Information Sciences 656 (2024) 119916

Contents lists available at ScienceDirect

Information Sciences

journal homepage: www.elsevier.com/locate/ins

Sequence recommendation using multi-level self-attention network with gated spiking neural P systems ☆

Xinzhu Bai [a], Yanping Huang [a], Hong Peng [a,*], Jun Wang [b], Qian Yang [a], David Orellana-Martín [c], Antonio Ramírez-de-Arellano [c], Mario J. Pérez-Jiménez [c]

[a] *School of Computer and Software Engineering, Xihua University, Chengdu, 610039, China*
[b] *School of Electrical Engineering and Electronic Information, Xihua University, Chengdu, 610039, China*
[c] *Research Group of Natural Computing, Department of Computer Science and Artificial Intelligence, University of Seville, Sevilla, 41012, Spain*

A R T I C L E I N F O

Keywords:
Sequence recommendation
Nonlinear spiking neural P systems
Gated spiking neural P systems
Self-attention mechanism

A B S T R A C T

Sequence recommendation is used to predict the user's next potentially interesting items and behaviors. It not only focuses on the user's independent interaction behavior, but also considers the user's historical behavior sequence. However, sequence recommendation still faces some challenges: the existing models still have shortcomings in addressing long-term dependencies and fully utilizing contextual information in sequence recommendation. To address these challenges, we propose a four-channel model based on a multi-level self-attention network with gated spiking neural P (GSNP) systems, termed SR-MAG model. The four channels are divided into two groups, and each group is composed of an attention channel and an GSNP attention channel. Moreover, they process long-term sequences and short-term sequences respectively to obtain long-term or short-term attention channel features. These features are then passed through a self-attention network to effectively extract user context information. The proposed SR-MAG model is tested on three real datasets and compared with 10 baseline methods. Experimental results demonstrate the effectiveness of the proposed SR-MAG model in sequence recommendation tasks.

1. Introduction

With the rise of the Internet economy, e-commerce platforms and personalized recommendations have gradually entered people's field of vision, among which sequential recommendations occupy an important position. Through sequential recommendation, recommender systems can better understand user interest evolution and behavior patterns, thereby providing a more personalized and accurate recommendation experience. This has important implications for recommendation applications in areas such as social media, music, and video streaming. Commonly used sequence recommendation methods include Markov [23] model, recurrent neural networks (RNN) [36] and convolutional neural networks (CNN) [13], and so on. These methods can model sequence dependencies and fuse contextual information, thus improving the accuracy and precision of sequence recommendation in personalized recommendation.

☆ This work was partially supported by the National Natural Science Foundation of China (No. 62076206, 62176216), the Research Fund of Sichuan Science and Technology Project (No. 2022ZYD0115), and the Zhejiang Lab BioBit Program (Grant No. 2022BCF05), China.

* Corresponding author.
E-mail address: ph.xhu@hotmail.com (H. Peng).

https://doi.org/10.1016/j.ins.2023.119916
Received 22 July 2023; Received in revised form 1 October 2023; Accepted 11 November 2023

Available online 21 November 2023
0020-0255/

In recent years, deep learning has had an important influence and role in the application of recommendation systems in natural language processing (NLP). Deep learning models have powerful representation learning and modeling capabilities in the NLP field. It can perform efficient, accurate feature extraction and association learning on data [18,24,30–32]. Moreover, in recommender systems, text data usually exists in the form of sequences, such as user historical behavior sequences, document word sequences, and so on. Deep learning models can model sequence data through structures such as recurrent neural networks and convolutional neural networks [12], and capture contextual information and long-term dependencies in the sequence. In addition, self-attention mechanisms have achieved satisfactory results in sequence recommendation, such as SASRec [11] and SHAN [34]. However, the existing self-attention methods determine the dependencies between different locations based on the correlation score, without considering their order information. Therefore, they are insufficient to distinguish user dynamics, changing long-term preferences and short-term interests [10].

1.1. Related work

Sequence recommendation is to use the user's historical behavior sequence to capture the user's interest evolution and behavior patterns. By analyzing the patterns and rules in the sequence, users' preferences, interest evolution trends and possible behavior patterns can be extracted. This information can be used in the personalized recommendation of the recommender systems, so that the systems can better adapt to the changes of users' interests and needs. Rendle et al. [23] discusses an early model for sequence recommendation, termed FPMC, which models user behavior sequences based on the idea of Markov chains. The FPMC not only considers the user's historical behavior, but also captures the transfer relationship between items. It can provide personalized sequence recommendations. But it has serious deficiencies in dealing with long-term dependencies, contextual information, and model representation capabilities.

In recent years, deep neural networks such as recurrent neural networks (RNN) and convolutional neural networks (CNN) have been widely used for sequence recommendation [27]. They model sequence data to capture contextual information and long-term dependencies in sequences, thereby improving the personalization and timing of the entire recommendation. Hochreiter et al. [6] proposed a long short-term memory network (LSTM), which introduces memory units and gating mechanisms on the basis of recurrent neural networks to effectively manage long-term dependencies in sequences. Cho et al. [3] proposed the gated recurrent unit (GRU). It is simpler than the LSTM model. And it uses two gates to control the input and output of the hidden state at the current moment. Vaswani et al. [26] proposed a neural network architecture called "Transformer", which is a sequence modeling that relies entirely on the self-attention mechanism. Transformer can better handle long-term dependencies in sequence data and achieve efficient parallel computing. However, there are still some problems with deep learning in sequence recommendation: the demand for large amounts of data, lack of interpretability, complex parameter tuning, and high demands on computing resources.

Recently, attention mechanisms have been used in recommender systems, which can adaptively learn the importance weights of different elements in a sequence, thereby improving the accuracy and personalization of recommendations. Ying et al. [35] designed hierarchical attention networks with long-term and short-term preferences. The user-item and item-item interactions are considered in the prediction list, and finally the recommendation results are given. Of course, there is also a method of combining the attention mechanism with RNN and CNN to solve the above problems. Tan et al. [25] proposed a bidirectional GRU neural network, which uses attention mechanism to model general interest and short-term consumption motivation, and then combines it with GRU to predict the user's next item. Kang et al. [11] proposed a self-attention-based sequential recommendation method (SASRec), which outperforms methods such as recurrent neural networks. Xu et al. [29] proposed a long short-term self-attention network (LSSA). This method divides user behavior sequences into long and short-term sequences, and performs self-attention network modeling separately.

Although the above-mentioned sequence prediction methods have been investigated, sequence recommendation still faces the following challenges: (i) how to solve the long-term dependency of sequences to achieve accurate recommendation; (ii) how to make full use of context information to predict and generate user behavior sequences.

