IEICE TRANS. INF. & SYST., VOL.E100–D, NO.1 JANUARY 2017

PAPER

Using a Single Dendritic Neuron to Forecast Tourist Arrivals to Japan

Wei CHEN[†], **Jian SUN**[†,††], ***Nonmembers***, **Shangce GAO**[†a)], ***Member***, **Jiu-Jun CHENG**[†††], **Jiahai WANG**[††††], ***and*** **Yuki TODO**[†††††], ***Nonmembers***

SUMMARY With the fast growth of the international tourism industry, it has been a challenge to forecast the tourism demand in the international tourism market. Traditional forecasting methods usually suffer from the prediction accuracy problem due to the high volatility, irregular movements and non-stationarity of the tourist time series. In this study, a novel single dendritic neuron model (SDNM) is proposed to perform the tourism demand forecasting. First, we use a phase space reconstruction to analyze the characteristics of the tourism and reconstruct the time series into proper phase space points. Then, the maximum Lyapunov exponent is employed to identify the chaotic properties of time series which is used to determine the limit of prediction. Finally, we use SDNM to make a short-term prediction. Experimental results of the forecasting of the monthly foreign tourist arrivals to Japan indicate that the proposed SDNM is more efficient and accurate than other neural networks including the multi-layered perceptron, the neuro-fuzzy inference system, the Elman network, and the single multiplicative neuron model.

key words: *artificial neural networks, chaos, dendritic neuron model, phase space reconstruction, time series prediction, tourism demand*

1. Introduction

In the past few decades, the significant growth of international tourism has been achieved in Japan, and the tourism industry has become a crucial contribution to Japan's economic development. According to the Japanese National Tourism Organization [1], the estimated number of international visitors to Japan in January 2016 reached to 1.85 million, recording the highest figure for January on a monthly basis [2]. Chinese tourists are going wild on a shopping spree in Japan, resulting a new word "Bakugai" in Japanese. It is highly important for Japanese tourism agencies including government bodies and the private sector to understand the trends affecting monthly tourist arrivals. Thus, the forecasting visitor arrivals is crucial for better tourism planning and administration.

Traditional tourism demand researches tend to use linear parametric time series forecasting models. The most popular are the autoregressive integrated moving average models [3]–[5], the naive method [6], [7], and the exponential smoothing model [8]. However, the predictions obtained using these traditional models are usually imprecise, and it is difficult to utilize these models to approximate nonlinear and irregular tourism time series [9], [10].

Recently, more and more nonlinear forecasting models are proposed to address the above issues in the time series prediction. A piecewise linear method is proposed to model and forecast the demand for the tourism, and the experimental results indicate that the piecewise linear model is significantly more accurate than those autoregressive models [11]. A regime switching detection and forecasting model is proposed in [12]. However, the performance of these models is limited to the problems of proper model selection and data dependency [10], [13].

On the other hand, machine learning techniques are developed for time series forecasting, such as support vector machines [14]–[16], fuzzy time-series methods [17], rough set approaches [18], [19], genetic programming [20], artificial neural networks (ANNs) [21]–[28] and their hybridizations [29]–[32]. These complex non-linear models overcome the limitation of linear models as they are able to capture non-linear pattern of data, thus improving their prediction performance.

Among them, ANNs are receiving increasing interests due to their ability to adapt to imperfect data, functions of self-organizing, self-study, data-driven, associated memory, and arbiter function mapping [9]. ANNs can learn from patterns and capture hidden functional relationships in a given data even if the functional relationships are not known or difficult to identify [33], [34]. Using the training methods, an ANN can be trained to identify the underlying correlation between the inputs and outputs, and finally to generate appropriate outputs. A number of researchers have utilized ANNs to predict tourism demand [24], [25], [27], [31], [35], [36]. Kon and Turner [24] provided a review of the applications of ANN in tourism. Empirical evidences show that ANNs outperform the classical linear models in tourism forecasting. For example, the best performance was obtained by an ANN method in [22] when compared it with the naive, decomposition, exponential smoothing and regression models.

Manuscript received April 8, 2016.
Manuscript revised September 15, 2016.
Manuscript publicized October 18, 2016.

[†]The authors are with Faculty of Engineering, University of Toyama, Toyama-shi, 930–8555 Japan.
[††]The author is with College of Computer Science and Technology, Taizhou University, China.
[†††]The author is with The Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai 200092, China.
[††††]The author is with Department of Computer Science, Sun Yat-sen University, Guangzhou 510006 China.
[†††††]The author is with Faculty of Electrical and Computer Engineering, Kanazawa University, Kanazawa-shi, 920–1192 Japan.
a) E-mail: gaosc@eng.u-toyama.ac.jp
DOI: 10.1587/transinf.2016EDP7152

Although various ANNs have been proposed for tourism time series, it is difficult to identify the best compared with others over all instances [37] because each ANN has its distinct characteristics and limitations, which influence the prediction performance. For example, despite the widespread applicability of the multiple-layered perceptron (MLP), the back-propagation-based MLP can only learn an input-output mapping for static or spatial patterns that are independent of time. The time-delayed ANN may be the simplest choice for representing a wide range of mappings between past and present values [38], but the fixed time delays in these ANNs remain constant throughout training after initialization, thereby risking a mismatch between the choice of time delay values and the temporal locations of important information in the input patterns [39]. The Elman recurrent ANN [40] has advantages compared with the MLP because the memory features obtained using a feedback mechanism can be used to extract time dependencies from the data. However, the traditional recurrent ANN algorithms based on the gradient descent approach are well known for their slow convergence and high computational costs [41], thus it is difficult to utilize them in actual applications.

In this paper, we propose a realistic single dendritic neuron model (SDNM) with synaptic nonlinearities in a dendritic tree for tourism forecasting. The distinct characteristic of SDNM is that the sense of locality of dendrites can be represented and manipulated. For a specific given task, SDNM is able to identify what type of synapse (excitatory or inhibitory) is needed, where the synapse should be located, which branch of the dendrite is needed, and which one is not needed [42], [43]. This is realized by modeling the synaptic nonlinearity with a sigmoid function, and thus enabling the single neuron to be capable of computing linearly non-separable functions and approximating any complex continuous function [44], [45]. On the other hand, although the tourist arrivals time series apparently is one-dimension, it actually contains high-dimensional information and is a result of many factors such as the tourism policy which strongly influences the number of tourists, and thus making the tourism data nonlinear, irregular, and difficult to be predicted. To address this problem, we employ the phase space reconstruction (PSR) technique based on the Takens's embedding theorem [46] to handle the chaotic properties of the tourism time series before using SDNM to perform the prediction. By doing so, a set of single observations from the tourist arrivals can be reconstructed into a series of multiple dimensional vectors with two parameters of time delay and embedding dimension. The acceptable dimensions and time delay of the attractors in the tourism time series can be obtained, thereby allowing the time series data to be manipulated without losing the dynamic behavior and structural topology. Based on the the maximum Lyapunov exponent of the reconstructed phase points of tourism time series, SDNM is then used to perform short-term predications. Experimental results of the forecasting of the monthly foreign tourist arrivals to Japan indicate that the proposed SDNM is more efficient and accurate than other neural networks including the MLP, the artificial neuro-fuzzy inference system (ANFIS), the Elman network (Elman), and the single multiplicative neuron model (SMN).

The rest of the paper is organized as follows. Section 2 describes the SDNM in details. Section 3 elaborates more about the prediction method by using PSR and SDNM. Experimental results and discussions are given in Sect. 4. Finally, concluding remarks are presented in Sect. 5.

2. Single Dendritic Neuron Model

Compared with ANNs which utilize more than one neurons in information processing procedure, many attentions have been paid to propose single neuron models, such as the single multiplicative neuron model [47], [48] and the sigma-pi unit [49]. However, these single neuron based models are based on the architecture of the McCulloch-Pitts neuron which uses weights to represent the degree of clustering between synapses. Thus, all sense of locality in dendrites is lost, and these models could not represent local interaction within a fixed dendritic tree. Moreover, the nonlinear computational capabilities of these McCulloch-Pitts based single neuron models are limited to solve complex problems, especially the non-linearly separated problems [50].

Different from the McCulloch-Pitts neuron based models which do not consider the dendritic structure in the neuron, it has been recently conjectured by a series of theoretical studies that individual neurons could act more powerfully as computational units by considering synaptic nonlinearities in a dendritic tree [51], [52]. The various types of synaptic plasticity and nonlinearity mechanisms allow synapses to play a more important role in computations [53]. Synaptic inputs from different neuronal sources can be distributed spatially on the dendritic tree and the plasticity in neuron can result from changing in synaptic strength or connectivity, and the excitability of the neurons themselves [54]. Moreover, a slight morphological difference can just cause great functional variation, acting as filters to determine what signals a single neuron receives and then how these signals are integrated [55].

