Modeling time series by aggregating multiple fuzzy cognitive maps

Tianming Yu¹, Qunfeng Gan² and Guoliang Feng¹

¹ School of Automation Engineering, Northeast Electric Power University, Jilin, Jilin, China
² College of Information and Control Engineering, Jilin Institute of Chemical Technology, Jilin, Jilin, China

ABSTRACT

Background: The real time series is affected by various combinations of influences, consequently, it has a variety of variation modality. It is hard to reflect the variation characteristic of the time series accurately when simulating time series only by a single model. Most of the existing methods focused on numerical prediction of time series. Also, the forecast uncertainty of time series is resolved by the interval prediction. However, few researches focus on making the model interpretable and easily comprehended by humans.

Methods: To overcome this limitation, a new prediction modelling methodology based on fuzzy cognitive maps is proposed. The bootstrap method is adopted to select multiple sub-sequences at first. As a result, the variation modality are contained in these sub-sequences. Then, the fuzzy cognitive maps are constructed in terms of these sub-sequences, respectively. Furthermore, these fuzzy cognitive maps models are merged by means of granular computing. The established model not only performs well in numerical and interval predictions but also has better interpretability.

Results: Experimental studies involving both synthetic and real-life datasets demonstrate the usefulness and satisfactory efficiency of the proposed approach.

Subjects Data Mining and Machine Learning, Data Science
Keywords Time series, Fuzzy cognitive maps, Granular computing

INTRODUCTION

Time series data modeling and prediction has always been a classic research topic, which can predict the future by analyzing, understanding and modeling historical data. Time series prediction has been widely used in many fields, such as economy, industry, climate prediction (Zeng & Khondker, 2016), medical (Ma et al., 2018; Zheng et al., 2018; Nishii et al., 2016), and so on. Its significance lies in its ability to provide the basis for decision-making in these fields effectively (Weigend & Gerschenfeld, 1994). Linear system theory, stochastic process theory (Yin et al., 2019; Ren, Zhang & Zhang, 2019; Zhang & Wang, 2021), black box theory and other methods have been used by researchers to develop many classic time series numerical models, such as autoregressive integrated moving average (ARIMA) (Lee & Tong, 2011), neural network models (Zhang, 2003; Khashei, Bijari & Ardali, 2012), support vector machines (Cao & Tay, 2003; Chaâbane, 2014), Bayesian networks (Guo, Liu & Sun, 2016) and other models. These models have been widely used in various fields and show good performance in numerical prediction.
However, these numerical models are difficult to be understood due to their low interpretability. In order to make the time series model interpretable and maintain the robustness of uncertainty, Fuzzy Set Theory provides a feasible choice. Fuzzification is introduced into modeling the time series, and then the fuzzy time series model is constructed (Song & Chissom, 1993). The main steps of the fuzzy time series model are as follows: the division of the interval of the domain of definition, the language description of the interval, the fuzzification of the time series, the establishment of the fuzzy logic relationship among variables of fuzzy time series, the calculation of predicted value and defuzzification (Singh, 2017; Bose & Mali, 2019). In addition, in some fields, only an approximation value is enough, so that it is not necessary to know the accurate value of the time series. For example, for daily temperature forecasts, the minimum and maximum values of temperature are more instructive. In the stock market, investors are more interested in the range of decline and rise in the future. In these fields, people prefer to model and forecast time series at the level of symbol or information granularity, rather than just predict the accurate value. In order to construct a higher-level representation method of the time series model, a time series modeling method based on granular computing is further proposed by researchers (Lu, Chen & Pedrycz, 2015; Froelich & Pedrycz, 2017). Granular computing (Bargiela & Pedrycz, 2008) is a human-centered information processing framework platform, which uses "reasonable information granulation principle" (Pedrycz, 2011; Pedrycz & Homenda, 2013) to divide complex and abstract information into simple and understandable information granules according to certain rules. This method does not excessively pursue the accurate value of the model, but mediates the “precision” and “interpretability” of the model so that the dynamic behavior of time series is easier to be understood.

As a soft computing tool, Fuzzy Cognitive Maps (FCMs) realize the reasoning process based on knowledge representation and can capture the behavior of a given dynamic system, so it is used to describe and model complex systems (Papageorgiou & Salmeron, 2012; Felix et al., 2019). FCMs are composed of concept nodes and directed weights. Nodes represent the main behavior characteristics of the system, and weights reflect the causal relationship between concept nodes. Based on the knowledge reasoning ability of FCMs, Stach, Kurgan & Pedrycz (2008) used real coded genetic algorithm to learn fuzzy cognitive map and predict time series at numerical and linguistic levels. Lu, Yang & Liu (2014) employed the prior states of the nodes to forecast the current states to improve the performance of FCMs models, which is termed as high-order fuzzy cognitive maps. Pedrycz, Jastrzebska & Homenda (2016) designed a scheme that uses information granules to describe numerical time series. In the space of amplitude and amplitude change of time series, the fuzzy c-means clustering algorithm is used to cluster the data, and each cluster center is used as a node of FCMs to model and predict the time series. Salmeron & Froelich (2016) used strategy of dynamic optimization to update the weight of FCMs dynamically to improve the prediction accuracy of time series model. This method dynamically adjusts the weights of FCMs according to the local characteristics of time series. Froelich & Pedrycz (2017) used the principle of reasonable granularity and the
principle of fuzzy c-means clustering to approximate the granularity of time series. In this way, a compromise between particularity and universality is designed when generating approximate values of time series. In addition, the time series can be more easily understood by human beings after granulating the obtained particles and assigning language items, which is a character that the method of directly analyzing the original data does not have. Feng et al. (2021) presented a rapid and robust FCMs learning method with maximum entropy, and developed a long-term time series prediction FCMs model based on this learning method (Feng et al., 2021).