1.2. Motivation and contribution

Spiking neural P (SNP) systems [9] are a class of neural-like computing models, abstracted from the mechanisms of spiking neurons. Nonlinear spiking neural P (NSNP) systems [22] are nonlinear variants of SNP systems with nonlinear processing capabilities. The nonlinear spiking mechanisms in NSNP systems have been used to develop several deep learning models, including long short-term memory model inspired from spiking neural P systems (LSTM-SNP) [16], gated spiking neural P systems (GSNP) [15], nonlinear spiking neural systems with autapses (NSNP-AU) [17], echo spiking neural P systems (ESNP) [21], and convolutional-type spiking neural P model (ConvSNP) [40]. These deep learning models have been used in sentiment analysis [7,8,14], medical image processing [33], time series prediction [19,20,38], information fusion [37] and classification problems [39].

To address the challenges in sequence recommendation, we propose a new sequence recommendation method based on a multi-level self-attention network with gated spiking neural P systems, termed SR-MAG model. The SR-MAG model can solve the long-term dependence problem in sequence recommendation and make full use of contextual information of user historical behavior. From a structural point of view, the SR-MAG adopts a four-channel structure, which considers the characteristics of the user's current interest and long-term interest at the same time, and uses the GSNP model and multi-level self-attention mechanisms to capture the characteristics of long-term sequences and short-term sequences.

Then, the long-term and short-term features are fed into a multi-level self-attention network to fuse these hybrid features. As a result, it not only enhances the ability to extract contextual information for sequence recommendation but also effectively alleviates the long-term dependencies in sequence recommendation. GSNP model is a variant of recurrent neural networks (RNN), inspired by nonlinear spiking mechanisms in NSNP systems. The purpose of GSNP model is to extract the features of long-term sequences. In the sequence recommendation, the self-attention mechanism can capture the key features in the long-term and short-term sequences, thereby improving the accuracy of the recommendation. Moreover, the proposed SR-MAG can perform contextual modeling on user historical behavior and effectively capture contextual information. This enables our SR-MAG model to provide more personalized and accurate recommendation results.

The contributions of this paper can be summarized as follows.

(1) We propose a novel sequence recommendation model, which uses a multi-level self-attention network with gated spiking neural P systems, termed SR-MAG. The SR-MAG integrates the GSNP model and multi-level self-attention mechanisms to capture the context information of the user's historical behavior sequence, and effectively alleviate the long-term dependence problem of sequence recommendation.
(2) The SR-MAG achieves a multi-channel long-term and short-term feature extraction. By designing a four-channel structure, the long-term sequence features are fused with the short-term sequence features to obtain mixed features. Then the self-attention mechanism is used to re-extract its features, and realize the dynamic extraction of user interests, thereby improving the accuracy of recommendation.
(3) We conduct extensive experiments on three real-world datasets. Experimental results demonstrate the advantage of the proposed SR-MAG model in achieving sequential recommendation tasks.

The remainder of this paper is organized as follows. Section 2 describes the proposed sequence recommendation model in detail. The experimental results and analysis are provided in Section 3. Finally, the conclusions are drawn in Section 4.

2. Proposed model

2.1. Model framework

Suppose the user set is $U = \{u_1, u_2, \dots, u_{|U|}\}$, where $|U|$ is the total number of users, and the item set is $I = \{i_1, i_2, \dots, i_{|I|}\}$, where $|I|$ is the total number of items. The entire historical behavior sequence is divided into multiple subsequences $H_u = \{S_1^u, S_2^u, \dots, S_t^u\}$ according to the time span, where u represents the user, and t is the total number of subsequences. In this study, the purpose of sequential recommendation is to predict the next item S_{t+1}^u for each user u.

In this study, we propose a four-channel network based on GSNP model and multi-level self-attention mechanisms for fusing long-term and short-term sequence, termed SR-MAG model. The network utilizes the GSNP model to solve the long-term dependency problem of sequences and efficiently extract contextual information. Moreover, the multi-level self-attention mechanism can capture the key features of long-term and short-term sequences by modeling the dependencies between different positions in the sequence.

Fig. 1 shows the proposed model framework, which consists of embedding layer, GSNP layer, multi-head attention layer and prediction layer. From a structural point of view, the proposed model has four channels and is divided into two groups. The first group is long-term attention channel and long-term GSNP attention channel, while the second group is short-term attention channel and short-term GSNP attention channel.

We divide the user historical behavior sequence into long-term sequence $L_t^u = \left(S_1^u \cup S_2^u \cup \cdots \cup S_t^u\right)$ and short-term sequence $S_t^u = \left(i_1, i_2, \dots, i_{|S_t^u|}\right)$. The long-term sequences are input to the long-term attention channel and the long-term GSNP attention channel in the first group, and the short-term sequences are input to the short-term attention channel and the short-term GSNP attention channel in the second group. The long- and short-term attention channels are to capture the long-term and short-term self-attention features of each user time step, and obtain the user's long-term interest $F_m^{(L)}$ and short-term interest $F_n^{(S)}$, respectively. The long-term and short-term GSNP self-attentions are used to capture the long-term and short-term self-attention features of user context information: $F_m^{(G)}$ and $F_n^{(G)}$. Then, the long-term self-attention channel feature $F_m^{(L)}$ is fused with the long-term GSNP attention channel feature $F_m^{(G)}$ to obtain the fusion feature $F_m^{(LG)}$ of long-term interest. At the same time, the short-term self-attention channel feature $F_n^{(S)}$ is fused with the short-term GSNP attention channel feature $F_n^{(G)}$ to obtain the fusion feature $F_m^{(SG)}$ for short-term interest. Subsequently, the two fused features are input into a self-attention network to dynamically capture the mixed interests $F_m^{(Fl)}$ and $F_m^{(Fs)}$. Thus, the effective extraction of user context information is realized. Finally, we concatenate the results to obtain the final user prediction representation.

We train the parameters of the proposed SR-MAG model by minimizing the joint BPR loss function. In the following, each component and channel of the proposed SR-MAG model are described in detail.

In summary, the proposed SR-MAG model is four-channel structure, where two channels are used to handle long-term sequence, while other two channels are used to short-term sequence. In the four channels, the first and third channels utilize contextual information from long-term and short-term sequences, respectively, while the second and fourth channels use GSNP models to extract long-term and short-term dependencies from long-term and short-term sequences, respectively. Therefore, this four channel structure and the use of GSNP are significant differences between the proposed SR-MAG model and existing sequence recommendation models. This design can effectively address the challenges in existing sequence recommendation models.