By taking the nonlinearity of synapses into consideration, a single dendritic neuron model (SDNM) has been proposed in our previous researches [42], [44], [45]. In [42], an unsupervised learning method was proposed for SDNM to learn two-dimensional eight-directionally selective problems. In [44], an error back-propagation (BP) method was used for training SDNM to perform cancer classification tasks. In [45], we demonstrated that SDNM could be approximately realized by using logic NOT, AND and OR operations, corresponding to its dendritic morphology, and thus was suitable for a simple hardware implementation in practice. In this study, we apply SDNM to perform the tourism arrivals forecasting. The details of SDNM are described in the following and its architecture is shown in Fig. 1.

SDNM is constituted by four layers including a synap-

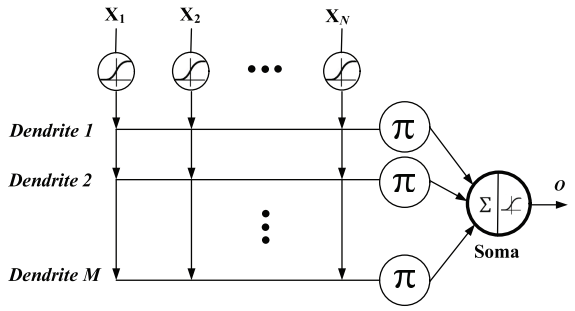

Fig. 1 The architecture of the single dendritic neuron model (SDNM).

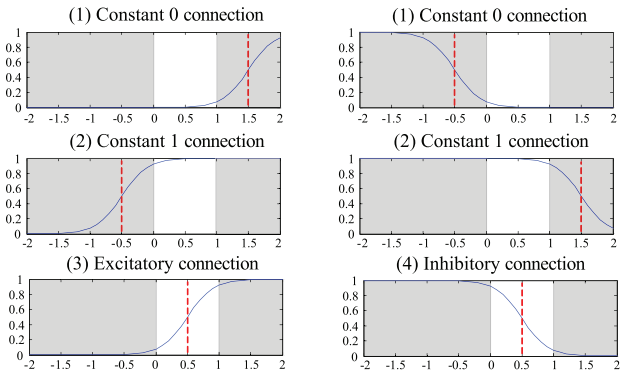

Fig. 2 Four kinds of connection cases in the synaptic layer.

tic layer which performs sigmoid functions, a dendrite layer which acts as a multiplicative function for the outputs of synapses, a membrane layer which actually is an addition function for the outputs of all dendritic branches, and a soma function which uses another sigmoid function to output the result of the entire single neuron.

2.1 Synaptic Layer

A synapse refers to the connection between neurons at a terminal bouton of a dendrite to another dendrite/axon or the soma of another neural cell. The direction of information flow is feedforward, from the presynaptic neuron to postsynaptic neuron. The synapse can be either excitatory or inhibitory which depends on changes in the postsynaptic potential caused by ionotropic. The function connecting the i-th ($i = 1, 2, \ldots, N$) synaptic input to the j-th ($j = 1, 2, \ldots, M$) synaptic layer is expressed by Eq. (1). The value k is a positive constant, and the weight w_{ij} and the threshold θ_{ij} are the connection parameters.

$$Y_{ij} = \frac{1}{1 + e^{-k(w_{ij}x_i - \theta_{ij})}} \tag{1}$$

where x_i is the input part of a synapse, referred to as the pre-synaptic terminal, and its range is [0, 1].

Depending on the values of w_{ij} and θ_{ij}, there are four kinds of connection cases as shown in Fig. 2, where the graph's horizontal axis represents the inputs of presynaptic neurons; the vertical axis shows the output of the synaptic layer. Because the range of x is [0,1], only the corresponding part needs to be observed. The four connection cases include: (1) A constant 0 connection (when $w_{ij} < 0 < \theta_{ij}$ or $0 < w_{ij} < \theta_{ij}$) where the output will approximately be 0 whenever the input changes from 0 to 1. (2) A constant 1 connection (when $\theta_{ij} < w_{ij} < 0$ or $\theta_{ij} < 0 < w_{ij}$) where the output will approximately be 1 whenever the input changes from 0 to 1. (3) Excitatory connection (when $0 < \theta_{ij} < w_{ij}$) where the synapse will be an excitatory type if the input changes from 0 to 1 and the output is proportional to the input. (4) Inhibitory connection (when $w_{ij} < \theta_{ij} < 0$) where the synapse will be an inhibitory type and the output will be inversely proportional to the input in this case.

It is worth pointing out that these four connection cases are crucial to identify the morphology of a neuron via specifying the locations and synapse types of dendrites. For a more detailed description of morphology detection, readers can refer to [42], [44], [45].

2.2 Dendrite Layer

The dendrite layer simply performs a multiplication on various synaptic connections of each branch. As mentioned before, the nonlinearity of synapses could be used to implement a type of multiplication instead of summation, thus our model adopts the multiplicative operation in the dendrite layer. It should be noted that a soft-minimization operator was utilized in our previous dendritic neuron model [42] to deal with binary input classification problem, while the multiplicative operation adopted in this study can address real number input problems. The multiplication is very equal to the logic AND operation as the value of inputs and outputs of the dendrites are either 1 or 0. In Fig. 1, the multiplication operator is represented by the symbol "π". The output equation for the j-th branch can be given as follows.

$$Z_j = \prod_{i=1}^{N} Y_{ij} \tag{2}$$

2.3 Membrane Function

Subsequently, the result received from the branch is calculated by a summation operation, which is similar to a logic OR operation in the binary case. The output is approximated as follows.

$$V = \sum_{j=1}^{M} Z_j \tag{3}$$

2.4 Soma Function

Finally, a sigmoid function is utilized to obtain the value of the output, which can be described as follows.

$$O = \frac{1}{1 + e^{-k_{soma}(V - \theta_{soma})}} \tag{4}$$

The parameter k_{soma} is set as a positive constant and the

threshold θ_{soma} is adjusted from 0 to 1.

2.5 BP-like Learning Method

The error between the ideal target vector T_p and the actual output vector O_p ($p = 1, 2, \ldots, P$) can be represented by Eq. (5) and P denotes the number of training samples.

$$E_p = \frac{1}{2}(T_p - O_p)^2 \tag{5}$$

According to the error back-propagation (BP) learning rule [44], we can perform learning using a function for modifying the connection parameters w_{ij} and θ_{ij} as the connection function during learning. The output vector produced by the input vector is compared to the target vector, which can decrease the error between output vector and teaching signal T_p vector by correcting w_{ij} and θ_{ij}. Eventually, the synapses can converge to one of the four synaptic connections. The connection parameters should be corrected using the gradient descent learning function, which is a method for modifying the value of the error function, as follows:

$$\Delta w_{ij}(t) = -\eta \sum_{p=1}^{P} \frac{\partial E_p}{\partial w_{ij}} \tag{6}$$

$$\Delta \theta_{ij}(t) = -\eta \sum_{p=1}^{P} \frac{\partial E_p}{\partial \theta_{ij}}, \tag{7}$$

where η represents the learning constant and it is a positive constant. The updating rules for w_{ij} and θ_{ij} are defined as:

$$w_{ij} = w_{ij} + \Delta w_{ij}(t) \tag{8}$$

$$\theta_{ij} = \theta_{ij} + \Delta \theta_{ij}(t) \tag{9}$$

where t denotes the learning epoch.

It should be pointed out that the BP learning method is carried out in a batch mode, where the weight changes resulted by BP are accumulated over an entire presentation of training samples before the updating (i.e., Eqs. (8) (9)) is applied. The reasons that the batch mode is selected are manifold: (1) it requires less weight update and provides a more accurate measurement of the required weight changes [56]; (2) the batch mode requires shorter training time since updates can be done much faster [57], [58]; and (3) the batch training is theoretically superior to on-line training because it uses the true gradient and is slightly more efficient in terms of computations [59].