When modeling time series, the existing methods usually model the entire time series as a whole, that is, assuming that the time series has only one pattern of variation. However, the change patterns of time series in different time periods and intervals are not the same. In addition, the main time series may also be derived from multiple objects. Therefore, it is difficult to mine all the change characteristics of time series by modeling the whole time series directly. If each change mode of time series can be modeled, the knowledge representation of time series will be more comprehensive. In this paper, a large number of short sequences are selected in a given time series by using a random strategy with putting back in order to contain various variation modes in the time series. Then, Fuzzy Cognitive Maps is used to model each subsequence. The output of each model is fused by the method of granulation, and finally, the output of the time series is obtained. The proposed method can not only obtain the numerical prediction value of the time series but also obtain the prediction interval. Furthermore, it has semantic descriptions, which makes it easier for people to understand the change characteristics of time series.

**MATERIALS & METHODS**

**Fuzzy cognitive maps**

The topological structure of FCMs is a weighted directed graph with feedback loops, which describe the characteristics of the physical system through nodes and weight edges. As shown in Fig. 1, a typical FCMs is composed of weight edges between conceptual nodes.
and nodes. These nodes represent the main variables in the system that to be studied, and the weighted edges represent the degree of mutual influence among nodes.

The semantics of standard FCMs can be represented by a 4-tuple \((C, W, A, f)\) (Felix et al., 2019). Suppose that FCMs has n nodes, \(C = \{C_1, C_2, \cdots, C_n\}\) is a set of \(n\) conceptual nodes, and \(W\) is the weight matrix of \(n \times n\) dimensional:

\[
W = \begin{bmatrix}
    w_{11} & w_{12} & \cdots & w_{1n} \\
    w_{21} & w_{22} & \cdots & w_{2n} \\
    \vdots & \vdots & \ddots & \vdots \\
    w_{n1} & w_{n2} & \cdots & w_{nn}
\end{bmatrix}
\]

(1)

where plus, minus and value size of \(w_{ij}\) \([-1, 1]\) respectively reflect the direction and degree of causality between nodes. \(w_{ij} > 0\) indicates that \(C_j\) will increase with the increase of \(C_i\), and the degree of influence is \(w_{ij}\), and vice versa. \(w_{ij} < 0\) denotes that \(C_j\) will decrease with the increase of \(C_i\), and the degree of influence is \(-w_{ij}\), and vice versa. \(w_{ij} = 0\) means that there is no causal relationship between \(C_i\) and \(C_j\). \(A: C_i \rightarrow A_i(t), A_i(t)\) is the state value of the node \(C_i\) at the moment \(t\), the value changes dynamically. \(f\) is the activation function, which is a nonlinear monotone increasing function, it integrates the input state values of all nodes associated with the target node into the definitional domain of the activation function. The state value of the node \(C_j\) at the time \(t + 1\) can be obtained by the following formula:

\[
A_j(t + 1) = f\left(\sum_{i=1}^{n} A_i(t)w_{ij}\right)
\]

(2)

The Sigmoid function is a common activation function:

\[
f(x) = \frac{1}{1 + e^{-\lambda x}}
\]

(3)

This activation function limits the state value of the node to the interval \([0, 1]\), where \(\lambda > 0\) determines the steepness of the activation function.

**Granular computing**

Granular computing was first proposed by Zadeh (1997, 2008), and the general definition of fuzzy information granule was given as follows:

\[
g = (x \text{ is } G) \text{ is } \mu
\]

(4)

where \(x\) denotes a variable on the discourse universe \(U\), \(G\) is a fuzzy subset of \(U\), \(\mu\) is the probability that \(x\) belongs to the fuzzy set \(G\). \(g\) is an information granule, its lexeme is “the probability level of \(x\) belongs to the fuzzy set \(G\) is \(\mu\)”. Granular computing is embodied in the process of data abstraction and information knowledge representation, different levels of abstraction can be represented by different sizes of information granules.

Bargiela & Pedrycz (2008) and Pedrycz (2018) further developed the interpretability research of granular computing on this basis and regarded information granular as an
information processing process of “human-centered”, which was used for knowledge representation of actual models and decision-making process.

Granular computing follows the principle of justifiable granularity, that is, reasonable information granulation should represent the data as much as possible, on the other hand, data should be abstracted as much as possible and expressed semantically. In short, the principle of granular computing is manifested in two aspects: coverage and specificity (Pedrycz & Vukovich, 2001). The coverage denotes how much data is covered by the constructed information granules, it requires data to be covered by information granular as possible. The specificity means that the generated information granular should have clear semantic meaning, the constructed information granular is required to be as small as possible, so that the semantic is clearer. It can be seen that the requirements of the two are contradictory.

Suppose there is a data set $Y = \{y_1, y_2, \cdots, y_p\}$. The coverage of the information granular is quantified by determining the number of elements in the data set covered by the information granule. Assume $i \in [1, p]$, the coverage $\text{cov}$ of information granular is defined as follows:

$$\text{cov} = \text{card}\{y|y_i \in g\}$$

(5)

$\text{card}(\cdot)$ represents the cardinality, that is, how many elements of the data set $Y$, belong to the information granule $g$. The specificity is quantified by the width of information granule $g$, and the specificity $sp$ is defined as follows:

$$sp = \exp(-\alpha|U - L|)$$

(6)

where $U$ and $L$ respectively represent the upper and lower bounds of the amplitude of the data covered by the information granule. The smaller the distance $U-L$ between the upper and lower bounds, the more specific the information granule. $\alpha > 0$, the level of information granularity provides flexibility for the constructed information granular, and its value affects the specificity of the information granular.