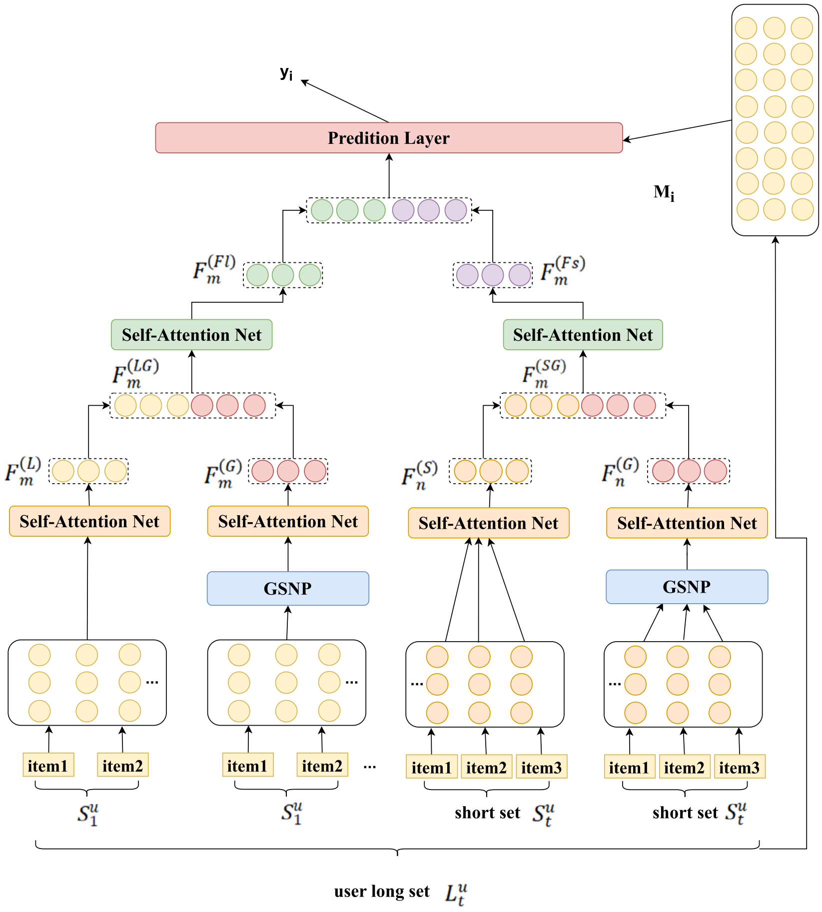

Fig. 1. The framework of the proposed SR-MAG model for sequence recommendation.

2.2. Embedding layer

We encode the item ID of a one-hot representation into a continuous low-dimensional vector space. Typically $M \in R^{|I| \times d}$, which defines a d-dimensional (dimensionality of the space) embedding lookup table, where $|I|$ denotes the number of unique items in the dataset. In this model, we set the maximum lengths of the processing itemsets of the long-term sequence $L_t^u = (i_1, i_2, \ldots, i_m)$ and the short-term sequence $S_t^u = (i_1, i_2, \ldots, i_n)$ to m and n, respectively. In our experiments, we set $m = 200$ and $n = 50$ on three real datasets. If there are more than m behaviors in the long-term sequence, we still select the first m behaviors. Similarly, for a sequence with less than m behaviors, the left side is padded with a zero padding value. Similarly, the same method is used for short-term series.

Finally, the separated sequences are applied to the embedding layer, which transforms them into item embedding matrices E^l and E^s, respectively.

2.3. GSNP model

Nonlinear spiking neural P (NSNP) systems [22] are neural-like computing models, inspired by the nonlinear mechanism of spiking neurons. The nonlinear spiking mechanism is a recognizable feature of NSNP systems, which distinguishes them from other neural-like computing models.

In our a previous work [15], a variant of gated recurrent unit (GRU) has been developed for time series forecasting, called gated spiking neural P systems or termed GSNP model. The GSNP model is inspired by the nonlinear spiking mechanism in NSNP systems. Based on the nonlinear spiking mechanism, the GSNP model introduces two gates: reset gate and output gate. The reset gate is used to control the spike generation according the input at the current time and the output at the previous time. The output gate is used to control the neuron's output. The two nonlinear gates are defined by

$$o_t = \sigma(W_o \cdot x_t + U_o \cdot h_{t-1} + b_o) \tag{1}$$

$$r_t = \sigma(W_r \cdot x_t + U_r \cdot h_{t-1} + b_r) \tag{2}$$

The spike generation and output for the neuron can be computed as follows:

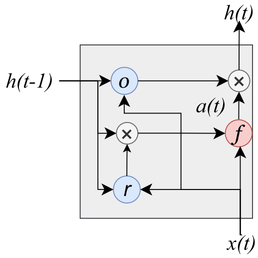

Fig. 2. Functional structure of neuron in GSNP model.

$$a_t = \tanh(W_a \cdot x_t + U_a \cdot (r_t \odot h_{t-1}) + b_a) \tag{3}$$

$$h_t = o_t \odot a_t \tag{4}$$

where symbol "$\odot$" denotes the inner product of two vectors; $W_r, U_r, b_r, W_o, U_o, b_o, W_a, U_a, b_a$ are training parameters that can be modified; σ is the sigmoid function.

Fig. 2 shows the functional structure of neuron in the GSNP model. In Fig. 2, symbols o, r and f denote output gate, reset gate and activation function, respectively. Eqs. (1)-(4) are the mathematical model of the GSNP model, which is very different from the existing LSTM and GRU. The GSNP model can exhibit the different dynamics from LSTM and GRU due to differences in mathematical models.

2.4. Multi-head self-attention layer

Self-attention is a mechanism in attention networks, which has global relevance and adaptability. It can capture the information of all positions in the sequence and improve the context modeling ability of the model. Moreover, it is adaptive, automatically adjusting attention weights according to different inputs. The self-attention used consists of three parts: multi-head attention, feedforward network, and multi-layer self-attention.

2.4.1. Multi-head attention

Multi-head attention was first used in neural machine translation [26] to model the semantics of entire sentences with great success. In our model, scaled dot-product attention is used. The multi-head attention can be described as follows:

$$Attention\left(\hat{Q}, \hat{K}, \hat{V}\right) = softmax\left(\frac{\hat{Q}\hat{K}^T}{\sqrt{d}}\right)\hat{V} \tag{5}$$

$$head_i = Attention\left(\hat{Q}_i, \hat{K}_i, \hat{V}_i\right) \tag{6}$$

$$H = Multihead\left(\hat{Q}, \hat{K}, \hat{V}\right) = Concat(head_i, ..., head_j) \tag{7}$$

where $\hat{Q}, \hat{K}, \hat{V} \in \mathbb{R}^{n\times d}$ and $\sqrt{d}$ are scaling factors used to avoid excessive inner product values, especially when d is too large.