Moreover, the partial differentials of E with respect to w_{ij} and θ_{ij} can be computed as follows.

$$\frac{\partial E_p}{\partial w_{ij}} = \frac{\partial E_p}{\partial O_p} \cdot \frac{\partial O_p}{\partial V} \cdot \frac{\partial V}{\partial Z_j} \cdot \frac{\partial Z_j}{\partial Y_{ij}} \cdot \frac{\partial Y_{ij}}{\partial w_{ij}} \tag{10}$$

$$\frac{\partial E_p}{\partial \theta_{ij}} = \frac{\partial E_p}{\partial O_p} \cdot \frac{\partial O_p}{\partial V} \cdot \frac{\partial V}{\partial Z_j} \cdot \frac{\partial Z_j}{\partial Y_{ij}} \cdot \frac{\partial Y_{ij}}{\partial \theta_{ij}} \tag{11}$$

The components in the above partial differential are shown as follows.

$$\frac{\partial E_p}{\partial O_p} = O_p - T_p \tag{12}$$

$$\frac{\partial O_p}{\partial V} = \frac{k_{soma} e^{-k_{soma}(v-\theta_{soma})}}{(1 + e^{-k_{soma}(v-\theta_{soma})})^2} \tag{13}$$

$$\frac{\partial V}{\partial Z_j} = 1 \tag{14}$$

$$\frac{\partial Z_j}{\partial Y_{ij}} = \prod_{L=1 \ and L \neq i}^{N} Y_{Lj} \tag{15}$$

$$\frac{\partial Y_{ij}}{\partial w_{ij}} = \frac{k x_i e^{-k(x_i w_{ij} - \theta_{ij})}}{(1 + e^{-k(x_i w_{ij} - \theta_{ij})})^2} \tag{16}$$

$$\frac{\partial Y_{ij}}{\partial \theta_{ij}} = \frac{-k e^{-k(x_i w_{ij} - \theta_{ij})}}{(1 + e^{-k(x_i w_{ij} - \theta_{ij})})^2} \tag{17}$$

2.6 Remarks regarding Characteristics of SDNM

- The architecture of SDNM is similar to those of multiplicative neuron models and sigma-pi models. They are multiple-layered and signals are transferred in a feed-forward manner. As a result, the functions used in these models can be reciprocated. For example, the radial basis functions using Gaussian kernels, a simplified fuzzy logic formulation and kernel-regression models are able to be represented by a variation of sigma-pi formulation [60]. Furthermore, some of them are isomorphic (e.g. the augmented two-layer neuron model 2LM is isomorphic to a traditional ANN [61]).
- Multiplication is both the simplest and one of the most widespread of all nonlinear operations in the nervous system [62]. Taking advantage of the multiplication operation which is essential and important to the information processing in a neuron [63], the computation in synapses is innovatively modelled using sigmoid functions. Depending on the values of the parameters in synapses, the output of synapses can successfully represent excitatory, inhibitory, constant 0 and constant 1 signals, which is benefit for identifying the morphology of a neuron [42].
- SDNM has been successfully applied on a number of classification problems, such as XOR [64], cancer diagnosis [44], Iris and Glass datasets [45]. On the contrary, some other dendritic neuron models are not able to solve such nonlinearly separated problems [50] (e.g., the Legenstein-Maass model [65]). More importantly, the classifier resulted from SDNM can be easily implemented in hardware [45] using logic circuits.

3. Forecasting Framework for Tourist Arrivals

The framework for forecasting the tourist arrivals based on PSR and SDNM is shown in Fig. 3, where PSR is utilized to analyze the behavior of tourism time series based on the Takens's embedding theorem and SDNM is used to perform the predication. Following Fig. 3, the procedures of the forecasting method are summarized as in the following.

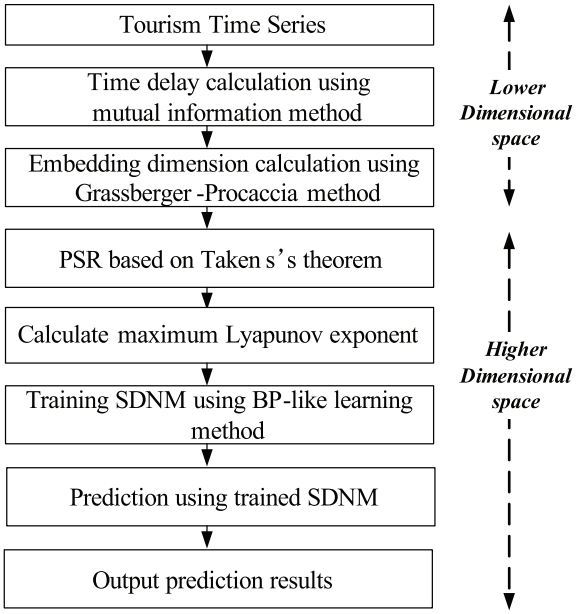

Fig. 3 Prediction framework based on the proposed SDNM.

3.1 Input Time Series Data

Let x_t be the one-dimensional tourism time series at time t, $(t = 1, 2, \ldots)$. First of all, x_t is input and processed using a normalization method to the range of [0, 1] according to Eq. (18).

$$y_t = \frac{x_t - MIN(x_t)}{MAX(x_t) - MIN(x_t)} \tag{18}$$

where y_t is the normalized data to alleviate the problem of inconsistent measures for different time series data, and MAX (MIN) returns the maximal (minimal) value of the vector. It is worth emphasizing that the normalization method is performed on both training and testing samples of the tourism time series. That is to say, the dataset used in the experiment has been pre-processed by normalizing them between 0 and 1. Other normalization methods, such as the mean and variance normalization, or simple normalization [33] are also worth being utilized [66] For a comprehensive review of the data normalization techniques in neural networks for forecasting, readers can refer to as in [33].

3.2 Phase Space Reconstruction

Real-world tourism time series perform chaotically and unpredictably according to long-term observations, and thus it is difficult to obtain reliable future forecasts. By contrast, they exhibit periodicity when reconstructed as a phase point in a phase space. Thus, making predictions in the phase space based on PSR is easier than using a one-dimensional time series. PSR is regarded as the basis of chaotic time series and widely used in non-linear system analysis. It is a theory for inferring geometrical and topological information related to a dynamical attractor based on observations. Takens [46] proposed the delay coordinates method of PSR for time series analysis, and proved that PSR can unfold the time series into an m-dimensional embedding space while retaining the topology of the higher dimensional dynamic system with the chaotic attractor.

Two parameters of the time delay τ and the embedding dimension m are very important in PSR. Theoretically any value of τ is acceptable for the choice of the delay time. However, the appearance of the reconstructed attractor depends strongly on the choice of embedding lag. A suitable value for τ must bear the function to sufficiently separate the data in the time series as to have a smooth reconstruction of the attractor. In this study, we use an appropriate embedding dimension m and time delay τ to reconstruct the phase space. The Grassberger-Procaccia algorithm [67] is used to determine the embedding dimension m and the mutual information function [68] is used to calculate the time delay τ. More details regarding the implementation of these two methods are interpreted in Sect. 4. As a result, a reconstructed phase space can be represented by a matrix $(P, T)'$ for the normalized time series y_t, $t = 1, \ldots, N$, where

$$P = \begin{pmatrix} y_1 & y_2 & \cdots & y_{N-1-\tau(m-1)} \\ y_{1+\tau} & y_{2+\tau} & \cdots & y_{N-1-\tau(m-2)} \\ & & \cdots & \\ y_{1+\tau(m-1)} & y_{2+\tau(m-1)} & \cdots & y_{N-1} \end{pmatrix} \tag{19}$$

$$T = (y_{2+\tau(m-1)}, y_{3+\tau(m-1)}, \ldots, y_N) \tag{20}$$

In the training and forecasting process of SDNM, P is used as the input data, while T is treated as the target data.

3.3 Maximum Lyapunov Exponent Calculation

To determine whether a long-term or a short term predication of the trajectory of the tourism can be made, it is necessary to calculate the Lyapunov exponents of the time series which is able to quantitatively characterize the chaotic attractor and provides an important measure for the sensitivity of the chaotic orbit to its initial conditions. Lyapunov exponents describe the growth or shrinkage rate of small perturbations in different directions in the phase space of the orbits. The time series changes into chaos when the Lyapunov exponent is computed as positive [69]. The Wolf method [69] is employed to compute the largest Lyapunov exponents of the chaotic time series based on the phase track in the present study. We assume that the initial time is t_0 and that the reconstructed first phase point is y_{t_0}, where the minimum length compares y_{t_0} with its adjacent phase points is L_0. By considering the evolution of the two phase points, the distance $L'_0 > \varepsilon$ has a positive threshold value when the time is t_1, $L'_0 = \|y_{t_1} - y_{t_0}\|$. If another phase point $y^1_{t_1}$ with $L^1 = \|y_{t_1} - y^1_{t_1}\| < L'_0$ is found, then L'_0 will be substituted. Finally, this calculation process is continued until y_t arrives at the end of the time series y_N. Hence, we can process the largest Lyapunov exponent as the following function.

$$\lambda_{max} = \frac{1}{t_m - t_0} \sum_{i=0}^{m} ln \frac{L'_i}{L_i} \tag{21}$$

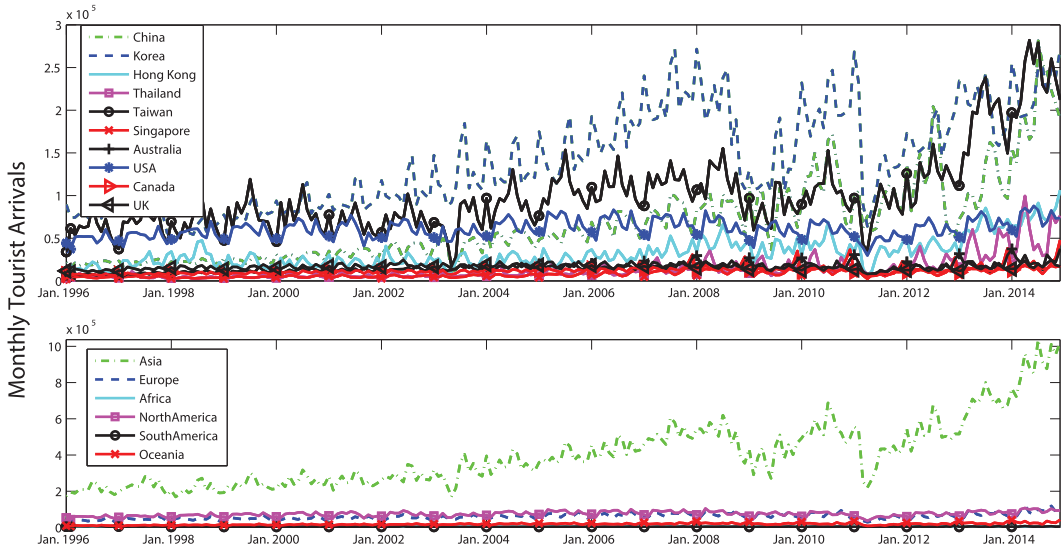

Fig. 4 The monthly tourist arrivals from ten major source markets and six continents to Japan.