The key to information granulation is to construct a function to make the coverage and specificity of information granular better. However, increasing the coverage of information granular leads to the decrease of specificity, and vice versa. To balance these two characteristics, the problem of constructing information granular is transformed into the following optimization problem:

$$\arg \max_{U,L} J = \text{cov} \times sp$$

(7)

**THE PROPOSED METHOD**

When fuzzy cognitive maps is used to model time series, the following four steps are generally followed: fuzzification of time series, learning of fuzzy cognitive maps, modeling and prediction, and defuzzification of predicted value (Stach, Kurgan & Pedrycz, 2008). This section will describe the whole modeling process in detail.
Fuzzification of time series

For a time series $X = \{x_1, x_2, \cdots, x_m\}$, $m$ is the number of sample data contained in the time series. A triangular fuzzy set is used to fuzzify the time series $X$. The membership function formula of the triangular fuzzy set is as follows:

$$
\mu(x; a, b, c) = \begin{cases} 
0 & x \leq a \\
\frac{x - a}{b - a} & a \leq x \leq b \\
\frac{c - x}{c - b} & b \leq x \leq c \\
0 & x \geq c 
\end{cases}
$$

(8)

where parameters $a$ and $c$ correspond to the left and right vertices of the triangle, $b$ is the middle vertex of the triangle, and $a \leq b \leq c$.

Suppose that the definitional domain of time series $X$ is divided into continuous and disjoint intervals by $n$ points $\{P_1, P_2, \cdots, P_n\}$, and each division point is a fuzzy linguistic variable. Then the membership degree of sample data $x_i$ to these fuzzy sets is expressed as $\{\mu_1(x_i), \mu_2(x_i), \cdots, \mu_n(x_i)\}$.

After the time series $X = \{x_1, x_2, \cdots, x_m\}$ is fuzzified by triangular fuzzy set, the matrix $U$ formed by the fuzzy time series can be obtained as follows:

$$
U = \begin{bmatrix}
u_{11} & u_{12} & \cdots & u_{1n} \\
u_{21} & u_{22} & \cdots & u_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
u_{m1} & u_{m2} & \cdots & u_{mn}
\end{bmatrix}
$$

(9)

where $m$ is the number of samples of time series $X$ and $n$ is the number of the divided fuzzy sets. The elements in each row of $U$ are the fuzzy membership values of the corresponding time series samples, and the sum of elements in each row is 1. The vector of triangular fuzzy linguistic variables is as follows:

$$
P = [P_1, P_2, \cdots, P_n]
$$

(10)

Modeling with fuzzy cognitive maps

After the fuzzy time series is determined, it can be used for learning FCMs. As a symbolic reasoning mechanism, the reasoning Eq. (2) of FCMs can be written as follows:

$$
A_j(t + 1) = f\left( A(t)W_j \right)
$$

(11)

where $A(t) = [A_1(t), A_2(t), \cdots, A_n(t)]$ are the state values of all nodes of FCMs at time $t$, $A_j(t + 1)$ is the state value of the $j$-th node of FCMs at the time $t + 1$, and $W_j = [w_{1j}, w_{2j}, \cdots, w_{nj}]^T$ is the $j$-th column of the weight matrix $W$. As activation function, a Sigmoid function has a unique inverse function, so after the Eq. (11) was inversely transformed, it can be expressed as:

$$
f^{-1}(A_j(t + 1)) = A(t)W_j
$$

(12)
The above equation is linear. Where $f^{-1}(y) = -\frac{1}{\lambda} \ln \frac{1-y}{y}$ is the inverse function of the Sigmoid function.

After the fuzzy time series matrix $U$ is obtained, it is used for learning the weight $W$ of FCMs. Let $Z = [U_1, U_2, \cdots, U_{m-1}]^T$, where $U_i = [u_{i1}, u_{i2}, \cdots, u_{in}]$, $Y_j = f^{-1}([u_{j1}, u_{j2}, \cdots, u_{jm}])$, put $Z$ and $Y_j$ into Eq. (12):

$$Y_j = ZW_j$$

(13)

$Z$ and $Y_j$ can be regarded as independent variables and dependent variables of historical data respectively. Therefore, the learning problem of FCMs weight can be transformed into the solution of the least square problem. The following objective function is constructed to solve $W_j$:

$$\arg \min_{W_j} \|ZW_j - Y_j\|_2$$

s.t. $\|W_j\|_{\infty} \leq 1$

(14)

$\|ZW_j - Y_j\|_2$ in the objective function is used to minimize the least square error between the actual value and the predicted value, to obtain a reasonable approximate solution of $W_j$. The bound term $\|W_j\|_{\infty} \leq 1$ ensures that the value of each element of the weight matrix is in the interval $[-1, 1]$, where the infinite norm $\|W_j\|_{\infty} = \max\{\|w_{1j}\|, \|w_{2j}\|, \cdots, \|w_{nj}\|\}$. The above formula is a standard convex optimization problem with linear constraints. At present, there are many convex optimization techniques to solve the problem, such as the commonly used interior point method and primal-dual method (Boyd & Vandenberghe, 2004).

