>
	NDCG@10	0.018	0.004	0.022	0.023	0.003	0.011	0.004	<b>0.026</b>
	MRR	0.018	0.002	0.021	0.021	0.002	0.007	0.001	<b>0.023</b>

<https://doi.org/10.1371/journal.pone.0272319.t003>



(a) HR@5 and MRR



(b) HR@10 and MRR

**Fig 4. Histograms comparing the model with other baselines in the case of the top- $k$  ( $k=5,10$ ) recommendations, respectively.** From the figure, it is clear that the model introduced in this paper achieves the best performance on both datasets.

<https://doi.org/10.1371/journal.pone.0272319.g004>

Table 4. HR@k and MRR(k = 5,10) of this model with different aggregation methods on two dataset(Tourism Website and Hetrec2011).

	Tourism Website				Hetrec2011			
	HR@5	MRR@5	HR@10	MRR@10	HR@5	MRR@5	HR@10	MRR@10
<i>Min</i>	<b>0.074</b>	<b>0.053</b>	<b>0.127</b>	<b>0.052</b>	<b>0.032</b>	<b>0.022</b>	<b>0.041</b>	<b>0.023</b>
<i>Mean</i>	0.065	0.053	0.118	0.051	0.025	0.018	0.037	0.019
<i>Sum</i>	0.034	0.028	0.096	0.033	0.023	0.015	0.035	0.020
<i>Max</i>	0.067	0.047	0.122	0.050	0.025	0.015	0.037	0.018

<https://doi.org/10.1371/journal.pone.0272319.t004>

and 4(b) illustrate histograms comparing the model introduced in this paper with other baselines in the case of the top- $k$  ( $k=5,10$ ) recommendations, respectively. The graph shows that the model introduced in this work results in the highest performance on both datasets. In summary, the model presents in this paper largely outperforms all baselines, which clearly answers RQ1.

### Parametric analysis

In this section, the model is analyzed in depth and, designed to better recognize the actions of our model in response to RQ2.

**Effect of aggregation approach.** The representation of the user's short-term intent is obtained using four types of aggregation. The usability of these four aggregation methods is then discussed. Table 4 illustrates the results of the four different aggregation methods, with  $HR@k$  and  $MRR(k = 5,10)$  as the metric. It can be seen that "minimum" achieves satisfactory results for both datasets.

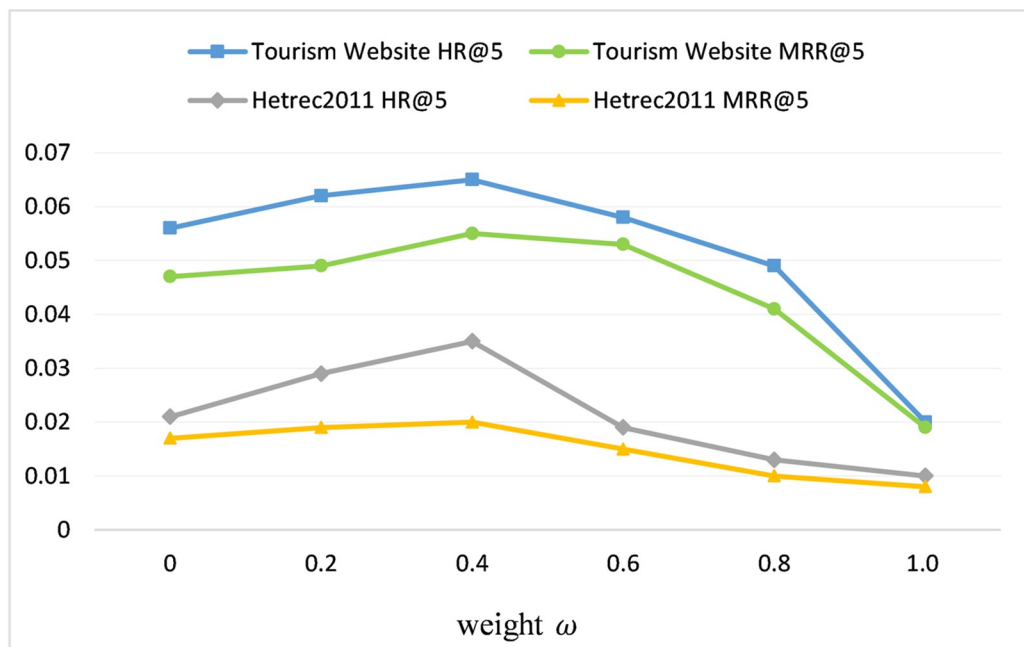
**Effect of weight  $\omega$ .** Fig 5(a) and 5(b) illustrates the results of setting different parameters  $\omega$  on the two datasets with  $HR@k$  and  $MRR(k=5,10)$  as measures. Parameter  $\omega$  manages the effects of the model for both the short- and long-term effects. From Fig 5, it is desirable to set the value of  $\omega$  at between 0.2 and 0.4, indicating that short-term intent is more important to a sequence recommendation.