In a chaotic system, we can only predict the time series in short intervals. The reciprocal Lyapunov exponent can be used to determine how short the intervals will be in theory [70], [71].

$$\Delta t = \frac{1}{\lambda_{max}} \tag{22}$$

When the maximum Lyapunov exponent exceeds zero, the system exhibits chaos. If it is greater than one, the predictable limit is less than the sampling frequency. Thus, the chaotic time series predictions are only of practical use when the chaotic system with the maximum Lyapunov exponent is between zero and one. If the positive exponent approaches zero, long-term predictions are possible.

3.4 Prediction Using SDNM

When the reconstruction of phase space and maximum Lyapunov exponent calculation accomplished, we carry out the prediction for tourism arrivals based on the SDNM described in Sect. 2. First, we divide all time series data into two parts: one is used as training data set and the other is used to verify the prediction accuracy. Then we implement the BP-like learning method to optimize the weights w_{ij} and thresholds θ_{ij} in the synaptic layer of SDNM until a learning termination condition is fulfilled. In this study, a maximum learning epoch L_{max} is used as the termination condition. Finally, we output the prediction results using some assessment methods.

4. Experimental Results and Analysis

We use our proposed method to study monthly foreign tourist arrivals to Japan from the eight major markets of China, Korea, Hong Kong, Thailand, Taiwan, Singapore, Australia, USA, Canada and UK, and form six continents of Asia, Europe, Africa, North America, South America, and Oceania, respectively, from January 1996 to December 2014. These data are published by Japanese National Tourism Organization [1]. Figure 4 illustrates these data in one-dimensional time series. For each sequence of the tourism arrival, there are 228 points, where the first 168 (14 years) points are employed for SDNM learning and the remaining 60 (5 years) points for verification. All experiments are conducted using Matlab (R2013) software on a personal PC with Intel(R) Core i5, 1.70GHz and 4GB memory.

4.1 Time Delay and Embedding Dimension

The time delay τ is calculated to take the value for which the mutual information has its first minimum [68]. The mutual information $I(y, y_\tau)$ between two time series $y = \{y_{t_1}, y_{t_2}, \ldots, y_{t_N}\}$ and $y_\tau = \{y_{t_1+\tau}, y_{t_2+\tau}, \ldots, y_{t_N+\tau}\}$ is the average bits where y was predicted by the measurement from y_τ. $I(y, y_\tau)$ can be represented as

$$I(\tau) = I(y, y_\tau) = H(y) + H(y_\tau) - H(y, y_\tau) \tag{23}$$

where $H(y)$ and $H(y_\tau)$ are the entropy of y and y_τ respectively. $H(y, y_\tau)$ is the mutual entropy between y and y_τ. Generally, the moment of the first minimal mutual information is taken as the optimal delay time for PSR. Figure 5 shows the time delay sequence of the monthly tourism arrivals time series from China to Japan with respect to the mutual information. It is apparent that the time delay is 3 months as the first minimal mutual information appears, namely $\tau = 3$. All the time delays for ten major source markets and six continents are summarized in Table 1 and Table 2 with the values located in the interval of [2, 7].

Once the time delay is determined, we use the Grassberger-Procaccia algorithm to calculate the embedding dimension. First, the correlation integral $C(r)$ is calculated:

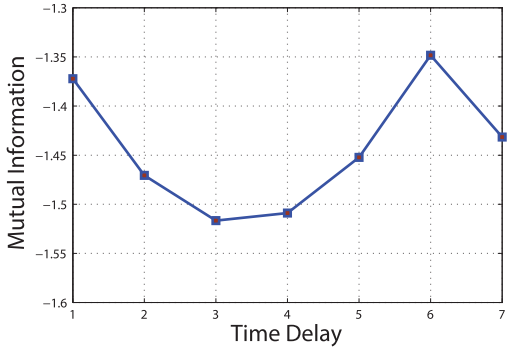

Fig. 5 The mutual information versus time delay for tourism time series from China to Japan after PSR.

Table 1 Results of PSR for the monthly tourist arrivals from **ten major source markets** to Japan: the embedding delay τ, the embedding dimension m, and the maximum Lyapunov exponents MLE.

Country	τ	m	MLE
China	3	8	0.2510
Korea	2	12	0.0779
Hong Kong	3	6	0.3060
Thailand	4	4	0.2121
Taiwan	5	4	0.1416
Singapore	2	18	0.0173
Australia	2	13	0.0333
USA	4	12	0.2520
Canada	2	10	0.1269
UK	2	12	0.0669

Table 2 Results of PSR for the monthly tourist arrivals from **six continents** to Japan: the embedding delay τ, the embedding dimension m, and the maximum Lyapunov exponents MLE.

Continent	τ	m	MLE
Asia	2	12	0.0067
Europe	2	14	0.0473
Africa	4	6	0.3339
North America	7	9	0.0146
South America	3	9	0.0691
Oceania	2	14	0.0467

$$C(r) = \frac{2}{N_m(N_m) - 1} \sum_{1 \le i \le j \le N_m} \varphi(r - |y_i - y_j|) \tag{24}$$

where $N_m = N - \tau(m-1)$, r is the chosen radius and $\varphi(\cdot)$ is the Heaviside function. The correlative dimension $D(m)$ ($D(m) = ln(C(r))/ln(r)$) increases with the increment of the embedding dimension m, and gradually converges to a saturation value. We plot $ln(C(r))$ vs. $ln(r)$ for different m, which is presented in Fig. 6, for the monthly tourism arrivals time series from China to Japan. Intuitively, several nearby parallel line segments exist in the figure, which indicate that when $ln(r)$ varies in [9.5, 13], the embedding dimension m varies from 1 to 20. The slopes of the line portion can be estimated as the correlation dimension which is shown in Fig. 7. The embedding dimension is determined as the value when $D(m)$ first reaches a stable value. Thus, we obtain $m = 8$ for the monthly tourism arrivals time series from China to Japan. The values of the embedding dimension for other time series instances are summarized in Table 1 and Table 2.

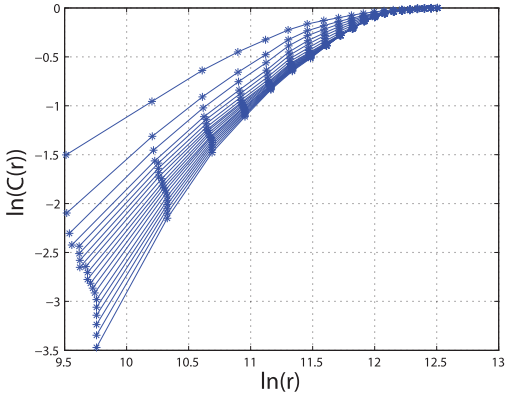

Fig. 6 The correlation function, ln(C(r)) versus ln(r) of the monthly tourist arrivals from China to Japan.

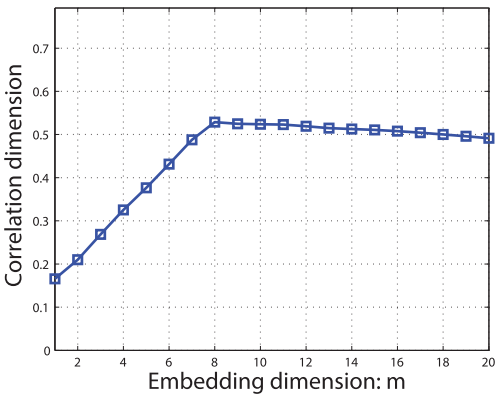

Fig. 7 Correlation dimension (fitted ln(C(r)/ln(r))) versus the embedding dimension for tourism time series from China to Japan after PSR.