The prediction of time series is based on historical data to infer future data. In this paper, the reasoning mechanism of FCMs is used to predict time series. The state vector of FCMs at the time $t$ is given as $U_t = [u_{t1}, u_{t2}, \cdots, u_{tn}]$, once the weight matrix $W$ is determined, then the predicted value $\hat{U}_{t+1}$ of the state vector at time $t + 1$ can be calculated by the following formula:

$$\hat{U}_{t+1} = f(U_tW)$$

(15)

Through the above formula, the state vector at the next moment of FCMs can be obtained, so the prediction matrix $\hat{U}$ of the state vector can be obtained. In order to obtain the final value of time series prediction, it is necessary to defuzzify the state vector, the formula is as follows:

$$\hat{x}_{t+1} = \frac{\sum_{j=1}^{n} u_{(t+1)j} P_j}{\sum_{j=1}^{n} u_{(t+1)j}}$$

(16)

where $n$ is the number of nodes of FCMs, $P_j$ is the triangular fuzzy linguistic value of the $j$-th node of FCMs, as shown in Eq. (10).
Aggregation of FCMs models

The data in time series change over time and come from various sources, that is, time series usually contain multiple modal features. To understand the characteristics of time series more deeply, it is necessary to build a model that can mine and use these modal features for modeling. Moreover, direct modeling of large time series will lead the model to be too complicated to understand. Therefore, the multi-modal modeling method is used to analyze the time series, to mine the change characteristics of the time series subsystem, and finally to represent the change characteristics of the whole time series.

Figure 2 shows the framework of the FCMs multimodal modeling system, Bootstrap is used to randomly extract $p$ subsets from the time series $X$, and each subset trains an FCMs sub-model. Then the outputs of each sub-model are fused by the information granulation method to obtain the final output value and its upper and lower limit value.

The specific steps of the multi-mode FCMs model are as follows:

(1) For the time series $X$ of length $m$, the triangular fuzzy set with the number of fuzzy sets $n$ is used for fuzzification, and the fuzzy time series $U$ is obtained.

(2) The bootstrap method is used to randomly select $p$ subsets from fuzzy time series $U$ of length $m$. It should be noted that the subsequences are not necessarily adjacent in order. In this study, the starting point of each subsequence is randomly selected, and the lengths of the subsequences are the same. Set the length of each subsequence to $k$, then the $i$-th ($i = 1, 2, \ldots, p$) subsequence can be represented as follows:

$$
\bar{U}^i = \begin{bmatrix}
\bar{u}_{11}^i & \bar{u}_{12}^i & \cdots & \bar{u}_{1n}^i \\
\bar{u}_{21}^i & \bar{u}_{22}^i & \cdots & \bar{u}_{2n}^i \\
\vdots & \vdots & \ddots & \vdots \\
\bar{u}_{k1}^i & \bar{u}_{k2}^i & \cdots & \bar{u}_{kn}^i
\end{bmatrix}
$$

(17)

(3) For each subsequence $\bar{U}^i$, the method described in the previous section is used for FCMs modeling, and $p$ FCMs models are obtained.

(4) How to fuse the output of the sub-model is the key to realize the multimodal model. Suppose that the weights of $p$ models are $\omega_1, \omega_2, \cdots, \omega_p$ respectively, the output and the weights of a model form a set of weighted data set
\( Y = \{ (y_1, \omega_1), (y_2, \omega_2), \ldots, (y_p, \omega_p) \} \), where \( \omega_i \in [0, 1] \) is the weight of data \( y_i (i = 1, 2, \ldots, p) \). Granulating the information of data set \( Y \), and the optimization objective function is constructed as follows:

\[
\arg \max_{U, L} J = \text{cov} \times \text{sp} \times \text{flag} \tag{18}
\]

where the constructed coverage function is \( \text{cov} = \sum \omega_i, y(i) \in [\hat{y}_L, \hat{y}_U], i \in [1, p] \), the special function is \( \text{sp} = \exp(-x|\hat{y}_L - \hat{y}_U|) \), \( \hat{y}_L \) and \( \hat{y}_U \) respectively represent the upper and lower bounds of the data amplitude covered by information granule. When it is used for time series prediction, the \( \text{flag} \) represents whether the actual value at the previous moment is within the upper and lower bounds:

\[
\text{flag} = \begin{cases} 
1, & y \in [\hat{y}_L, \hat{y}_U] \\
0, & \text{other}
\end{cases}
\]

The purpose of constructing the objective function is to make the weight of the data covered by the constructed information granule as large as possible, and the interval of the information granule as small as possible. When making interval prediction for time series, it is always hoped that the obtained interval value should be as small as possible, but at the same time, the actual value should be kept within the prediction interval. The physical meaning of the interval obtained by (18) is the upper and lower boundary of the prediction interval.

(5) Suppose that there are q data in the interval of the information granule obtained in the previous step, the weights of these data obtained by recalculation are \( \lambda_1, \lambda_2, \ldots, \lambda_q \), and the calculation formula of the final prediction value is:

\[
\hat{y} = \sum_{i=1}^{q} \lambda_i y_i \tag{19}
\]

where \( y_i \in [\hat{y}_L, \hat{y}_U] \), \( \hat{y}_L \) and \( \hat{y}_U \) represent the upper and lower bounds of the prediction interval respectively.