**Effect of the number of dimensions  $d$ .** Fig 6(a) and 6(b) shows the results for different numbers of dimensions  $d$  on two datasets using  $HR@k$  and  $MRR(k=5,10)$  as a measure and keeping the other parameters the same. From the figure, it can be concluded that a larger dimensionality does not mean a higher performance, considering the overfitting problem.

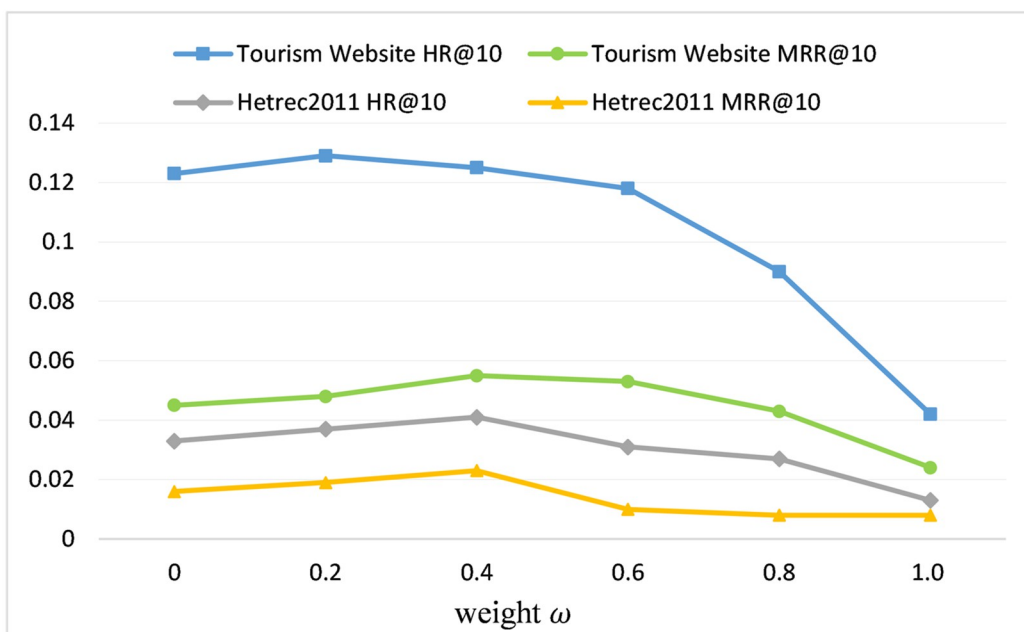
**Effect of modelling the item relationships.** Fig 7(a) and 7(b) show the impact of item relationship modelling on the model presented in this paper on two different datasets. It can be concluded from the figure that the model introduced in this paper is more effective than the single model that models the user preferences, illustrating the importance of the item relationship modelling. Modelling the item relationships is more helpful in analysing user preferences.