In the proposed SR-MAG model, according to the different channels, the inputs of the attention mechanisms for the four channels are the following matrices or features:

(1) the embedding matrix of long-term sequence;
(2) the embedding matrix of short-term sequence;
(3) the output characteristics of GSNP neurons for long-term sequence;
(4) the output characteristics of GSNP neurons for short-term sequence.

The attention network takes the above matrices or features as inputs for different channels, and uses linear projection to transform them into three matrices representing $\hat{Q}$, $\hat{K}$ and $\hat{V}$. Then, the matrices of four different channels are input into different attention networks, and the weighted sum of similar values for queries and keys is calculated. Afterwards, the input of the attention network for long-term and short-term feature interest fusion is long-term fusion features and short-term fusion features. Similarly, they perform linear projection to obtain the corresponding query and key value pairs. Finally, the weighted sum of their situations is obtained.

In the experiment, we set the number of long-term and short-term self-attention heads to $i = 2$, and the number of GSNP self-attention heads to $i = 1$. Finally, the outputs are concatenated to obtain the total value.

2.4.2. Feedforward network

Although the previous use of scaling dot product attention to obtain the corresponding features of long-term and short-term sequences, it is still a linear model with weak expressive ability. To this end, we add a non-linear activation function *Relu* to the attention layer to endow nonlinear factors, thereby improving expressiveness:

$$F = Relu\left(HW_{f1} + b_{f1}\right)W_{f2} + b_{f2} \tag{8}$$

where W_{f_1} and W_{f_2} are weight matrices of size $d \times d$, and b_{f1} and b_{f2} are d-dimensional offset vectors. F will serve as the output of the single-layer multi-head attention mechanism, and also as the input of the next attention layer, thus capturing item-item interaction features.

2.4.3. Multi-layer self-attention

Although the self-attention network of the first layer already captures the adaptive weight embedding of historical items, different layers can capture different types of features [1]. Multi-layer attention networks can generate deeper feature representations by introducing multiple attention layers. Moreover, it has better local and global information fusion capabilities. Therefore, different layers can be described as follows:

$$F^j = SAN\left(F^{j-1}\right) \tag{9}$$

where j represents the number of layers of the self-attention network (SAN).

In this study, we set the number of long-term and short-term self-attention layers $j = 2$, and the number of GSNP self-attention layers $j = 1$. From Eq. (11), we can derive the output $F_m^{(L)}$ of the long-term self-attention channel and the output $F_n^{(S)}$ of the short-term self-attention channel. From Eqs. (4) and (11), we can obtain the output $F_m^{(G)}$, $F_n^{(G)}$ of the long-term and short-term GSNP self-attention channels. After that, we fuse the long-term features and short-term features respectively to obtain the long-term and short-term mixed feature values $F_m^{(LG)}$ and $F_m^{(SG)}$. To obtain more accurate general interest and current interest, we re-input the mixed feature value into a self-attention network layer to achieve more complex association learning and fusion of local or global information. Finally, we get the long-term and short-term fusion self-attention feature values $F_m^{(Fl)}$ and $F_n^{(Fs)}$ as follows:

$$F_m^{(LG)} = Concat(F_m^{(G)}, F_m^{(L)}) \tag{10}$$

$$F_m^{(SG)} = Concat(F_n^{(G)}, F_n^{(S)}) \tag{11}$$

$$F_m^{(Fl)} = SAN\left(F_m^{(LG)}\right) \tag{12}$$

$$F_m^{(Fs)} = SAN\left(F_m^{(SG)}\right) \tag{13}$$

2.5. Prediction layer

After the user's long-term and short-term sequence features captured by the above different self-attention channels, we achieve two fusions of long-term and short-term self-attention networks. Therefore, we effectively and accurately obtain the dynamic long-term and short-term feature interests of users. Moreover, they make full use of long-term and short-term contextual information to maximize feature extraction. We use a traditional latent factor model to compute the relevance of an item to the next item at time step t. The prediction scores and correlations are calculated as follows:

$$Re_{i,t}^{FL} = F_m^{(Fl)} M_i^T \tag{14}$$

$$Re_{i,t}^{FS} = F_m^{(Fs)} M_i^T \tag{15}$$

$$y_i = \left(F_m^{(Fl)} + F_m^{(Fs)}\right) M_i^T \tag{16}$$

where $Re_{i,t}^{FL}$ denotes the long-term prediction score, $Re_{i,t}^{FS}$ denotes the short-term prediction score, and y_i corresponds to the score of all candidate items i. As a result, we obtain a vector y of scores for all possible items. Finally, they are sorted in descending order, and the items with the top K values are selected as recommendations.

2.6. Model training

For the training of the model, the joint BPR optimization criterion is chosen to minimize the objective function. We train the model by minimizing the objective function. For each user, a positive sample (items the user likes) and a negative sample (items the user dislikes) are selected from their historical behavior, and the correct ranking is established by comparing their scores. The objective function is given as follows:

$$L = \sum_{i=2}^{n} \sum_{j \in H^u} -\ln \sigma\left(Re_{i,t} - Re_{j,t}\right) + \sum_{i=2}^{m} \sum_{\tilde{j} \in R^u} -\ln \sigma\left(\tilde{R}e_{\tilde{\imath},j} - \tilde{R}e_{\tilde{j},i}\right) + \lambda \|\Theta\|^2 \tag{17}$$

where Θ is the model parameter set, and $\sigma(\cdot)$ is the sigmoid function. While training, we optimize the network using $Adam$. $Adam$ is a variant of stochastic gradient descent (SGD), which can adaptively change the corresponding parameters.

Table 1
Statistics of three datasets.

DataSet	user	item	train session	test session	Avg. session length
Gowalla	5622	11364	29173	5622	4.63
ML-10M	7036	7287	82869	7036	21.08
Tmall	17617	48804	81637	17616	3.30

3. Experimental results and analysis

In this section, the performance of the proposed SR-MAG model on three real-world datasets is evaluated. These datasets are widely used in the field of recommender systems. Moreover, we also provide ablation experiments to verify the accuracy of long-term and short-term fusion and the effectiveness of core components.

3.1. Experimental configuration

3.1.1. Datasets

Three real-world datasets (Gowalla, Movielens-10M and Tmall) acquired from three different platforms are used to verify the performance of the proposed SR-MAG model. Gowalla[1] is a social networking and location sharing platform that includes location check-in data of users on the platform, that is, the user's check-in records (user ID, location latitude and longitude, check-in timestamp, etc.). Movielens-10M[2] is a data set about movie ratings, which contains user rating information for movies from IMDB (the Movie DataBase). The dataset consists of 71567 users. The experiment selects data from 2005 to 2009. Tmall[3] is an e-commerce dataset obtained in the 2015 IJCAI competition. It is the real transaction data on China's largest B2C e-commerce platform. The dataset includes user purchase behavior data, product information and user attributes.