Root mean square error (RMSE) is used to evaluate the prediction performance of the model:

\[
\text{RMSE} = \sqrt{\frac{1}{N} \sum_{t=1}^{N} (x(t) - y(t))^2} \tag{20}
\]

where \( x(t) \) denotes the actual value at time \( t \), and \( y(t) \) represents the simulated value at time \( t \). Obviously, the smaller the RMSE value is, the better the prediction performance of the model is. \( N \) is the number of data.
In order to obtain the final output of the multimodal model, it is necessary to determine the weight of each sub-model. The weighting strategy uses the following three methods:

(1) Average weight

Use the arithmetic average of each sub-model output as the output of the multi-modal system, that is, the weight of $p$ sub-models is $1/p$. The weight formula $\omega_i$ of the $i$-th sub-model is as follows:

$$\omega_i = \frac{1}{p} \quad (21)$$

(2) Model weight

The RMSE of each model is calculated. The smaller the RMSE is, the greater the weight of the corresponding sub-model is. Take the reciprocal of the RMSE of each sub-model and normalize it. Take the reciprocal of each RMSE and normalize it.

$$\omega_i = \frac{1}{\sum_{i=1}^{p} 1/RMSE_i} \quad (22)$$

(3) Dynamic weight

The absolute difference between the predicted value and the actual value at the last moment is used to determine the weight of the model. In this way, the weight of each sub-model changes in real-time. The weight formula $\omega_i(t)$ of the $i$-th sub-model is as follows:

$$\omega_i(t) = \frac{1/|x_i(t-1) - y_i(t-1)|}{\sum_{i=1}^{p} 1/|x_i(t-1) - y_i(t-1)|} \quad (23)$$

After the weight of the sub-model is determined, the output of the multimodal model is calculated as follows:

$$\hat{y} = \sum_{i=1}^{p} \omega_i y_i \quad (24)$$

**RESULTS**

RMSE is used to evaluate the numerical prediction performance of the constructed model, and the formula is shown in (20). PICP (Prediction interval coverage probability) and PINAW (Prediction interval normalized average width) are used to evaluate interval prediction performance (Khosravi, Nahavandi & Creighton, 2013; Quan, Srinivasan & Khosravi, 2013), and the formula is as follows:

$$PICP = \frac{1}{N} \sum_{t=1}^{N} \delta_t \quad (25)$$

$$PINAW = \frac{1}{NR} \sum_{t=1}^{N} (\hat{y}_U - \hat{y}_L) \quad (26)$$
where \( N \) is the number of prediction samples, \( \delta_t \) is a Boolean value:

\[
\delta_t = \begin{cases} 
1, & x \in [\hat{y}_L, \hat{y}_U] \\
0, & x \notin [\hat{y}_L, \hat{y}_U]
\end{cases}
\]

\( x \) is the actual value, \( \hat{y}_L \) and \( \hat{y}_U \) are the lower and upper bounds of the prediction interval respectively, \( R \) is the range of amplitude changes of time series data. Based on these two evaluation indexes, CWC (coverage width based criterion) is constructed:

\[
CWC = PINAW \left(1 + \frac{PICP}{e^{-PICP}}\right)
\]

(27)

Among these evaluation indexes, RMSE is used to evaluate the error between the predicted value and the true value. The smaller the RMSE is, the better the performance of the prediction model is. PICP and PINAW are used to evaluate the indicators of the prediction interval. The larger the PICP, the higher the probability that the prediction interval contains the actual sample values. The smaller the PINAW is, the smaller the prediction interval is. An excellent prediction model should have the characteristics of high PICP and small PINAW, CWC, RMSE.

In order to demonstrate the working process of the method proposed in this paper and verify its effectiveness, it experiments with artificial and actual data sets. The data sets include both small and large time series:

(1) Mackey-Glass (MG). Mg is a chaotic, aperiodic, non-convergent sequence, which is obtained by the following delay differential equation:

\[
\frac{dx_t}{dt} = \frac{0.2x_{t-\tau}}{1 + x_{t-\tau}^{10}} - 0.1x_t
\]

(28)

let \( x_0 = 1.2, \tau = 17 \), when \( t < 0, x_t = 0 \), Substituting these parameters into the above formula, and the resulting sequence contains 1,201 samples, the sequence is shown in Fig. 3A.

(2) Exchange rate. A time series of the monthly exchange rate between the Australian dollar and the US dollar. The data set contains 314 samples of monthly exchange rates from July 1969 to August 1995, as shown in Fig. 3B.

(3) Vatnsdalsa. The average daily flow (\( m^3/s \)) of the Vatnsdalsa River from January 1, 1972 to December 31, 1974, there are 1,095 samples, as shown in Fig. 3C.

(4) Wind speed. The time interval of the data set is 10 min, and there are 39,195 samples, as shown in Fig. 3D.

In the experiment, the first 80% of the above time series data set samples were used as the training set and the last 20% as the test set. The parameters of the model are shown in Table 1. In order to reduce the complexity of the model, the number of nodes in each sub-FCMs model is \( n = 3 \), the number of samples in each subsequence \( k = 5 \), the number of sub-sequences randomly selected by the bootstrap method is \( p = 100 \), and the granularity level \( \alpha = 1 \).

The experimental platform is a laptop computer with a CPU of 2.3 GHz and a memory of 4G. The software is Matlab R2018a. In the experiment, the starting point of each
sampled data is different, and the change of each sub-sequence is also different. Therefore, the FCMs models trained by these subsequences are also different. The purpose of using a large number of random selection methods is to include the changes of time series data as much as possible.