### Conclusion

In this paper, a new sequential recommendation method based on a self-attention mechanism is introduced. The model considers the short- and long-term intentions of the user, alongside the relationship between items to infer the user's next action. It utilizes self-attention to understand the user's short-term intentions from their most latest behavior and to model the item relationships. Experiments are conducted on both datasets, and the model proposed in this paper achieves optimal performance compared to some other baselines because both the long- and short-term preferences of users and the relationship between items are considered. The analysis indicates that our model accurately obtains the importance of the relationship



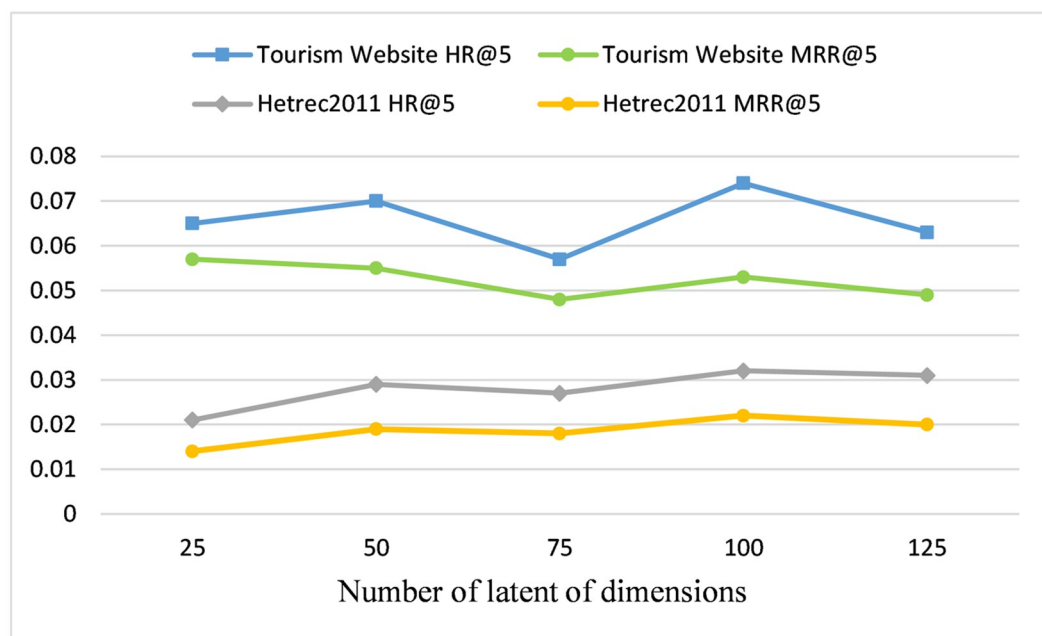
(a) HR@5 and MRR



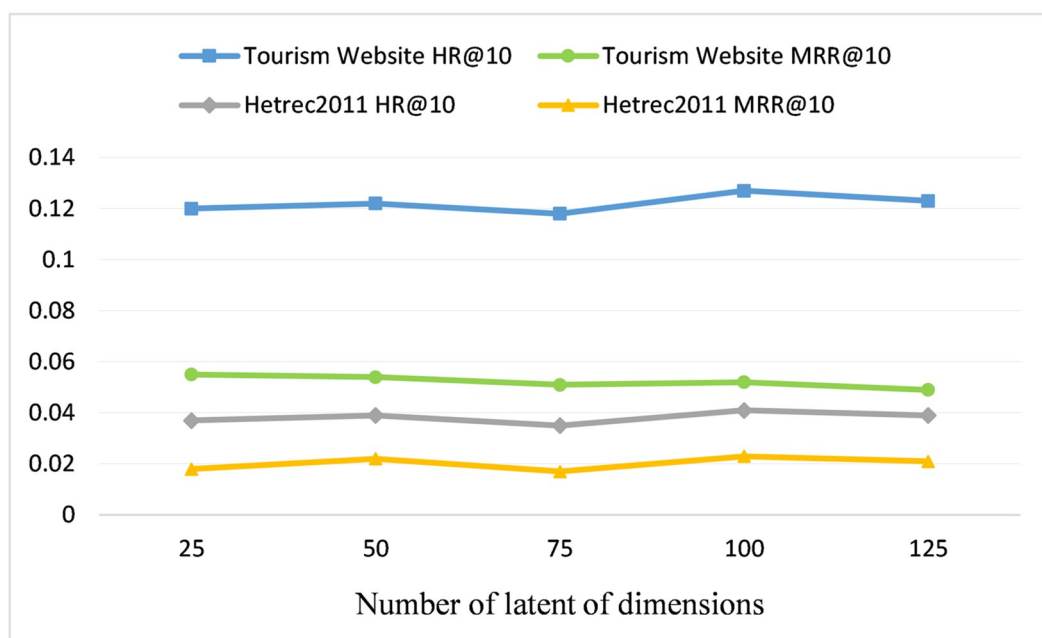
(b) HR@10 and MRR

**Fig 5. Effects of the weight  $\omega$ .**

<https://doi.org/10.1371/journal.pone.0272319.g005>



(a) HR@5 and MRR



(b) HR@10 and MRR

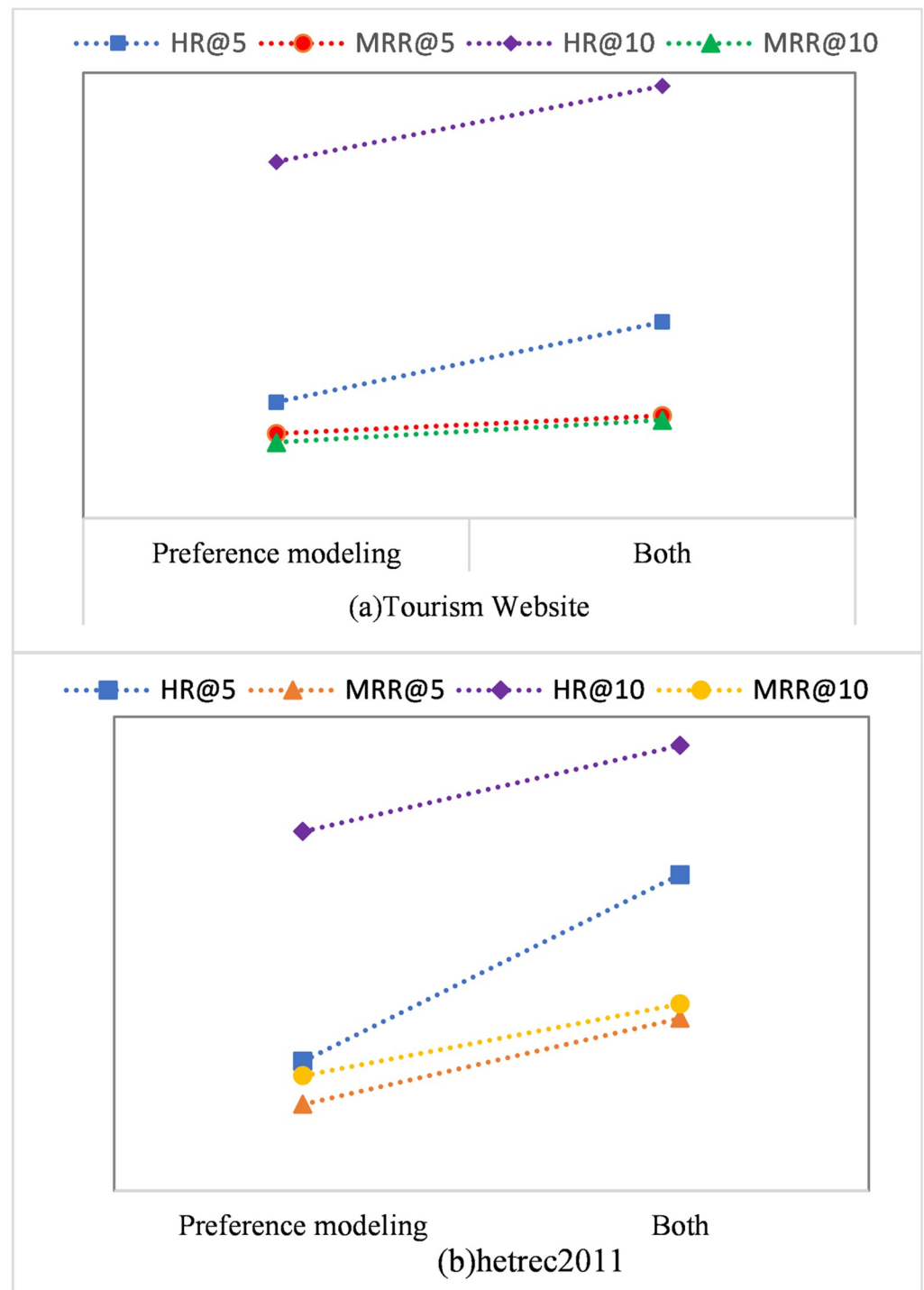
**Fig 6. Effects of number of latent dimensions  $d$ .**

<https://doi.org/10.1371/journal.pone.0272319.g006>

between user behavior and items. In addition, it is effective to extend the self-attention to a sequence recommendation method.