4.2 PSR and Lyapunov Exponent

Using the obtained time delay τ and embedding dimension m, we reconstruct the phase space by Eqs. (19) and (20) from the original one-dimension time series. The reconstructed phase space is exhibited using a three-dimensional phase space, although the calculated embedding dimension $m = 8$ in the tourism time series of China, which means that it is difficult to explicitly map the higher-dimensional information onto a lower-dimensional space. However, we locate three vectors in different three-dimensional phase space without losing the distortion factor because the three dimensions contributed to the geometric representation, and thus they can also intuitively represent the structure of the attractor. Figure 8 depicts the results of PSR using two three-dimensional vectors $(y_t, y_{t+3\tau}, y_{t+6\tau})$ and $(y_{t+\tau}, t_{t+4\tau}, t_{t+7\tau})$ for the PSR results of China, respectively. Both three-dimensional vectors show clear chaotic attractors, which suggest that the distributed trace for the tourism exhibits the property of dissipation, and thereby indicating that it is an ordered dynamic system despite possessing the features of a strange attractor. Similar PSR results can also be plotted for the other tourism time series.

The Lyapunov exponents are the average exponential

rates of divergence or convergence of adjacent orbits in phase space. All maximum Lyapunov exponents (*MLE*) for the tourism time series are calculated to verify whether the tourism system is chaotic and further to determine the limitation of the predication. The *MLE* results are also summarized in Table 1 and Table 2, and these *MLE* are positive values between [0, 1] for all cases, indicating chaotic behaviors. Besides, as the obtained *MLE* has relative large values, it is more reliable to predict the tourism arrivals in a shorter time range (i.e., to perform a short-term forecasting).

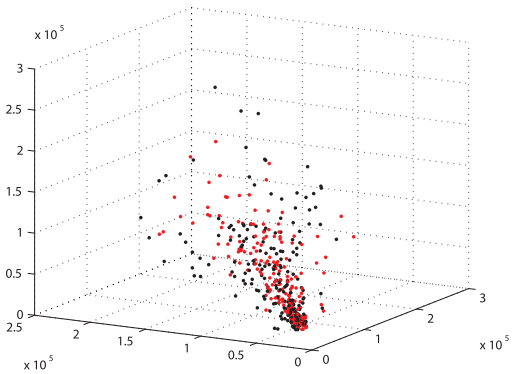

Fig. 8 Three-dimension phase space for tourism time series from China to Japan after PSR.

4.3 Short-Term Forecasting and Performance Comparison

Generally, with the length of the time range to be forecasted increasing, the predication accuracy will decrease. In this study, we use five years as the forecasting time length to evaluate the performance of our proposed method. It is worth emphasizing that, within the five years, the former estimated values will be used to forecast the latter values based on PSR. In addition, user-defined parameters in SDNM influence the prediction performance for the tourism time series. These parameters include the number of dendrites M, the parameter k in synapses (Eq. (1)), the parameters k_{soma} and θ_{soma} in the soma function (Eq. (4)), the BP learning rate η (Eqs. (6) and (7)), and the maximum learning epoch L_{max}. It should be noted that the input number parameter N is set to be the embedding dimension m in SDNM.

It is not trivial to set user-defined parameters to obtain the best performance for SDNM, and generally no sys-

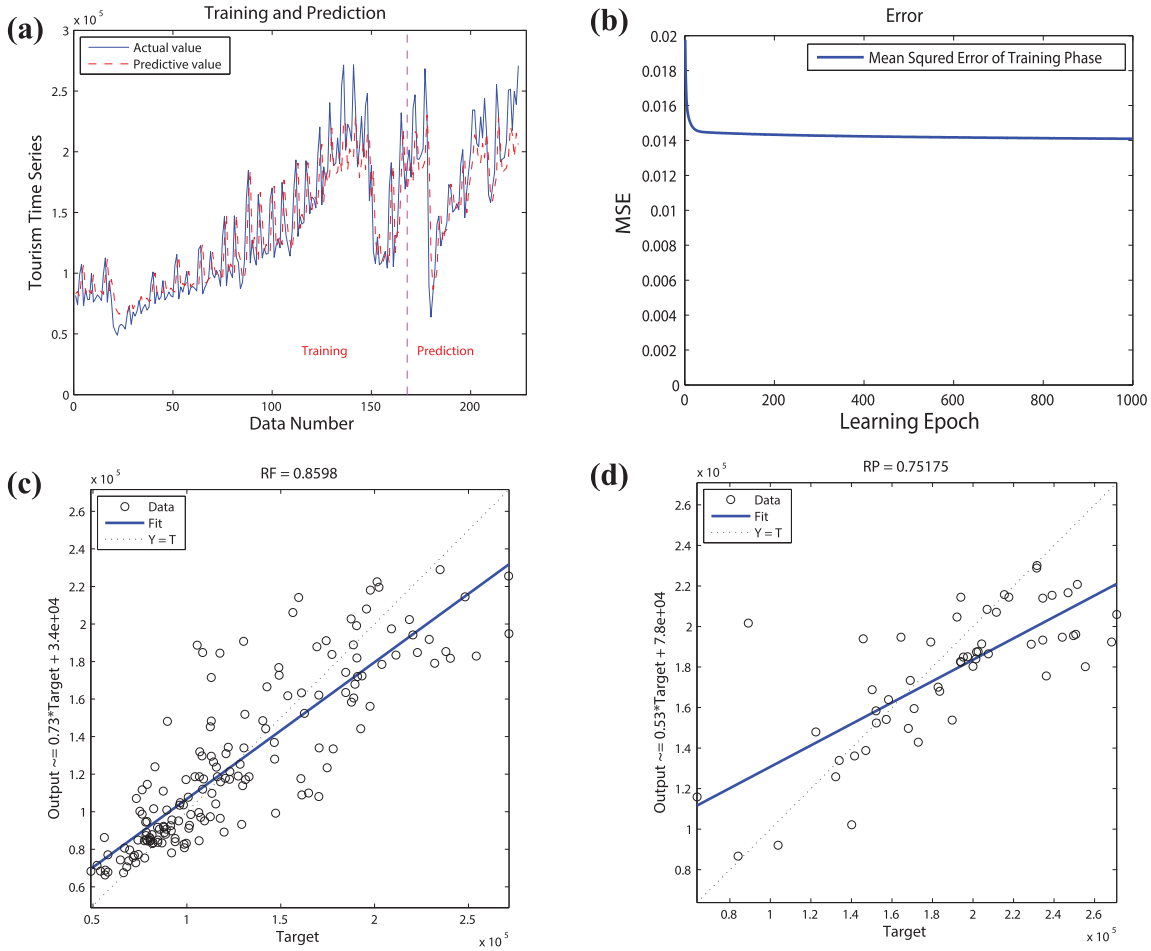

Fig. 9 Performance of proposed method for the tourism time series of Korea to Japan: (a) training and prediction results, (b) convergence graph based on the BP-like learning, (c) the correlation coefficient of fitting, and (d) the correlation coefficient of prediction.

Table 3 Results based on the $L_{16}(4^5)$ orthogonal array and factor assignment.

No.	M	k	k_{soma}	θ_{soma}	η	MSE ($\times 10^{-2}$)	Time
1	1	1	1	0	0.005	3.63 ± 0.58	2.8
2	1	3	3	0.3	0.01	2.01 ± 0.45	2.8
3	1	5	5	0.5	0.05	1.57 ± 0.37	2.8
4	1	10	10	0.9	0.1	1.83 ± 0.46	2.8
5	3	1	3	0.5	0.1	1.61 ± 0.57	8.8
6	3	3	1	0.9	0.05	1.56 ± 0.73	8.8
7	3	5	10	0	0.01	2.33 ± 0.75	8.8
8	3	10	5	0.3	0.005	1.98 ± 0.81	8.8
9	5	1	5	0.9	0.01	3.45 ± 1.24	12.7
10	5	3	10	0.5	0.005	2.23 ± 0.95	12.7
11	5	5	1	0.3	0.1	2.35 ± 1.02	12.7
12	5	10	3	0	0.05	2.77 ± 1.45	12.7
13	10	1	10	0.3	0.05	3.05 ± 1.68	18.5
14	10	3	5	0	0.1	2.94 ± 1.43	18.5
15	10	5	3	0.9	0.005	2.37 ± 1.02	18.5
16	10	10	1	0.5	0.01	1.84 ± 0.53	18.5

tematic procedure exists to find out the optimal values for these parameters except the exhaustive method which is very time-consuming. Also, it is clear that the parameter M plays a significant influence on the computational time of SDNM. Moreover, we find that $L_{max} = 1000$ is a sufficient large learning epoch to make the training algorithm converge for all tested time series data (e.g., as shown in Fig. 9 (b)).

As a preliminary experiment, we use Taguchi's method [72] to find a reasonable setting combination of these parameters. Taguchi's method tests part of the possible combinations among factors and levels instead of full factorial analysis, and it commits to a minimum of experimental runs and best estimation of the factor main effects over the process [73]. The number of levels for each of the five factors (i.e., the user-defined parameters) is set as follows: four levels for the number of dendrites, that is $M = 1, 3, 5, 10$; four levels for the parameter k, that is $k = 1, 3, 5, 10$; four levels for the parameter k_{soma}, that is $k_{soma} = 1, 3, 5, 10$; four levels for the parameter θ_{soma}, that is $\theta = 0, 0.3, 0.5, 0.9$; and four levels for the BP learning rate η, that is $\eta = 0.005, 0.01, 0.05, 0.1$, respectively. A full factorial design of experiment should result in a total of $4^5 = 1024$ experiments. In contrast with the full factorial analysis, the Taguchi's method uses the orthogonal arrays reducing the number of experimental runs, and controlling the cost of time, manpower and materials, effectively. Thus, an orthogonal array $L_{16}(4^5)$ which contains only 16 experiments is adopted in the preliminary study.