According to the semantics given by the fuzzy set of the data domain, the three nodes of FCMs are defined as low amplitude, medium amplitude, and high amplitude. When the triangular fuzzy set is used for the fuzzification of the time series, the minimum, median and maximum values of the time series are selected as the linguistic variables of the
fuzzy set. Table 2 shows the specific values corresponding to the node semantics in the FCMs model corresponding to each data set. Taking the MG sequence as an example, the value corresponding to the semantic “low amplitude” is 0.2192, the value corresponding to “medium amplitude” is 0.7665, and the value corresponding to “high amplitude” is 1.3117. When the semantic description of the language variable is given, the predicted value output by the model can be classified into corresponding semantics according to the membership degree.

**DISCUSSION**

Figures 4–7 shows the test set of these four data sets, the prediction curve, and the prediction interval obtained by using the dynamic weight method. Figures 4–6 shows the prediction of all test sets of the corresponding data sets. Due to a large amount of wind speed data, part of the data are intercepted in Fig. 7 for display. The black curve represents the actual value, the red curve represents the predicted value, and the cyan area represents the predicted interval. The gray area, white and yellow band area in the figure corresponds to the semantics of “low amplitude”, “medium amplitude” and “high amplitude” respectively. Observing these figures, it can be seen that the proposed model can not only obtain the accurate value of the predicted value, but also obtain the interval value of the predicted value, and can describe the predicted value semantically. Thus, the data are described from multiple dimensions, which is more convenient for people to understand.

**Table 2** The semantics of FCMs nodes.

| Data      | Low      | Medium   | High     |
|-----------|----------|----------|----------|
| MG        | 0.2192   | 0.7665   | 1.3137   |
| Exchange rate | 0.5980   | 1.0430   | 1.4880   |
| Vatnsdalsa | 3.6700   | 28.8350  | 54.0000  |
| Wind speed | 0.0900   | 9.9400   | 19.7900  |

**Figure 4** The prediction results of MG time series. DOI: 10.7717/peerj-cs.726/fig-4
In order to quantitatively illustrate the performance of the multi-modal prediction model, the experimental results of each data set under the corresponding evaluation index are given in Table 3. By observing the data in the table, it can be found that the method...
using average weight and model weight has a higher PICP value, which indicates that the probability of prediction interval containing an actual value is higher. However, the PINAW index of the average weight method and the model weight method is higher, that is, the interval distance is relatively wide, which means that the accuracy of the interval is sacrificed to improve the coverage. Although the PICP index of the dynamic weight method is slightly lower, the PINAW index is smaller, which indicates that the prediction interval generated by this method is narrower and the accuracy is higher. The value of the comprehensive index CWC can better reflect the advantages and disadvantages of each method. It can be seen from the table that the dynamic weight method in all the data set experiments has a smaller CWC value. This shows that it is not appropriate to give all sub-modes the same weight or weight does not change. Further, observe the numerical prediction accuracy index RMSE generated by each weighting method in the table, similar conclusions can also be found. The RMSE value generated under the dynamic weight is the smallest, which indicates that the numerical accuracy of the prediction is higher.

To further analyze the performance of the proposed model, it is compared with the method of training FCMs by using all data. Figure 8 shows the RMSE curve of training FCMs with the whole data, in which the number of traversal nodes ranges from 3 to 20. As can be seen from Fig. 8, the predicted RMSE decreases first with the increase of nodes, but when the number of nodes exceeds a certain value, the RMSE of the predicted value will increase instead. It shows that increasing the number of nodes of FCMs cannot improve the prediction accuracy, and the excessive number of nodes increases the complexity of the model. From Fig. 8, it can be seen that the optimal number of nodes for the four data are 8, 7, 5, and 10 when modeling FCMs with the overall data.

Table 4 shows the RSME comparison results of the proposed method, the single method and LSTM(Long Short-Term Memory)method for time series data prediction accuracy. Among them, the number of nodes selected by the overall modeling method is the best, the
LSTM selected has 200 hidden units, and the maximum number of iterations is 250. It can be seen from the table that the multi-modal method has achieved high prediction accuracy. Table 5 shows the operation time of each algorithm on different data. The overall data training model takes the least time. The granulation process of the multimodal model was time-consuming, it was necessary to traverse and calculate the sequence data predicted by all sub-models in order to obtain the interval that maximized the 

\[ \text{Eq. (18)} \]

Its time complexity is \( \frac{n(n - 1)}{2} \), which means that the time complexity of the algorithm is \( O(n^2) \). LSTM needs to train a large model, so the time consumption of it is the most.

Figure 8 Plots of RMSE vs varying number of nodes with overall data. (A) MG, (B) exchange rate, (C) vatnsdalsa, (D) wind speed.

| Data        | Multi-modal RMSE | Overall model RMSE | LSTM RMSE |
|-------------|-------------------|--------------------|-----------|
| MG          | 0.0265            | 0.0424             | 0.0348    |
| Exchange rate | 0.0168            | 0.0344             | 0.0225    |
| Vatnsdalsa  | 0.7600            | 2.4006             | 1.2041    |
| Wind speed  | 0.7626            | 0.8028             | 0.8416    |

Table 5 The execution time of algorithms (s).

| Data        | Proposed | Single model | LSTM     |
|-------------|----------|--------------|----------|
| MG          | 51.95    | 1.47         | 179.41   |
| Exchange rate | 46.04    | 1.16         | 57.44    |
| Vatnsdalsa  | 51.26    | 0.92         | 157.23   |
| Wind speed  | 335.05   | 4.06         | 4,137.44 |