In the future, more work tends to include more additional information to further improve the accuracy of the recommendation, such as studying the time information of the user's





**Fig 7.** The effects of item relationship modelling on two different datasets.  $MRR@k$  is the  $MRR$  that predicts the next  $k$  items.

<https://doi.org/10.1371/journal.pone.0272319.g007>

evaluation of the item, and hopefully more knowledge will be investigated to enhance this model. This work is also applicable to other relevant sequence recommendation tasks.

## Supporting information

### S1 Data. Data used in this model.

(RAR)

## Author Contributions

**Conceptualization:** Guangjie Liu, Xin Ma, Jinlong Zhu, Yu Zhang, Danyang Yang, Jianfeng Wang, Yi Wang.

## References

1. Dhelim S, Aung N, Bouras M A. A Survey on Personality-Aware Recommendation Systems. *Artificial Intelligence Review*. 2021, 55(3): 2409–2454. <https://doi.org/10.1007/s10462-021-10063-7>
2. Yuanzhe Peng. A Survey on Modern Recommendation System based on Big Data. *Clinical orthopaedics and related research*. 2022,abs/2206.02631.
3. Li G, Hua J, Yuan T, Wu J, Jiang Z, Zhang H, et al. Novel Recommendation System for Tourist Spots Based on Hierarchical Sampling Statistics and SVD++. *Mathematical Problems in Engineering*. 2019;1–15. <https://doi.org/10.1155/2019/6329746>
4. Smirnov A, Kashevnik A, Ponomarev A, Shilov N, Teslya N. Recommendation system for tourist attraction information service. 14th Conference of Open Innovation Association FRUCT; 2013 Nov 11–15; Ember Espoo,Finland; 2013. pp. 148–155.
5. Balaji T K, Annavarapu C, Bablani A. Machine learning algorithms for social media analysis: A survey. *Computer Science Review*. 2021, 40(12):100395.
6. Covington P, Adams J, Sargin E. Deep neural networks for youtube recommendations. *Proceedings of the 10th ACM Conference on Recommender Systems*; 2016 Sep 15–19; Boston, MA, USA; 2016. pp. 191–198.
7. Eksombatchai C, Jindal P, Liu JZ, Liu Y, Sharma R, Sugnet C, et al. Pixie: A System for Recommending 3+ Billion Items to 200+ Million Users in Real-Time. *Proceedings of the 27th International Conference on World Wide Web*; 2018 Apr 23–27; Lyon, France; 2018. pp. 1775–1784.
8. He X, He Z, Song J, Liu Z, Jiang Y-G, Chua T-S. NAIS: Neural Attentive Item Similarity Model for Recommendation. *IEEE Transactions on Knowledge and Data Engineering*; 2018; 30,2354–2366.
9. Kabbur S, Ning X, Karypis G. FISM: factored item similarity models for top-N recommender systems. *Proceedings of the 19th ACM SIGKDD international conference on Knowledge discovery and data mining*; 2013 Aug 11–14; Chicago, IL, USA; 2013. pp. 659–667.
10. Sarwar B, Karypis G, Konstan J, Riedl J. Item-based collaborative filtering recommendation algorithms. *Proceedings of the 10th international conference on World Wide Web*; 2001 May 1–5; China, Hong Kong; 2001. pp. 285–295.
11. Wang X, He X, Wang M, Feng F, Chua TS. Neural Graph Collaborative Filtering. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*; 2019 Jul 21–25; Paris, France; 2019. pp. 165–174.
12. Xue F, He X, Wang X, Xu J, Liu K, Hong R. Deep Item-based Collaborative Filtering for Top-N Recommendation. *ACM Trans. Inf. Syst.* 2019; 37, 33:1–33:25. <https://doi.org/10.1145/3314578>
13. Xin, X., He, X., Zhang, Y., Zhang, Y. and Jose, J. Relational Collaborative Filtering: Modeling Multiple Item Relations for Recommendation. *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*; 2019 Jul 21–25; Paris, France; 2019. pp. 125–134.
14. Chen X, Xu H, Zhang Y, Tang J, Cao Y, Qin Z, et al. Sequential Recommendation with User Memory Networks. *Proceedings of the Eleventh ACM International Conference on Web Search and Data Mining*; 2018 Feb 5–9; Marina Del Rey, CA, USA; 2018. pp. 108–116.
15. He R, McAuley J. Fusing Similarity Models with Markov Chains for Sparse Sequential Recommendation. *IEEE 16th International Conference on Data Mining (ICDM)*; 2016 Dec 12–15; Barcelona, Spain; 2016. pp. 191–200.