Table 3 summarizes the experimental results based on the orthogonal array and factor assignment, where the MSE values are displayed in the form of "Mean ± Standard Deviation" over 25 runs, and computational times are average values in seconds. As a result, aiming to reduce the running time of training and forecasting, we adopt an acceptable setting of these user-defined parameters based on our preliminary experimental results, shown as: $M = 1$, $k = 5$, $k_{soma} = 5$, $\theta_{soma} = 0.5$, $\eta = 0.05$, and $L_{max} = 1000$. Nevertheless, it is worth noticing that we have to be cautious about generalizing our conclusions here until a full factorial analysis is completed.

Table 4 Experimental results the monthly tourist arrivals from **ten major source markets** to Japan.

		MLP	Elman	ANFIS	SMN	SDNM
China	*MSE*	0.041	0.052	0.020	0.021	0.019
	RF	0.53	0.46	0.75	0.73	0.77
	RP	0.11	0.04	0.66	0.65	0.68
	Time	12.6	14.7	11.9	4.9	2.9
Korea	*MSE*	0.032	0.025	0.027	0.018	0.015
	RF	0.71	0.78	0.76	0.80	0.82
	RP	0.34	0.43	0.55	0.69	0.73
	Time	12.3	14.5	11.6	4.5	2.7
H. K.	*MSE*	0.045	0.050	0.038	0.029	0.021
	RF	0.64	0.44	0.57	0.71	0.80
	RP	0.17	0.12	0.22	0.45	0.71
	Time	12.8	14.6	11.8	4.8	2.8
Thail.	*MSE*	0.041	0.037	0.039	0.026	0.019
	RF	0.60	0.69	0.77	0.78	0.83
	RP	0.25	0.48	0.63	0.69	0.75
	Time	12.9	14.5	11.7	4.9	2.8
Taiwan	*MSE*	0.034	0.037	0.019	0.020	0.013
	RF	0.78	0.73	0.80	0.83	0.86
	RP	0.66	0.69	0.71	0.68	0.82
	Time	12.8	14.3	11.8	4.6	2.8
Sing.	*MSE*	0.021	0.011	0.008	0.008	0.006
	RF	0.73	0.88	0.92	0.95	0.97
	RP	0.55	0.80	0.86	0.89	0.92
	Time	12.4	14.1	11.7	4.5	2.7
Austr.	*MSE*	0.028	0.029	0.018	0.017	0.015
	RF	0.79	0.77	0.80	0.82	0.85
	RP	0.65	0.64	0.72	0.80	0.83
	Time	12.5	14.2	11.9	4.6	2.8
USA	*MSE*	0.031	0.036	0.019	0.021	0.019
	RF	0.65	0.71	0.79	0.81	0.82
	RP	0.63	0.56	0.74	0.62	0.78
	Time	12.4	14.5	11.7	4.6	2.7
Canada	*MSE*	0.029	0.031	0.016	0.019	0.016
	RF	0.80	0.73	0.83	0.81	0.85
	RP	0.71	0.58	0.71	0.64	0.81
	Time	12.5	14.6	11.8	4.5	2.8
UK	*MSE*	0.028	0.029	0.015	0.018	0.013
	RF	0.76	0.83	0.88	0.78	0.90
	RP	0.69	0.67	0.84	0.69	0.84
	Time	12.6	14.7	11.9	4.5	2.8

We use three assessments to evaluate the performance of SDNM, and compare SDNM with the traditional MLP network model [74], the Elman neural network [75], the ANFIS [76] and the SMN [47]. The three assessment criterions are calculated based on Eqs. (25) and (26).

- The mean square error (MSE) of the predictor for the normalized data is:

$$MSE = \frac{1}{2n}\sum_{i=1}^{n}(O_i - T_i)^2 \tag{25}$$

- The correlation coefficient of fitting (RF) and the correlation coefficient of prediction (RP) is calculated for the training phase and predication phase, respectively:

$$R = \frac{\sum_{i=1}^{n}(T_i - \overline{T_i})(O_i - \overline{O_i})}{\sqrt{\sum_{i=1}^{n}(T_i - \overline{T_i})^2 \sum_{i=1}^{n}(O_i - \overline{O_i})^2}} \tag{26}$$

where O_i is the vector of the output of the used predication

Table 5 Experimental results the monthly tourist arrivals from **six continents** to Japan.

		MLP	Elman	ANFIS	SMN	SDNM
Asia	*MSE*	0.052	0.047	0.029	0.039	0.020
	RF	0.31	0.45	0.72	0.72	0.78
	RP	0.11	0.19	0.60	0.59	0.62
	Time	12.7	14.9	11.9	4.7	2.8
Europe	*MSE*	0.036	0.031	0.018	0.021	0.013
	RF	0.79	0.75	0.83	0.76	0.88
	RP	0.43	0.51	0.76	0.53	0.80
	Time	12.8	14.8	11.8	4.8	2.7
Africa	*MSE*	0.031	0.025	0.017	0.018	0.013
	RF	0.69	0.76	0.86	0.91	0.93
	RP	0.55	0.62	0.78	0.83	0.87
	Time	12.8	14.8	11.8	4.8	2.8
N. Ame.	*MSE*	0.025	0.022	0.019	0.020	0.017
	RF	0.84	0.80	0.85	0.74	0.84
	RP	0.77	0.78	0.81	0.72	0.77
	Time	12.9	14.9	11.8	4.9	2.9
S. Ame.	*MSE*	0.031	0.029	0.021	0.018	0.016
	RF	0.79	0.89	0.82	0.79	0.89
	RP	0.65	0.73	0.76	0.73	0.81
	Time	12.8	14.8	11.8	4.9	2.9
Oceania	*MSE*	0.033	0.029	0.023	0.018	0.017
	RF	0.78	0.79	0.75	0.85	0.87
	RP	0.62	0.75	0.67	0.76	0.80
	Time	12.8	14.8	11.9	4.8	2.8

model, T_i is the vector of the true values, and n is the number of testing data samples ($P = 168$ in the training phase and $n = 60$ in the predication phase).

We implement all predication models for 25 independent runs and the average performance values are summarized in Tables 4 and 5 for the ten major source market data and six continents data, respectively. In the preliminary experiment, we empirically adjust the network size together with the parameter values to make the compared models achieve their roughly best performance. These parameter settings in MLP, Elman, ANFIS and SMN also follow the general suggestions in the original research paper and the obtained preliminary experimental data. From Tables 4 and 5, it is clear that all the *MSE* obtained by SDNM are less than 0.021, thereby demonstrating the high accuracy of the predictions. The *RF* and *RP* values are higher than those obtained using MLP, Elman, ANFIS, and SMN with the same training data and test data. The computational time consumed by SDNM is the least among the five compared models.

Moreover, we plot a typical running result for the tourism time series of Korea in Fig. 9, where (a) depicts the data fitting graphs of training and predication; (b) gives a convergence graph of the training phase; (c) illustrates the correlation coefficient of fitting; and (d) is the correlation coefficient of prediction. From this figure, we can find that both training and predication the outputs values obtained by SDNM are quite near the actual values, and a quick convergence is acquired, suggesting that the SDNM is somewhat easy to be trained. Relative high values of the correlation coefficients in training phase ($RF = 0.8598$) and predication phase ($RP = 0.75175$) can be obtained, verifying that the proposed model ban be utilized with great confidence. All in all, from the experimental results it can be said that SNDM outperforms its competitor models in terms of predication accuracy and computational time.

5. Conclusions

In this study, we presented a short-term forecasting model based on a single dendritic neuron model (SDNM) for the tourism arrivals predication. First, chaotic properties of the tourism time series were confirmed using three classic indicators in the Takens's theorem, including the time delay, the embedding dimension, and the maximum Lyapunov exponent. Then SDNM was used to perform the predication based on the reconstruction technique of phase space. Experimental results showed the model's high prediction accuracy and fitting effect. Performance comparisons demonstrated the superiority of SDNM.

The contributions of this study lie in three aspects. Theoretically it strengthens the assumption that a neural network model performs better than linear models when predicting nonlinear variables [10], [13], [24]. From the application perspective, SDNM based on PSR provides an effect alternative to learn the chaotic propensities of tourism time series. In practice, the comparative experiment results might give some insights into the selection of neural models for decision makers.

This study opens the door to the following future research. First, more applications should be made on optimization, classification, and predication problems for SDNM to further verify its information processing capacity. Second, settings of the user-defined parameters need to be investigated systematically and some self-adaptive setting mechanisms should be developed. Last but not least, the hardware implementation of the approximated SDNM [45] can also be realized.