Yu et al. (2021), *PeerJ Comput. Sci.*, DOI 10.7717/peerj-cs.726
CONCLUSIONS

Time series often have a variety of change characteristics, that is, the change of the series is multimodal. If each change mode of the time series is modeled separately and the sub-models are fused effectively, the change rules of the time series can be reflected more effectively. A time series processing framework based on a fuzzy cognitive map is proposed in this paper. In this method, a large number of random time segments are selected to extract the possible modal variation characteristics of time series. Fuzzifying the time series and giving the time series semantic description is more conducive to people's cognition and decision-making. The output of each sub model is granulated and integrated, in the process of granulation, the predicted interval of time series can be obtained by maximizing the weight of the sub-model and minimizing the granulation interval. A variety of weight selection methods are studied for the effective fusion of sub-models. The experimental results show that the dynamic weight method can obtain a better prediction interval index and higher prediction accuracy.

In the proposed method, the bootstrap method is used to select a large number of time segments in order to ensure that all the changing modes of time series can be selected. However, this method will inevitably produce redundant time segments, which will reduce the efficiency of the model. Therefore, it is necessary to study more effective methods of time series change modes. For the weights of the sub-models, three basic weight methods are verified in this paper, and the relevant weight selection method can be further studied to improve the performance of the model. In addition, modeling complex system using distributed FCMs is a feasible scheme. A complex system can be decomposed into multiple subsystems. Each subsystem is used to construct one corresponding FCMs model. Then construct a global FCMs model using these sub FCMs models.

ADDITIONAL INFORMATION AND DECLARATIONS

Funding
This work was supported by the Research Initiation Project of Northeast Electric Power University (No.11847). The funders had no role in study design, data collection and analysis, decision to publish, or preparation of the manuscript.

Grant Disclosures
The following grant information was disclosed by the authors: Research Initiation Project of Northeast Electric Power University: 11847.

Competing Interests
The authors declare that they have no competing interests.

Author Contributions
- Tianming Yu performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, and approved the final draft.
Qunfeng Gan performed the experiments, performed the computation work, prepared figures and/or tables, and approved the final draft.

Guoliang Feng conceived and designed the experiments, performed the experiments, analyzed the data, performed the computation work, prepared figures and/or tables, authored or reviewed drafts of the paper, and approved the final draft.

Data Availability
The following information was supplied regarding data availability:
Raw data and code are available in the Supplemental Files.

Supplemental Information
Supplemental information for this article can be found online at http://dx.doi.org/10.7717/peerj-cs.726#supplemental-information.

REFERENCES
Bargiela A, Pedrycz W. 2008. Toward a theory of granular computing for human-centered information processing. IEEE Transactions on Fuzzy Systems 16(2):320–330 DOI 10.1109/TFUZZ.2007.905912.

Bose M, Mali K. 2019. Designing fuzzy time series forecasting models: a survey. International Journal of Approximate Reasoning 111(3):78–99 DOI 10.1016/j.ijar.2019.05.002.

Boyd S, Vandenberghe L. 2004. Convex optimization. Cambridge: Cambridge University Press.

Cao LJ, Tay FEH. 2003. Support vector machine with adaptive parameters in financial time series forecasting. IEEE Transactions on Neural Networks 14(6):1506–1518 DOI 10.1109/TNN.2003.820556.

Chaâbane N. 2014. A novel auto-regressive fractionally integrated moving average-least-squares support vector machine model for electricity spot prices prediction. Journal of Applied Statistics 41(3):635–651 DOI 10.1080/02664763.2013.847068.

Felix G, Nápoles G, Falcon R, Froelich W, Vanhoof K, Bello R. 2019. A review on methods and software for fuzzy cognitive maps. Artificial Intelligence Review 52(3):1707–1737 DOI 10.1007/s10462-017-9575-1.

Feng G, Lu W, Pedrycz W, Yang J, Liu X. 2021. The learning of fuzzy cognitive maps with noisy data: a rapid and robust learning method with maximum entropy. IEEE Transactions on Cybernetics 51(4):2080–2092 DOI 10.1109/TCYB.2019.2933438.

Feng G, Zhang L, Yang J, Lu W. 2021. Long-term prediction of time series using fuzzy cognitive maps. Engineering Applications of Artificial Intelligence 102:104274 DOI 10.1016/j.engappai.2021.104274.

Froelich W, Pedrycz W. 2017. Fuzzy cognitive maps in the modeling of granular time series. Knowledge-Based Systems 115(5):110–122 DOI 10.1016/j.knosys.2016.10.017.

Guo H, Liu X, Sun Z. 2016. Multivariate time series prediction using a hybridization of VARMA models and Bayesian networks. Journal of Applied Statistics 43(16):2897–2909 DOI 10.1080/02664763.2016.1155111.

Khashei M, Bijari M, Ardali GAR. 2012. Hybridization of autoregressive integrated moving average (ARIMA) with probabilistic neural networks (PNNs). Computers & Industrial Engineering 63(1):37–45 DOI 10.1016/j.cie.2012.01.017.
Khosravi A, Nahavandi S, Creighton D. 2013. Prediction intervals for short-term wind farm power generation forecasts. IEEE Transactions on Sustainable Energy 4(3):602–610 DOI 10.1109/TSTE.2012.2232944.

Lee YS, Tong LI. 2011. Forecasting time series using a methodology based on autoregressive integrated moving average and genetic programming. Knowledge-Based Systems 24(1):66–72 DOI 10.1016/j.knosys.2010.07.006.