Similar to the work of Xu et al. [29], we use the same method for the above datasets. The first is to remove items with less than 20 user interactions. Secondly, a user's one-day interaction project is regarded as a session, and it is regarded as a short-term performance. Then, we remove sessions that have less than three sessions in a day or where the user has fewer than three sessions. As in Ref. [4], at time step t, the processed long-term sequence $L_t^u = \{i_1^u, i_2^u, \ldots, i_t^u\}$ and short-term sequence $S_t^u = \{i_{t-|d|}^u, i_{t-|d|+1}^u, \ldots, i_t^u\}$ are respectively used as the user's long-term and short-term preferences, where $|d|$ is the length of a day's session. We select the most recent interaction of each user as the test set, and the rest of the data as the training set. The statistics of the final three datasets after processing are shown in Table 1.

3.1.2. Evaluation metrics

We evaluate the performance of all models using the following metrics, which have been widely used in the related work.

(i) HR@K: A recall-based indicator, which refers to the ratio of the number of users who have interacted with the total number. Generally, if the number of items in the recommendation list that overlaps with real related items is as high as possible, the index will be higher, which proves that the ability of the recommender system to retrieve related items is better.

$$HR@K = \frac{1}{|u|} \sum_u I(|S_u(K) \cap T_u|) \tag{18}$$

where S_u is the set of the top K items in the ranked list of all unique items, and T_u is the items in the test set that the user interacts with in group S; $I(x)$ is a tabulation function equal to 1 when $x > 0$ and 0 otherwise.

(ii) NDCG@K (Normalized discounted cumulative gain): it is a commonly used metric to measure the ranking accuracy of recommender systems. It comprehensively considers the relevance and ranking quality of items in the recommendation list. It will give higher weights to the top-ranked related items. Other words, the higher the metric, the better the ranking quality of the recommendation list.

$$NDCG@K = \frac{1}{Z} \sum_{j=1}^{K} \frac{2^{I\left(\left|s_u^j(K) \cap T_u\right|\right)} - 1}{\log_2(j+1)} \tag{19}$$

where $S_u^j(K)$ is the predicted score ranking jth in S_u, and Z is a normalization constant.

(iii) MAP: A metric for evaluating the ranking accuracy of recommender systems. It takes into account the average precision of each related item in the recommendation list, and average the accuracies. Generally, the range of MAP is between 0 and 1. The higher its value, the better the ranking accuracy of the recommendation list. Overall, it synthesizes the ranking quality of a model on different users, also considers the average accuracy of each user.

[1] http://snap.stanford.edu/data/loc-gowalla.html.

[2] https://grouplens.org/datasets/movielens/.

[3] https://ijcai-15.org/index.php/repeat-buyers-prediction-competition.

Table 2
Parameter settings.

Parameters	Parameter value
Learning rate	0.0001
Batch_size	128
Epochs	200
Dropout_rate	0.2
Hidden_units	200
Maxlen_long	200
Maxlen_short	50
Attention_lblocks	2
Attention_sblocks	2
Attention_lheads	2
Attention_sheads	2
Fusion_att_blocks	1
Fusion_att_heads	1
GSNPCells	32
GSNP_dropout	0.2

$$MAP = \frac{1}{K}\sum_{j=1}^{K} p(j) \times rel(j) \tag{20}$$

where $p(j)$ is the precision of the cutoff rank list from 1 to j. If item j is in the test set, $rel(j) = 1$, otherwise 0.

In the experiments, we choose $K = 10, 20$. At the same time, each metric is repeated 20 times, and then the average value is computed.

3.1.3. Parameter settings and implementation

According to the structure in Fig. 1, the proposed SR-MAG model[4] is implemented using Keras and Adam optimizer. The experimental results of the proposed SR-MAG model are obtained under a set of hyperparameters. This set of hyperparameters is determined by using a usual grid search method.

Table 2 shows this set of hyperparameters. In Table 2, "Maxlen_long" is the maximum length of a long-term sequence, and "Maxlen_short" is the maximum length of the short-term sequence; "Attention_lblocks" is the number of layers in the long-term attention network, and "Attention_sblocks" is the number of layers in the short-term attention network; "Attention_lheads" is the number of heads in the long-term attention network, and "Attention_sheets" is the number of heads in the short-term attention network; "Fusion_ att_blocks" is the number of layers in the fused attention network, and "Fusion_ att_heads" is the number of heads in the converged attention network.

3.2. Performance comparison

3.2.1. Comparative methods

To demonstrate the effectiveness and accuracy of the proposed SR-MAG model, different recommendation methods and different strategies are compared. These comparative methods are listed as follows:

(i) Pop: This model is the earliest to use a large amount of user behavior data and item metadata to generate recommendation results. Recommendations are mainly based on popularity, that is, the most popular and hottest items are recommended to users.
(ii) FPMC [23]: A personalized recommendation model based on Markov chain. It combines the ideas of matrix factorization and Markov chain modeling for predicting the user's next item in sequence recommendation. The model enables personalized recommendations by predicting the probability of a user's next item in a sequence.
(iii) TransRec [5]: A matrix completion method for recommender systems. It mainly addresses the sparsity and cold-start problems in recommender systems. By introducing the feature information of users and items into the model, it can make recommendations better in cold start situations.
(iv) RUM [2]: A user modeling model based on recurrent neural networks, which predicts user behavior by modeling users' long-term and short-term interests. It can better understand the user's interest evolution process, thereby improving the effect of recommendation.
(v) Att-BNN [25]: A bi-GRU neural network that pays more attention to users' long-term historical preferences and short-term consumption motivations for sequential recommendation.

[4] https://github.com/Baixinzhu/SRMAG.

Table 3
Comparison results of all models on the Gowalla dataset. The best and second best results in each column are bolded and underlined.