Acknowledgments

This research was partially supported by the National Natural Science Foundation of China (Grant Nos. 11572084, 11472061, and 61472284), the Shanghai Rising-Star Program (No. 14QA1400100), the Natural Science Foundation Programs of Shanghai (No. 13ZR1443100), the program of Further Accelerating the Development of Chinese Medicine Three Year Action of Shanghai (2014-2016) No. ZY3-CCCX-3-6002, and JSPS KAKENHI Grant No. 15K00332 (Japan).

References

[1] http://www.jnto.go.jp/.
[2] http://www.tourism.jp/.
[3] H. Qu and H.Q. Zhang, "Projecting international tourist arrivals in east asia and the pacific to the year 2005," Journal of Travel Research, vol.35, no.1, pp.27–34, 1996.
[4] C. Goh and R. Law, "Modeling and forecasting tourism demand for arrivals with stochastic nonstationary seasonality and intervention," Tourism Management, vol.23, no.5, pp.499–510, 2002.

[5] R. Law, "Initially testing an improved extrapolative hotel room occupancy rate forecasting technique," Journal of Travel & Tourism Marketing, vol.16, no.2-3, pp.71–77, 2004.
[6] C.J.S.C. Burger, M. Dohnal, M. Kathrada, and R. Law, "A practitioners guide to time-series methods for tourism demand forecasting — a case study of durban, south africa," Tourism Management, vol.22, no.4, pp.403–409, 2001.
[7] F.-L. Chu, "Forecasting tourism demand: a cubic polynomial approach," Tourism Management, vol.25, no.2, pp.209–218, 2004.
[8] J.H. Kim and T. Ngo, "Modelling and forecasting monthly airline passenger flows among three major australian cities," Tourism Economics, vol.7, no.4, pp.397–412, 2001.
[9] H. Song and G. Li, "Tourism demand modelling and forecasting—a review of recent research," Tourism Management, vol.29, no.2, pp.203–220, 2008.
[10] E. Olmedo, "Comparison of near neighbour and neural network in travel forecasting," Journal of Forecasting, vol.35, no.3, pp.217–223, 2016.
[11] F.-L. Chu, "A piecewise linear approach to modeling and forecasting demand for macau tourism," Tourism Management, vol.32, no.6, pp.1414–1420, 2011.
[12] K.-H. Huarng, T.H.-K. Yu, and F.S. Parellada, "An innovative regime switching model to forecast taiwan tourism demand," The Service Industries Journal, vol.31, no.10, pp.1603–1612, 2011.
[13] X. Hong, R.J. Mitchell, S. Chen, C.J. Harris, K. Li, and G.W. Irwin, "Model selection approaches for non-linear system identification: a review," International Journal of Systems Science, vol.39, no.10, pp.925–946, 2008.
[14] U. Thissen, R. van Brakel, A.P. de Weijer, W.J. Melssen, and L.M.C. Buydens, "Using support vector machines for time series prediction," Chemometrics and Intelligent Laboratory Systems, vol.69, no.1, pp.35–49, 2003.
[15] P.-F. Pai, W.-C. Hong, P.-T. Chang, and C.-T. Chen, "The application of support vector machines to forecast tourist arrivals in barbados: An empirical study," International Journal of Management, vol.23, no.2, p.375, 2006.
[16] W.-C. Hong, Y. Dong, L.-Y. Chen, and S.-Y. Wei, "Svr with hybrid chaotic genetic algorithms for tourism demand forecasting," Applied Soft Computing, vol.11, no.2, pp.1881–1890, 2011.
[17] C.-H. Wang, "Predicting tourism demand using fuzzy time series and hybrid grey theory," Tourism management, vol.25, no.3, pp.367–374, 2004.
[18] C. Goh and R. Law, "Incorporating the rough sets theory into travel demand analysis," Tourism Management, vol.24, no.5, pp.511–517, 2003.
[19] C. Goh, R. Law, and H.M.K. Mok, "Analyzing and forecasting tourism demand: A rough sets approach," Journal of Travel Research, vol.46, no.3, pp.327–338, 2008.
[20] M. Alvarez-Diaz, J. Mateu-Sbert, and J. Rossello-Nadal, "Forecasting tourist arrivals to balearic islands using genetic programming," International Journal of Computational Economics and Econometrics, vol.1, no.1, pp.64–75, 2009.
[21] R. Law and N. Au, "A neural network model to forecast japanese demand for travel to hong kong," Tourism Management, vol.20, no.1, pp.89–97, 1999.
[22] R. Law, "Back-propagation learning in improving the accuracy of neural network-based tourism demand forecasting," Tourism Management, vol.21, no.4, pp.331–340, 2000.
[23] V. Cho, "A comparison of three different approaches to tourist arrival forecasting," Tourism Management, vol.24, no.3, pp.323–330, 2003.
[24] S.C. Kon and L.W. Turner, "Neural network forecasting of tourism demand," Tourism Economics, vol.11, no.3, pp.301–328, 2005.
[25] A. Palmer, J.J. Montano, and A. Sesé, "Designing an artificial neural network for forecasting tourism time series," Tourism Management, vol.27, no.5, pp.781–790, 2006.
[26] C.-F. Chen, M.-C. Lai, and C.-C. Yeh, "Forecasting tourism demand based on empirical mode decomposition and neural network," Knowledge-Based Systems, vol.26, pp.281–287, 2012.
[27] O. Claveria, E. Monte, and S. Torra, "Tourism demand forecasting with neural network models: different ways of treating information," International Journal of Tourism Research, vol.17, no.5, pp.492–500, 2015.
[28] L. Wang, Y. Zeng, and T. Chen, "Back propagation neural network with adaptive differential evolution algorithm for time series forecasting," Expert Systems with Applications, vol.42, no.2, pp.855–863, 2015.
[29] K.-Y. Chen and C.-H. Wang, "Support vector regression with genetic algorithms in forecasting tourism demand," Tourism Management, vol.28, no.1, pp.215–226, 2007.
[30] M. Khashei, S.R. Hejazi, and M. Bijari, "A new hybrid artificial neural networks and fuzzy regression model for time series forecasting," Fuzzy Sets and Systems, vol.159, no.7, pp.769–786, 2008.
[31] K.-Y. Chen, "Combining linear and nonlinear model in forecasting tourism demand," Expert Systems with Applications, vol.38, no.8, pp.10368–10376, 2011.
[32] P.-F. Pai, K.-C. Hung, and K.-P. Lin, "Tourism demand forecasting using novel hybrid system," Expert Systems with Applications, vol.41, no.8, pp.3691–3702, 2014.
[33] G. Zhang, B.E. Patuwo, and M.Y. Hu, "Forecasting with artificial neural networks: The state of the art," International Journal of Forecasting, vol.14, no.1, pp.35–62, 1998.
[34] M. Ardalani-Farsa and S. Zolfaghari, "Chaotic time series prediction with residual analysis method using hybrid elman–narx neural networks," Neurocomputing, vol.73, no.13-15, pp.2540–2553, 2010.
[35] V. Cho, "A study on the temporal dynamics of tourism demand in the asia pacific region," International Journal of Tourism Research, vol.11, no.5, pp.465–485, 2009.
[36] J.P. Teixeira and P.O. Fernandes, "Tourism time series forecast-different ann architectures with time index input," Procedia Technology, vol.5, pp.445–454, 2012.
[37] D.H. Wolpert and W.G. Macready, "No free lunch theorems for optimization," IEEE Transactions on Evolutionary Computation, vol.1, no.1, pp.67–82, 1997.
[38] R. Sitte and J. Sitte, "Analysis of the predictive ability of time delay neural networks applied to the s&p 500 time series," IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews, vol.30, no.4, pp.568–572, 2000.
[39] X. Lin, Z. Yang, and Y. Song, "Short-term stock price prediction based on echo state networks," Expert systems with applications, vol.36, no.3, pp.7313–7317, 2009.
[40] R. Chandra, "Competition and collaboration in cooperative coevolution of elman recurrent neural networks for time-series prediction," IEEE Transactions on Neural Networks and Learning Systems, vol.26, no.12, pp.3123–3136, 2015.
[41] H. Jaeger and H. Haas, "Harnessing nonlinearity: Predicting chaotic systems and saving energy in wireless communication," Science, vol.304, no.5667, pp.78–80, 2004.
[42] Y. Todo, H. Tamura, K. Yamashita, and Z. Tang, "Unsupervised learnable neuron model with nonlinear interaction on dendrites," Neural Networks, vol.60, pp.96–103, 2014.
[43] A. Destexhe and E. Marder, "Plasticity in single neuron and circuit computations," Nature, vol.431, no.7010, pp.789–795, 2004.
[44] Z. Sha, L. Hu, Y. Todo, J. Ji, S. Gao, and Z. Tang, "A breast cancer classifier using a neuron model with dendritic nonlinearity," IEICE Trans. Inf. & Syst., vol.E98-D, no.7, pp.1365–1376, 2015.
[45] J. Ji, S. Gao, J. Cheng, Z. Tang, and Y. Todo, "An approximate logic neuron model with a dendritic structure," Neurocomputing, vol.173, pp.1775–1783, 2016.
[46] F. Takens, "Detecting strange attractors in turbulence," Dynamical Systems and Turbulence, Warwick 1980, vol.898, pp.366–381, Springer, 1981.
[47] R.N. Yadav, P.K. Kalra, and J. John, "Time series prediction with single multiplicative neuron model," Applied Soft Computing, vol.7, no.4, pp.1157–1163, 2007.