Lu W, Chen X, Pedrycz W. 2015. Using interval information granules to improve forecasting in fuzzy time series. International Journal of Approximate Reasoning 57(1–2):1–18 DOI 10.1016/j.ijar.2014.11.002.

Lu W, Yang J, Liu X. 2014. The modeling and prediction of time series based on synergy of high-order fuzzy cognitive map and fuzzy c-means clustering. Knowledge-Based Systems 70:242–255 DOI 10.1016/j.knosys.2014.07.004.

Ma YY, Zheng JC, Chen L, Dong K, Xiao XM. 2018. Neuroblastomas in Eastern China: a retrospective series study of 275 cases in a regional center. PeerJ 6(9):e5665 DOI 10.7717/peerj.5665.

Nishii T, Kono A, Tani W, Suehiro E, Negi N, Takahashi S, Sugimura K. 2016. Four-dimensional noise reduction using the time series of medical computed tomography datasets with short interval times: a static-phantom study. PeerJ 4(2):e1680 DOI 10.7717/peerj.1680.

Papageorgiou EL, Salmeron JL. 2012. A review of fuzzy cognitive maps research during the last decade. IEEE Transactions on Fuzzy Systems 21(1):66–79 DOI 10.1109/TFUZZ.2012.2201727.

Pedrycz W. 2011. The principle of justifiable granularity and an optimization of information granularity allocation as fundamentals of granular computing. Journal of Information Processing Systems 7(3):397–412 DOI 10.3745/JIPS.2011.7.3.397.

Pedrycz W. 2018. Granular computing for data analytics: a manifesto of human-centric computing. IEEE/CAA Journal of Automatica Sinica 5(6):1025–1034 DOI 10.1109/JAS.2018.7511213.

Pedrycz W, Homenda W. 2013. Building the fundamentals of granular computing: a principle of justifiable granularity. Applied Soft Computing 13(10):4209–4218 DOI 10.1016/j.asoc.2013.06.017.

Pedrycz W, Jastrzebska A, Homenda W. 2016. Design of fuzzy cognitive maps for modeling time series. IEEE Transactions on Fuzzy Systems 24(1):120–130 DOI 10.1109/TFUZZ.2015.2428717.

Pedrycz W, Vukovich G. 2001. Abstraction and specialization of information granules. IEEE Transactions on Systems, Man and Cybernetics - PartB: Cybernetics 31(1):106–111 DOI 10.1109/3477.907568.

Quan H, Srinivasan D, Khosravi A. 2013. Short-term load and wind power forecasting using neural network-based prediction intervals. IEEE Transactions on Neural Networks and Learning Systems 25(2):303–315 DOI 10.1109/TNNLS.2013.2276053.

Ren M, Zhang Q, Zhang J. 2019. An introductory survey of probability density function control. Systems Science & Control Engineering 7(1):158–170 DOI 10.1080/21642583.2019.1588804.

Salmeron JL, Froelich W. 2016. Dynamic optimization of fuzzy cognitive maps for time series forecasting. Knowledge-Based Systems 105(6):29–37 DOI 10.1016/j.knosys.2016.04.023.

Singh P. 2017. A brief review of modeling approaches based on fuzzy time series. International Journal of Machine Learning and Cybernetics 8(2):397–420 DOI 10.1007/s13042-015-0332-y.

Song Q, Chissom BS. 1993. Fuzzy time series and its models. Fuzzy Sets and Systems 54(3):269–277 DOI 10.1016/0165-0114(93)90372-O.
Stach W, Kurgan LA, Pedrycz W. 2008. Numerical and linguistic prediction of time series with the use of fuzzy cognitive maps. *IEEE Transactions on Fuzzy Systems* **16**(1):61–72 DOI 10.1109/TFUZZ.2007.902020.

Weigend AS, Gerschenfeld NA. 1994. Time series prediction: forecasting the future and understanding the past. *International Journal of Forecasting* **10**(3):463–466.

Yin X, Zhang Q, Wang H, Ding Z. 2019. Rbfnn-based minimum entropy filtering for a class of stochastic nonlinear systems. *IEEE Transactions on Automatic Control* **65**(1):376–381 DOI 10.1109/TAC.2019.2914257.

Zadeh LA. 1997. Toward a theory of fuzzy information granulation and its centrality in human reasoning and fuzzy logic. *Fuzzy Sets and Systems* **90**(2):111–127 DOI 10.1016/S0165-0114(97)00077-8.

Zadeh LA. 2008. Is there a need for fuzzy logic? *Information Sciences* **178**(13):2751–2779 DOI 10.1016/j.ins.2008.02.012.

Zeng YF, Khondker A. 2016. A time-series analysis of relevant pollutants in Hamilton (Ontario) and induced mortality. *PeerJ* 4:e1904v1 DOI 10.7287/peerj.preprints.1904v1.

Zhang GP. 2003. Time series forecasting using a hybrid ARIMA and neural network model. *Neurocomputing* **50**:159–175 DOI 10.1016/S0925-2312(01)00702-0.

Zhang Q, Wang H. 2021. A novel data-based stochastic distribution control for non-Gaussian stochastic systems. Epub ahead of print 9 March 2021. *IEEE Transactions on Automatic Control* DOI 10.1109/TAC.2021.3064991.

Zheng YF, Li XB, Huang YP, Jia LF, Li WR. 2018. Time series clustering of mRNA and lncRNA expression during osteogenic differentiation of periodontal ligament stem cells. *PeerJ* 6(4):e5214 DOI 10.7717/peerj.5214.