Method	HR@		NDCG@		MAP
	10	20	10	20	
Pop	0.0583	0.0822	0.0316	0.0377	0.0487
FPMC	0.1599	0.2200	0.0956	0.1107	0.0852
TransRec	0.2028	0.2778	0.1211	0.1399	0.1083
RUM	0.2387	0.3260	0.1316	0.1512	0.1123
Att-BNN	0.2623	0.3468	0.1450	0.1664	0.1219
LANCR	0.2058	0.2752	0.1212	0.1386	0.1067
SHAN	0.1613	0.2314	0.0905	0.1082	0.0802
SASRec	0.2327	0.3111	0.1316	0.1513	0.1125
LSSA	<u>0.2992</u>	<u>0.4035</u>	<u>0.1566</u>	<u>0.1828</u>	<u>0.1263</u>
SR-MAG	**0.3084**	**0.4037**	**0.1595**	**0.1835**	**0.1269**

Table 4
Comparison results of all models on the Movielens-10M dataset. The best and second best results in each column are bolded and underlined.

Method	HR@		NDCG@		MAP
	10	20	10	20	
Pop	0.0079	0.0166	0.0039	0.0071	0.0064
FPMC	0.1912	0.2741	0.0942	0.1142	0.0734
TransRec	0.1086	0.1612	0.0565	0.0697	0.0574
RUM	0.1295	0.2269	0.0768	0.0842	0.0621
SHAN	0.0423	0.0763	0.0221	0.0387	0.0239
SASRec	0.2057	0.3023	<u>0.1016</u>	<u>0.1269</u>	<u>0.0857</u>
HAFLS	**0.2124**	**0.3249**	**0.1029**	**0.1281**	**0.0884**
SR-MAG	<u>0.2084</u>	<u>0.3068</u>	0.0917	0.1165	0.0708

Table 5
Comparison results of all models on the Tmall dataset. The best and second best results in each column are bolded and underlined.

Method	HR@		NDCG@		MAP
	10	20	10	20	
Pop	0.0124	0.0321	0.0063	0.0119	0.0098
FPMC	0.0465	0.0813	0.0242	0.0356	0.0192
TransRec	0.0596	0.0872	0.0283	0.0415	0.0218
RUM	0.0695	0.1057	0.0314	0.0423	0.0287
SHAN	0.0783	0.1153	0.0361	0.0487	0.0305
SASRec	0.0776	0.1064	0.0387	0.0461	0.0332
HAFLS	<u>0.0844</u>	<u>0.1259</u>	<u>0.0427</u>	<u>0.0529</u>	<u>0.0376</u>
SR-MAG	**0.1562**	**0.2051**	**0.0750**	**0.0874**	**0.0573**

(vi) LANCR [28]: An attention-based recommendation model that combines matrix factorization and attention to predict the user's next recommendation. This model can more accurately capture user interest and item characteristics. Also it can improve the personalization of recommendations.

(vii) SHAN [34]: A two-layer hierarchical attention network model, where the first layer learns user long-term preferences with historical purchase item representations, and the second layer outputs end-user representations by coupling user long-term and short-term preferences.

(viii) SASRec [11]: A sequential recommendation model with self-attention mechanism. It can effectively process long sequences and capture contextual information of user behavior, so as to provide more accurate and personalized recommendation results.

(ix) LSSA [29]: A sequence recommendation model based on long-term sequence and short-term sequence self-attention mechanism. It can accurately predict users' general interests and short-term interests.

(x) HAFLS [4]: A model that captures long-term and short-term sequences with a hierarchical attention structure that can learn user preference features and dynamic transitions.

3.2.2. Results and discussion

To verify the overall performance and effectiveness of the proposed SR-MAG model, we compare it with the aforementioned baseline methods on three real datasets. The experimental results of all models on the three datasets are shown in Tables 3–5.

From Table 3, we can see that on the gowalla dataset, the proposed SR-MAG model outperforms all compared models on all metrics. This shows the effectiveness of long-term and short-term self-attention fusion in the proposed SR-MAG model, and also confirms the effectiveness of GSNP on long-term interest.

For the Movielens dataset (in Table 4), the proposed SR-MAG model achieves the second-best performance on the HR@(K = 10, 20) metrics, the third-best performance on the NDCG@(K = 10), and the fourth-best performance on both NDCG@ (K = 20) metrics and MAP metrics. On the Movielens dataset, the proposed SR-MAG model is slightly lower than the HAFLS model. After analysis, it is found that the reason for this result is the large sparsity of the Movielens dataset: many user-movie ratings are not observed. This indicates that when there is large sparsity in the dataset, this sparsity limits the performance improvement of the proposed RS-MAG model.

From Table 5, we can observe that on the Tmall dataset, the proposed SR-MAG model outperforms all baseline models in all three metrics. Compared with the second-best baseline model HAFLS, the proposed SR-MAG model significantly improves the performance, where HR@K improves by more than 75.64%, NDCG@K improves by more than 63.46%, and MAP improves by 52.39%. This shows that the proposed RS-MAG model has a great advantage in dealing with long-term data sets, which is more conducive to the prediction of long-term interest.

Overall, compared with the best baselines LSSA and HAFLS, the reason why the proposed SR-MAG model is better than them is that the channel combining GSNP and attention network is added to make it more accurately capture long-term interest. Secondly, a self-attention network is added to the final fusion of long-term and short-term interests, so that it can make full use of contextual information and capture long-term and short-term preferences to accurately predict the next item.

In sequence recommendation, long-term sequences usually refer to the historical behavior data of users, including user behavior records over a long period of time. It can be used to identify users' long-term interests and trends, such as purchasing many science fiction novels in the past few months, which can be considered as a long-term interest of the user. Short term interest refers to the user's recent behavioral data. In our model, the user's recent day's behavior is represented as a session (i.e., the user's short-term preferences).

Long term sequences contain the behavioral history of users over a long period of time, including data that spans multiple months or even years. Short term sequences typically focus on the user's recent behavior, typically including data from the user's recent hours, days, or weeks. In the proposed SR-MAG model, we use the user's last day to represent their short-term interest.

Long term sequences are usually used to establish a user's long-term interest model, such as understanding the user's music taste, movie preferences, or shopping preferences. It is suitable for scenarios that require a deep understanding of users' long-term behavior. Short term sequences are mainly used to respond to users' immediate needs, such as real-time new recommendations, online advertising placement, or social media updates. It is more suitable for scenarios that require timely feedback and instantaneous decision-making.

For existing methods, they make different treatments for different sequence lengths. For long sequences, they use models such as LSTM and GRU to capture long time dependencies and complex sequence patterns. For short sequences, they often use simple models, such as content-based recommendations or matrix decomposition methods. However, the proposed SR-MAG models the long and short term sequences separately, and then weights and fuses their results, which greatly improves the accuracy of sequence recommendation.

