A time series forecasting based on cloud model similarity measurement

Yan, Gaowei; Jia, Songda; Ding, Jie; Xu, Xinying; Pang, Yusong

DOI
10.1007/s00500-018-3190-1

Publication date
2018

Document Version
Accepted author manuscript

Published in
Soft Computing

Citation (APA)
Yan, G., Jia, S., Ding, J., Xu, X., & Pang, Y. (2018). A time series forecasting based on cloud model similarity measurement. *Soft Computing*, *23 (2019)*(14), 5443-5454. https://doi.org/10.1007/s00500-018-3190-1

Noname manuscript No.
(will be inserted by the editor)

A time series forecasting based on cloud model similarity measurement

Gaowei Yan[1] · Songda Jia[1] · Jie Ding[1] · Xinying Xu[1] · Yusong Pang[1,2]

Received: date / Accepted: date

Abstract

In this paper, a local cloud model similarity measurement (CMSM) is proposed as a novel method to measure the similarity of time series. Time series similarity measurement is an indispensable part for improving the efficiency and accuracy of prediction. The randomness and uncertainty of series data are critical problems in the processing of similarity measurement. CMSM obtains the internal information of time series from the general perspective and local trend using the cloud model, which reduces the uncertainty of measurement. The neighbor set is selected from time series by CMSM and used to construct a prediction model based on least squares support vector machine. The proposed technique reduces the potential for overfitting and uncertainty and improves model prediction quality and generalization. Experiments were performed with four datasets selected from Time Series Data Library. The experimental results show the feasibility and effectiveness of the proposed method.

Keywords Time series forecasting · Similarity measurement · Cloud model · Uncertainty · Least squares support vector machine

✉ Yusong Pang
y.pang@tudelft.nl

Gaowei Yan
yangaowei@tyut.edu.cn

Songda Jia
jiasongda0226@link.tyut.edu.cn

Jie Ding
dingjie2015@foxmail.com

Xinying Xu
xuxinying@tyut.edu.cn

[1] Department of Automation, Taiyuan University of Technology, Taiyuan 030024, China

[2] Section Transport Engineering and Logistics, Delft University of Technology, 2628 CD Delft, The Netherlands

1 Introduction

Time series forecasting model has been widely applied to the abundant decision making in the fields of natural science, engineering, and economic management (De Gooijer and Hyndman 2006; Coyle et al. 2005; Jayawardena et al. 2002). It is used to predict future values by capturing the inherent structural characteristics of data based on previously collected historical data. Two prediction strategies are usu-ally employed in time series prediction model. One is the single-step prediction, which reflects short-term tendency of subjects and equates the immediate prediction. Another is multi-step prediction, which shows long-term change over time (Xiong et al. 2013; Bao et al. 2014). These two strate-gies play disparate roles in different circumstances.

The models for time series prediction can be primarily divided into statistical models and artificial intelligence models. In statistical models, the methods of moving average, exponential smoothing, and autoregressive integrated moving average model (ARIMA), etc., are employed separately or combined (Ruta et al. 2011). However, these approaches are limited when dealing with nonlinear data, resulting in unstable model and inaccurate prediction result (Bhardwaj et al. 2013). The artificial intelligence models show strong nonlinear mapping ability and therefore, predict time series more accurately. Artificial neural network (ANN), extreme learning machine (ELM), adaptive neural fuzzy inference system (ANFIS), recurrent neural network (RNN), support vector machine (SVM), and least squares support vector machine (LSSVM) are the key models in the artificial intelligence models. ANN has been extensively applied to many areas of time series prediction because of its ability in capturing the nonlinear flexible and strong learning ability (Coyle et al. 2005; Khashei and Bijari 2010; Wu and Lee 2015). Van Heeswijk et al. (2009) applied the ELM to time series prediction using adaptive ensemble ELM models, which shows stability and adaptability during series prediction. Sisman-Yilmaz et al. (2004) employed improved ANFIS in multivariable time series prediction and achieved good prediction accuracy. In Giles et al. (2001), RNN was applied to the prediction of time series. Thanks to the loopback structure and information feedback process, RNN addresses difficulties with non-stationarity, overfitting, and unequal a priori class probabilities. SVM (Saimurugan et al. 2011) is a different category in machine learning techniques, which uses risk minimization principle and owns better generalization ability. LSSVM introduces loss function and kernel function using SVM as fundamental. In Suykens et al. (2002), LSSVM was used in time series regression and prediction. The high training efficiency and strong generalization ability make LSSVM achieve good prediction precision in nonlinear time series forecasting.

Artificial intelligence models are required to be trained using the time series data. Selecting the appropriate training set from the time series data is of great importance in artificial intelligence models since it can increase the quality of the prediction. Global and local models are two kinds of training set selection methods (Chen and Lee 2015). In global model method, all the sequences from the training domain are used for model training. However, this will lead to heavy computation and the dissimilarity sequences in the domain and will disturb the training process, resulting in low prediction accuracy. Therefore, for large-scale and dissimilar sequences, local model method is more popular. In this method, sequences in the training domain that are similar to the predictive sequence are selected and used as the training set for model training. The set that contains all the similar sequences is called the nearest neighbor set (Chen and Lee 2015). McNames (1998) proposed a local model method for the nonlinear time series prediction. It used a weighted Euclidean metric method to measure similarity and employed the cross-validation error to assess model accuracy. Jayawardena et al. (2002) presented the generalized degrees of freedom, which was used to determine similar sequences. In addition to the memory function brought by the loopback structure, in Cherif and Bon (2014), RNN was applied to the local model, which improved the prediction performance of the model (Goel et al. 2017). Local neural fuzzy time series forecasting model based on LSSVM was proposed in Miranian and Abdollahzade (2013). It used hierarchical binary tree learning algorithm to quickly and effectively select similar sequences and obtained a better result in chaotic time series forecasting. Chen and Lee (2015) employed hybrid Euclidean distance method to measure the similarity of the predictive sequence and all sequences for similar sequences selection.