16. Koren Y. Collaborative filtering with temporal dynamics. *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*; 2009 Jun 28- 1 Jul; Paris, France; 2009. pp. 447-456.
17. Quadrana, M., Jannach, D. and Cremonesi, P. Tutorial: Sequence-Aware Recommender Systems. *Companion of The 2019 World Wide Web Conference*; 2019 May 13-17; San Francisco, CA, USA; 2019. pp. 1316.
18. Rendle, S., Freudenthaler, C. and Schmidt-Thieme, L. Factorizing personalized markov chains for next-basket recommendation. *Proceedings of the 19th international conference on World wide web*; 2010 Apr 26-30; Raleigh, North Carolina, USA; 2010. pp. 811-820.
19. Wu CY, Ahmed A, Beutel A, Smola AJ, Jing H. Recurrent Recommender Networks. *Proceedings of the 10th ACM International Conference on Web Search and Data Mining*; 2017 Feb 6-10; Cambridge, United Kingdom; 2017. pp. 495-503.
20. He X, Liao L, Zhang H, Nie L, Hu X, Chua TS. Neural Collaborative Filtering. *Proceedings of the 26th International Conference on World Wide Web*; 2017 Apr 3-7; Perth, Australia; 2017. pp. 173-182.
21. Bordes A, Usunier N, Garcia-Duran A, Weston J, Yakhnenko O. Translating Embeddings for Modeling Multi-relational Data. *Advances in Neural Information Processing Systems*; 2013 Dec 5-8; Lake Tahoe, Nevada, United States; 2013. pp. 2787–2795.
22. He R, Kang WC, McAuley J. Translation-based Recommendation. *Proceedings of the 11th ACM Conference on Recommender Systems*; 2017 Aug 27-31; Como, Italy; 2017. pp. 161-169.
23. He X, He Z, Du X, Chua TS. Adversarial Personalized Ranking for Recommendation. *Proceedings of The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*; 2018 Jul 8-12; Ann Arbor, MI, USA; 2018. pp. 355-364.
24. Iscen A, Tolias G, Avrithis Y, Chum O. Mining on Manifolds: Metric Learning without Labels. 2018 Jun 18-22; Salt Lake City, UT, USA; 2018. pp. 7642-7651.
25. Fathollahi M S, Razzazi F. Music similarity measurement and recommendation system using convolutional neural networks. *International Journal of Multimedia Information Retrieval*. 2021; 5:1–11.
26. Hidasi B, Karatzoglou A, Baltrunas L, Tikk DJ. Session-based Recommendations with Recurrent Neural Networks. 4th International Conference on Learning Representations; 2016 May 2-4; San Juan, Puerto Rico; abs/1511.06939.
27. Jannach D, Ludewig M. (2017) When Recurrent Neural Networks meet the Neighborhood for Session-Based Recommendation. *Proceedings of the Eleventh ACM Conference on Recommender Systems*; 2017 Aug 27-31; Como, Italy; 2017. pp. 306-310.
28. Zhang L, Wang P, Li J. Attentive Hybrid Recurrent Neural Networks for sequential recommendation. *Neural Computing and Applications*. 2021; 33(17):11091–11105. <https://doi.org/10.1007/s00521-020-05643-7>
29. Zhang S, Tay Y, Yao L, Sun A. Next item recommendation with self-attention. *Computing Research Repository*. 2018; abs/1808.06414.
30. Zhong S T, Huang L, Wang C D. An Autoencoder Framework With Attention Mechanism for Cross-Domain Recommendation. *IEEE Transactions on Cybernetics*. 2020; 99:1–13.
31. Feng W, Li T, Yu H, Yan Z. A Hybrid Music Recommendation Algorithm Based on Attention Mechanism. *MICROPOROUS AND MESOPOROUS MATERIALS*. 2021;328–339.
32. Xu H, Ding Y, Sun J, Zhao K, Chen Y. Dynamic Group Recommendation Based on the Attention Mechanism. *Future Internet*. 2019; 11(9):198. <https://doi.org/10.3390/fi11090198>
33. Guo M, Xu T, Liu J, Liu Z, Jiang P, Mu T, et al. Attention Mechanisms in Computer Vision: A Survey. *Computational Visual Media*. 2022; 8(3): 331–368. <https://doi.org/10.1007/s41095-022-0271-y>
34. Xu K, Ba JL, Kiros R, Cho K, Courville A, Salakhutdinov R, et al. Show, attend and tell: neural image caption generation with visual attention. *Proceedings of the 32nd International Conference on Machine Learning*; 2015 Jul 6-11; Lille, France; 2015. pp. 2048-2057.
35. Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate. *International Conference on Learning Representations*; 2014 May 7-9; San Diego, CA, USA; arXiv:1409.0473.
36. Woo S, Park J, Lee JY, Kweon IS. CBAM: Convolutional Block Attention Module. 15th European Conference on Computer Vision; 2018 Sep 8-14; Munich, Germany; 2018. pp. 3-19.
37. Li J, Ren P, Chen Z, Ren Z, Lian T, Ma J. Neural Attentive Session-based Recommendation. *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management*; 2017 Nov 6-10; Singapore; 2017. pp. 1419-1428.