3.3. The impact of model components

To evaluate the contribution of each component in model training and prediction, we conduct ablation experiments. The results are shown in Table 6. Here No_FuAtt means that only the self-attention channel and GSNP self-attention channel are considered to capture the long-term and short-term interests of users, and there is no modeling of fusion self-attention for long-term and short-term sequences. Similarly, No_FAGA means to remove the modeling of GSNP self-attention channel and final self-attention fusion. It can be observed that SR-MAG (full type) performs best on both Gowalla and Movielens datasets. Among them, No_FuAtt is lower than the SR-MAG model, which confirms the effectiveness of the fusion attention mechanism, indicating that the fusion self-attention mechanism can establish correlations in long-term and short-term sequences, and understand important features in sequences more comprehensively. Similarly, the overall performance of No_FAGA is lower than that of the SR-MAG model, indicating that the fusion of GSNP and self-attention channels can effectively obtain the context information of long-term and short-term sequences. Overall, through these experiments, the importance of the fusion of self-attention mechanism and GSNP is proved.

3.4. The impact of superparameters

In this section, we also investigate the effect of different numbers of fusion self-attention blocks on modeling users' overall preferences. Since different layers of self-attention blocks can model complex correlations, extract multi-granularity features and capture long-distance dependencies in sequences, we set FSAN_blocks = 1,2,3,4 to indicate the number of self-attention blocks on the long-short-term fusion sequence, and use grid search to find their best combination. Table 7 shows the MAP results of the proposed SR-MAG model on the Gowalla and Movielens datasets. From the experimental results, the performance is the best when the fusion self-attention block is 1. Therefore, it can be concluded that the low-layer self-attention network can better capture the long-term and short-term dynamic characteristics of users.

Table 6
Model ablation experiment results.

Gowalla					
Model	MAP	HR@10	HR@20	NDCG@10	NDCG@20
SR-MAG	**0.1269**	**0.3084**	**0.4037**	**0.1595**	**0.1835**
No_FuAtt	0.1188	0.2593	0.3541	0.1408	0.1647
No_FAGA	0.1197	0.2710	0.3630	0.1447	0.1678
Movielens-10M					
Model	MAP	HR@10	HR@20	NDCG@10	NDCG@20
SR-MAG	**0.0708**	**0.2084**	**0.3068**	**0.0917**	**0.1165**
No_FuAtt	0.0622	0.1877	0.2878	0.0798	0.1050
No_FAGA	0.0499	0.1122	0.1709	0.0561	0.0708

Table 7
The effect of the number of fused self-attention blocks on MAP results.

Dataset	FSAN-Block			
	1	2	3	4
gowalla	**0.1269**	0.0864	0.1205	0.0912
movielens	**0.0708**	0.0593	0.0705	0.0519

4. Conclusion

In this paper, we propose a sequence recommendation model, termed SR-MAG model. The proposed SR-MAG model uses a four-channel structure, which integrates the self-attention network and the GSNP model. The four attention channels are long-term sequential attention channel, short-term sequential attention channel, long-term GSNP attention channel and short-term GSNP attention channel. The GSNP model is a recurrent neural network model inspired by nonlinear spiking mechanisms in NSNP systems. By introducing the GSNP model into sequential recommendation, we are able to exploit the contextual information of user behavior and effectively alleviate the long-term dependency of sequential recommendation. Moreover, our proposed self-attention feature fusion method fuses general interest features and current interest features into the self-attention network to capture more accurate mixed features, so as to adaptively obtain accurate recommendation of item-item association information. Experimental results on three real datasets show that the proposed SR-MAG model achieves the best recommendation performance on two datasets and slightly better performance on another dataset.

The proposed SR-MAG model adopts a four channel structure and uses the GSNP model to handle long-term and short-term dependencies in the sequence to address the challenges in sequence recommendation. However, we also noticed that the proposed SR-MAG model on this dataset did not achieve the best performance. The reason for this result is the large sparsity of this dataset. This is a limitation of the proposed SR-MAG model. How to overcome this limitation and better handle this sparsity problem is a direction for our future consideration.

CRediT authorship contribution statement

(1) Xinzhu Bai: Conceptualization, Software and experiments, Writing
(2) Yanping Huang: Conceptualization, Software and experiments
(3) Hong Peng: Conceptualization, Writing
(4) Jun Wang: Conceptualization, Writing
(5) Qian Yang: Conceptualization, Experiments
(6) David Orellana-Martín: Conceptualization, language
(7) Antonio Ramírez-de-Arellano: Conceptualization, language
(8) Mario J. Pérez-Jiménez: Conceptualization

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

The data that has been used is confidential.

References

[1] A. Anastasopoulos, D. Chiang, Tied multitask learning for neural speech translation, arXiv preprint, arXiv:1802.06655, 2018.

[2] X. Chen, H. Xu, Y. Zhang, J. Tang, Y. Cao, Z. Qin, H. Zha, Sequential recommendation with user memory networks, in: Proceedings of the 11th ACM International Conference on Web Search and Data Mining, WSDM 2018, 2018, pp. 108–116.

[3] K. Cho, B. van Merrienboer, C. Gülçehre, D. Bahdanau, F. Bougares, H. Schwenk, Y. Bengio, Learning phrase representations using RNN encoder-decoder for statistical machine translation, in: Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, 2014, pp. 1724–1734.

[4] Y. Du, Z. Peng, J. Niu, J. Yan, A unified hierarchical attention framework for sequential recommendation by fusing long and short-term preferences, Expert Syst. Appl. 201 (2022) 117102.

[5] R. He, W.-C. Kang, J. McAuley, Translation-based recommendation, in: Proceedings of 9th ACM Conference on Recommender Systems, RecSys 2017, ACM, 2017, pp. 161–169.

[6] S. Hochreiter, J. Schmidhuber, Long short-term memory, Neural Comput. 9 (8) (1997) 1735–1780.

[7] Y. Huang, Q. Liu, H. Peng, J. Wang, Q. Yang, D. Orellana-Martín, Sentiment classification using bidirectional LSTM-SNP model and attention mechanism, Expert Syst. Appl. 221 (2023) 119730.

[8] Y. Huang, H. Peng, Q. Liu, Q. Yang, J. Wang, D. Orellana-Martín, M.J. Pérez-Jiménez, Attention-enabled gated spiking neural P model for aspect-level sentiment classification, Neural Netw. 157 (2023) 437–443.

[9] M. Ionescu, Gh. Păun, T. Yokomori, Spiking neural P systems, Fundam. Inform. 71 (2006) 279–308.

[10] D. Jannach, L. Lerche, M. Jugovac, Adaptation and evaluation of recommendations for short-term shopping goals, in: Proceedings of 9th ACM Conference on Recommender Systems, RecSys 2015, 2015, pp. 211–218.

[11] W.-C. Kang, J. McAuley, Self-attentive sequential recommendation, in: Proceedings of 2018 International Conference on Data Mining, ICDM 2018, IEEE, 2018, pp. 197–206.