[48] L. Zhao and Y. Yang, "Pso-based single multiplicative neuron model for time series prediction," Expert Systems with Applications, vol.36, no.2, pp.2805–2812, 2009.
[49] C. Zhang, W. Wu, and Y. Xiong, "Convergence analysis of batch gradient algorithm for three classes of sigma-pi neural networks," Neural Processing Letters, vol.26, no.3, pp.177–189, 2007.
[50] R.P. Costa and P.J. Sjöström, "One cell to rule them all, and in the dendrites bind them," Frontiers in Synaptic Neuroscience, vol.3, Article 5, 2011.
[51] C. Koch, T. Poggio, and V. Torre, "Nonlinear interactions in a dendritic tree: localization, timing, and role in information processing," Proceedings of the National Academy of Sciences, vol.80, no.9, pp.2799–2802, 1983.
[52] N. Brunel, V. Hakim, and M.J. Richardson, "Single neuron dynamics and computation," Current opinion in neurobiology, vol.25, pp.149–155, 2014.
[53] L.F. Abbott and W.G. Regehr, "Synaptic computation," Nature, vol.431, no.7010, pp.796–803, 2004.
[54] A. Destexhe and E. Marder, "Plasticity in single neuron and circuit computations," Nature, vol.431, no.7010, pp.789–795, 2004.
[55] Y.-N. Jan and L.Y. Jan, "Branching out: mechanisms of dendritic arborization," Nature Reviews Neuroscience, vol.11, no.5, pp.316–328, 2010.
[56] K. Swingler, Applying neural networks: a practical guide, Morgan Kaufmann, 1996.
[57] H. Demuth and M. Beale, Neural network toolbox for use with MATLAB, MathWorks Inc., Natick, MA, 1993.
[58] M.H. Hassoun, Fundamentals of artificial neural networks, MIT press, 1995.
[59] J.C. Principe, N.R. Euliano, and W. Lefebvre, Neural and adaptive systems: fundamentals through simulations, Wiley, New York, 1999.
[60] B.W. Mel and C. Koch, "Sigma-pi learning: On radial basis functions and cortical associative learning.," NIPS, pp.474–481, 1989.
[61] M.P. Jadi, B.F. Behabadi, A. Poleg-Polsky, J. Schiller, and B.W. Mel, "An augmented two-layer model captures nonlinear analog spatial integration effects in pyramidal neuron dendrites," Proceedings of the IEEE, vol.102, no.5, pp.782–798, 2014.
[62] F. Gabbiani, H.G. Krapp, C. Koch, and G. Laurent, "Multiplicative computation in a visual neuron sensitive to looming," Nature, vol.420, no.6913, pp.320–324, 2002.
[63] C. Koch and I. Segev, "The role of single neurons in information processing," Nature Neuroscience, vol.3, pp.1171–1177, 2000.
[64] T. Jiang, D. Wang, J. Ji, Y. Todo, and S. Gao, "Single dendritic neuron with nonlinear computation capacity: A case study on xor problem," 2015 IEEE International Conference on Progress in Informatics and Computing (PIC), pp.20–24, 2015.
[65] R. Legenstein and W. Maass, "Branch-specific plasticity enables self-organization of nonlinear computation in single neurons," Journal of Neuroscience, vol.31, no.30, pp.10787–10802, 2011.
[66] J. Sola and J. Sevilla, "Importance of input data normalization for the application of neural networks to complex industrial problems," IEEE Transactions on Nuclear Science, vol.44, no.3, pp.1464–1468, 1997.
[67] P. Grassberger and I. Procaccia, "Estimation of the kolmogorov entropy from a chaotic signal," Physical Review A, vol.28, no.4, pp.2591–2593, 1983.
[68] A.M. Fraser and H.L. Swinney, "Independent coordinates for strange attractors from mutual information," Physical Review A, vol.33, no.2, pp.1134–1140, 1986.
[69] A. Wolf, J.B. Swift, H.L. Swinney, and J.A. Vastano, "Determining lyapunov exponents from a time series," Physica D: Nonlinear Phenomena, vol.16, no.3, pp.285–317, 1985.
[70] P. Shang, X. Na, and S. Kamae, "Chaotic analysis of time series in the sediment transport phenomenon," Chaos, Solitons & Fractals, vol.41, no.1, pp.368–379, 2009.
[71] H.D.I. Abarbanel, "Analysis of observed chaotic data," Institute for Nonlinear Science, Springer, 1996.
[72] G. Taguchi, R. Jugulum, and S. Taguchi, Computer-based robust engineering: essentials for DFSS, ASQ Quality Press, 2004.
[73] J.F.C. Khaw, B.S. Lim, and L.E.N. Lim, "Optimal design of neural networks using the taguchi method," Neurocomputing, vol.7, no.3, pp.225–245, 1995.
[74] D.E. Rumelhart, G.E. Hinton, and R.J. Williams, Learning internal representations by error propagation, Institute for Cognitive Science, University of California, San Diego, 1985.
[75] J.L. Elman, "Finding structure in time," Cognitive Science, vol.14, no.2, pp.179–211, 1990.
[76] C. Vairappan, H. Tamura, S. Gao, and Z. Tang, "Batch type local search-based adaptive neuro-fuzzy inference system (anfis) with self-feedbacks for time-series prediction," Neurocomputing, vol.72, no.7-9, pp.1870–1877, 2009.

Wei Chen received his M.S. degree from University of Toyama, Toyama, Japan. He is currently a Ph.D. candidate at the Graduate School of Science and Engineering for Education, University of Toyama, Japan. His research interests are in the areas of evolutionary computation, optimization, pattern recognition and neural networks.

Jian Sun received the B.S. degree from Nanjing Normal University, Nanjing, China in 2000 and an M.S. degree from Yangzhou University, Yangzhou, China, in 2010. Now she is working toward the D.E. degree at University of Toyama, Toyama, Japan. Her main research interests are computational intelligence, multiple-valued logic and artificial neural networks.

Shangce Gao received the B.S. degree from Southeast University, Nanjing, China in 2005, and the M.S. and Ph. D. degrees in intellectual information systems and innovative life science from University of Toyama, Toyama, Japan in 2008 and 2011, respectively. He is currently an Associate Professor with the Faculty of Engineering, University of Toyama, Toyama, Japan. From 2011 to 2012, he was an associate research fellow with the Key Laboratory of Embedded System and Service Computing, Ministry of Education, Tongji University, Shanghai, China. From 2012 to 2014, he was an associate professor with the College of Information Sciences and Technology, Donghua University, Shanghai, China. His main research interests include computational intelligence, nature-inspired technologies, swarm intelligence, and neural networks for optimization and real-world applications. He was a recipient of the Shanghai Rising-Star Scientist award, the Chen-Guang Scholar of Shanghai award, the Outstanding Academic Performance Award of IEICE, and the Outstanding Self-financed Students Abroad Award of Chinese Government, and the Outstanding Academic Achievement Award of IPSJ.

Jiu-Jun Cheng received his Ph.D. degree from Beijing University of Posts and Telecommunications in 2006. He is presently a professor of Tongji University, Shanghai, China. In 2009, he was a Visiting Professor at the Aalto University, Espoo, Finland. He has over 40 publications including conference and journal papers. His research interests span the area of neural computing, mobile computing and social network with a focus on mobile/Internet interworking, service computing, and Internet of Vehicles.

Jiahai Wang received the B.S. degree from Gannan Teachers College, Ganzhou, China, in 1999, the M.S. degree from Shandong University, Jinan, China, in 2001, and the Ph.D. degree from University of Toyama, Toyama, Japan, in 2005. In 2005, he joined Sun Yat-sen University, Guangzhou, China, where he is currently an Associate Professor with the Department of Computer Science. His main research interests include computational intelligence and its applications.

Yuki Todo received the B.S. degree from Zhejiang University, Zhejiang, China, the M.S. degree from Beijing University of Posts and Telecommunications, Beijing, China, and the D.E. degree from Kanazawa University, Kanazawa, Japan, in 1983, 1986, and 2005, respectively. From 1987 to 1989, she was an Assistant Professor with the Institute of Microelectronics, Shanghai Jiaotong University, Shanghai, China. From 1989 to 1990, she was a research student at Nagoya University, Nagoya, Japan. From 1990 to 2000, she was a Senior Engineer with Sanwa Newtech Inc., Miyazaki, Japan. From 2000 to 2011, she worked with Tateyama Systems Institute, Toyama, Japan. In 2012, she joined Kanazawa University, where she is now an Associate Professor with the Faculty of Electrical and Computer Engineering. Her current research interests include multiple-valued logic, neural networks, and optimization.