38. Feng Y, Lv F, Shen W, Wang M, Sun F, Zhu Y, et al. Deep Session Interest Network for Click-Through Rate Prediction. *Proceedings of the 28th International Joint Conference on Artificial Intelligence*; 2019 Aug 10-16; Macao, China; 2019. pp. 2301-2307.
39. Vaswani A, Shazeer N, Parmar N, Uszkoreit J, Jones L, Gomez AN, et al. Attention is all you need. *Proceedings of the 31st International Conference on Neural Information Processing Systems*; 2017 Dec 4-9; Long Beach, CA, USA; 2017. pp. 5998-6008.
40. Christakopoulou E, Karypis G. Local Latent Space Models for Top-N Recommendation. *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*; 2018 Aug 19-23; London, UK; 2018. pp. 1235-1243.
41. Lee J, Kim S, Lebanon G, Singer Y. Matrix approximation under local low-rank assumption. *International Conference on Learning Representations*; 2013 May 2-4; Scottsdale, Arizona, USA; arXiv: 1301.3192.
42. Liberti L, Lavor C, Maculan N, Mucherino A. Euclidean distance geometry and applications. *Quantitative Biology*. 2012; 56(1):3–69.
43. Yang B, Yih Wt, He X, Gao J, Deng L. Embedding entities and relations for learning and inference in knowledge bases. *International Conference on Learning Representations*; 2014 May 7-9; San Diego, CA, USA; arXiv: 1412.6575.
44. Li J, and Wang Y and McAuley J. Time Interval Aware Self-Attention for Sequential Recommendation. *web search and data mining*. 2020. pp.322–330. <https://doi.org/10.1145/3336191.3371786>
45. Xu C, Feng J, Zhao P, Zhuang F, Sheng VS. Long- and short-term self-attention network for sequential recommendation. *Neurocomputing*. 2021; 423(8):580–589. <https://doi.org/10.1016/j.neucom.2020.10.066>
46. Zhang, S., Yao, L., Tay, Y., Xu, X., Zhang, X. and Zhu, L. (2018) Metric factorization: Recommendation beyond matrix factorization. *International Conference on Learning Representations*; 2018 Apr 30-3 May; Vancouver, Canada; arXiv: 1802.04606.