[12] Y. Kim, Convolutional neural networks for sentence classification, in: Proceedings of Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, 2014, pp. 1746–1751.

[13] A. Krizhevsky, I. Sutskever, G.E. Hinton, Imagenet classification with deep convolutional neural networks, Commun. ACM 60 (6) (2017) 84–90.

[14] Q. Liu, Y. Huang, Q. Yang, H. Peng, J. Wang, An attention-aware long short-term memory-like spiking neural model for sentiment analysis, Int. J. Neural Syst. 33 (8) (2023) 2350037.

[15] Q. Liu, L. Long, H. Peng, J. Wang, Q. Yang, X. Song, A. Riscos-Núñez, M.J. Pérez-Jiménez, Gated spiking neural P systems for time series forecasting, IEEE Trans. Neural Netw. Learn. Syst. 34 (9) (2023) 6227–6236.

[16] Q. Liu, L. Long, Q. Yang, H. Peng, J. Wang, X. Luo LSTM-SNP, A long short-term memory model inspired from spiking neural P systems, Knowl.-Based Syst. 235 (2022) 107656.

[17] Q. Liu, H. Peng, L. Long, J. Wang, Q. Yang, M.J. Pérez-Jiménez, D. Orellana-Martín, Nonlinear spiking neural systems with autapses for predicting chaotic time series, IEEE Trans. Cybern. (2023).

[18] F. Liu, S. Shen, Z. Fu, H. Wang, A. Zhou, J.-Y. Qi, LGCCT: a light gated and crossed complementation transformer for multimodal speech emotion recognition, Entropy 24 (7) (2022) 1010.

[19] L. Long, Q. Liu, H. Peng, J. Wang, Q. Yang, Multivariate time series forecasting method based on nonlinear spiking neural P systems and non-subsampled shearlet transform, Neural Netw. 152 (2022) 300–310.

[20] L. Long, Q. Liu, H. Peng, Q. Yang, X. Luo, J. Wang, X. Song, A time series forecasting approach based on nonlinear spiking neural systems, Int. J. Neural Syst. 32 (8) (2022) 2250020.

[21] L. Long, R. Lugu, X. Xiong, Q. Liu, H. Peng, J. Wang, D. Orellana-Martín, M.J. Pérez-Jiménez, Echo spiking neural P systems, Knowl.-Based Syst. 253 (2022) 109568.

[22] H. Peng, Z. Lv, B. Li, X. Luo, J. Wang, X. Song, T. Wang, M.J. Pérez-Jiménez, A. Riscos-Núñez, Nonlinear spiking neural P systems, Int. J. Neural Syst. 30 (10) (2020) 2050008.

[23] S. Rendle, C. Freudenthaler, L. Schmidt-Thieme, Factorizing personalized Markov chains for next-basket recommendation, in: Proceedings of the 19th International Conference on World Wide Web, 2010, pp. 811–820.

[24] S. Shen, F. Liu, A. Zhou, Mingling or misalignment? Temporal shift for speech emotion recognition with pre-trained representations, CoRR, arXiv:2302.13277 [abs], 2023.

[25] Q. Tan, F. Liu, Recommendation based on users' long-term and short-term interests with attention, Math. Probl. Eng. (2019) 7586589.

[26] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, Ł. Kaiser, I. Polosukhin, Attention is all you need, in: Proceedings of 2017 Conference and Workshop on Neural Information Processing Systems, NIPS 2017, 2017, pp. 5998–6008.

[27] K. Villatel, E. Smirnova, J. Mary, P. Preux, Recurrent neural networks for long and short-term sequential recommendation, in: Proceedings of 9th ACM Conference on Recommender Systems, RecSys 2018, 2018.

[28] X. Wei, X. Zuo, B. Yang, Sequential recommendation based on long-term and short-term user behavior with self-attention, Lect. Notes Comput. Sci. 11775 (2019) 72–83.

[29] C. Xu, J. Feng, P. Zhao, F. Zhuang, V.S. Sheng, Long-and short-term self-attention network for sequential recommendation, Neurocomputing 423 (2021) 580–589.

[30] Y. Xue, X. Cai, F. Neri, A multi-objective evolutionary algorithm with a new initialisation mechanism and self-adaptive crossover operator for large-scale feature selection in classification, Appl. Soft Comput. 127 (2022) 109420.

[31] Y. Xue, Y. Tang, X. Xu, J. Liang, F. Neri, Multi-objective feature selection with missing data in classification, IEEE Trans. Emerg. Top. Comput. Intell. 6 (2) (2022) 355–364.

[32] Y. Xue, H. Zhu, F. Neri, A feature selection approach based on NSGA-II with reliefF, Appl. Soft Comput. 134 (2023) 109987.

[33] B. Yang, L. Qin, H. Peng, C. Guo, X. Luo, J. Wang, SDDC-Net: a U-shaped deep spiking neural P convolutional network for retinal vessel segmentation, Digit. Signal Process. 136 (2023) 104002.

[34] H. Ying, F. Zhuang, F. Zhang, Y. Liu, G. Xu, X. Xie, H. Xiong, J. Wu, Sequential recommender system based on hierarchical attention networks, in: Proceedings of 2018 International Joint Conference on Artificial Intelligence, IJCAI 2018, 2018.

[35] H. Ying, F. Zhuang, F. Zhang, Y. Liu, G. Xu, X. Xie, H. Xiong, J. Wu, Sequential recommender system based on hierarchical attention networks, in: Proceedings of 2018 International Joint Conference on Artificial Intelligence, IJCAI 2018, 2018.

[36] W. Zaremba, I. Sutskever, Learning to execute, arXiv preprint, arXiv:1410.4615, 2014.

[37] G. Zhang, H. Rong, P. Paul, Y. He, F. Neri, M.J. Pérez-Jiménez, A complete arithmetic calculator constructed from spiking neural P systems and its application to information fusion, Int. J. Neural Syst. 31 (01) (2021) 2050055.

[38] Y. Zhang, Q. Yang, Z. Liu, H. Peng, J. Wang, A prediction model based on gated nonlinear spiking neural system, Int. J. Neural Syst. 33 (6) (2023) 2350029.

[39] G. Zhang, X. Zhang, H. Rong, P. Paul, M. Zhu, F. Neri, Y-S. Ong, A layered spiking neural system for classification problems, Int. J. Neural Syst. 32 (08) (2022) 2250023.

[40] S. Zhao, L. Zhang, H. Peng, Z. Liu, J. Wang, ConvSNP: a deep learning model embedded with SNP-like neurons, J. Membrane Comput. 4 (1) (2022) 87–95.