Time series data show uncertainty. In other words, the data at some particular time series are not accurate or even missed. The uncertainty is caused by the restriction of acquisition accuracy of instruments or the detection technology, noise and bandwidth limit in the transmission process, the generalization treatment for data security, and data missing, etc. The data uncertainty is less considered in the conventional similar sequences selection methods in the previous models. Thus, the sequence data for model training are not well selected, and the final prediction accuracy is affected indirectly. The cloud model (Li and Du 2007) combines the fuzziness and randomness of the qualitative concepts in natural language. It can realize the mapping from the quantitative value to the qualitative linguistic value, which reduces the impact of the outliers and missing values. To solve the uncertainty and improve the effectiveness of prediction, the cloud model can be combined with LSSVM for similarity measurement and prediction.

In this paper, a time series forecasting method based on cloud model similarity measurement is proposed, which evaluates the similarity of time series from the overall level and local trend and predicts it in combination with LSSVM.

The main contributions of this paper are as follows:

1. Transform the identified time series into digital feature sequences, reducing the randomness and uncertainty of the sequence data.
2. Measure the similarity of the time series from both the overall level and the local trend to improve the accuracy of the measurement.
3. A local model based on LSSVM is constructed so that the prediction accuracy has been improved.

The remainder of the paper is organized as follows. In Sect. 2, we briefly describe the related work about measuring similarity of series. Section 3 presents the similarity measurement based on the cloud model. Section 4 introduces the process of proposed method. We show the empirical results of the comparisons between several existing methods in Sect. 5. Finally, the conclusions are drawn in Sect. 6.

2 Related work

Similarity measurement plays a key role in the local model for the selection of nearest neighbor set. Many researchers use Euclidean distance or its deformation to measure the similarity of time series (Wang et al. 2013; Keogh et al. 2001; Chiu et al. 2003). However, Euclidean distance has inherent defects, such as a poor noise robustness and a lack of deformable identification on timeline (Chiu et al.

2003). The dynamic time warping (DTW) (Adwan and Arof 2012) can effectively eliminate defects of the Euclidean distance. Unfortunately, the computationally demanding of this method restricts its applications. Bernecker et al. (2011) constructed the angle sequence using the adjacent segment angle to approximate time series. In this method, Euclidean distance is transformed to the angle distance, which is used to measure the similarity. Symbolic aggregate approximation method was proposed by Korn and Muthukrishnan (2000). This algorithm is widely used due to its simple structure and its independence from specific experimental data. The piecewise linear radian representation method was proposed in Lin et al. (2003). It defined the radian distance, and similarity is measured based on the distance. The experimental results showed the accuracy and stability of this method in multiresolution. Popivanov and Miller (2002) used the wavelet transform for similarity search in time series. To improve the prediction accuracy, Wu and Lee (2015) defined a hybrid distance, which measured the degree of two shape contours between two time series. In this method, the trend of the sequences is taken into account. However, all these methods above do not consider data uncertainty problem.

In order to solve the uncertainty problem in time series data, a time series prediction method based on cloud model was proposed for the first time (Jiang et al. 2000). In this method, cloud model is employed to represent knowledge, and the predictive knowledge is divided into quasiperiodic regularity and current tendency. The final prediction results are obtained by two different granularities of prediction knowledge. In Zhang et al. (2004), a similarity measurement based on cloud droplets distance was proposed. According to the cloud droplets distribution, the cloud similarity algorithm based on interval was presented in Cai et al. (2011). Zhang et al. (2007) considered digital characteristics of cloud model as vectors and proposed likeness comparing method based on cloud model, which is successfully used in collaborative filtering recommendation system. In Li et al. (2011), two kinds of normal cloud model, which were based on the expectation curve and maximum boundary curve, respectively, were proposed for the similarity calculation. The maximum and minimum close degree based on cloud model was proposed in Lu and Qin (2014).

Although traditional cloud models can handle uncertainty, these similarity measurement methods are complicated and computationally demanding. To simplify this, Sun et al. (1964) proposed an Overlap based Expectation curve of Cloud Model (OECM) algorithm. In this paper, the OECM algorithm is employed to measure similarity for selecting nearest neighbor set and is combined with LSSVM model for forecasting.

3 Similarity measurement based on cloud model

3.1 Cloud model

Cloud model can describe the fuzziness and randomness of concepts and implement the transformation between a qualitative concept and its quantitative linguistic value. The brief description of cloud model is shown as follows.

Definition 1 (Li and Du 2007) Suppose that U is a quantitative numerical universe of discourse and C is a qualitative concept in U. If value $x \in U$ is a random implementation of concept C, and the membership degree $\mu_C(x) \in [0, 1]$ of x belonging to C is a random variable with tendency:

$$\mu : U \to [0, 1],\ \forall x \in U, x \to \mu(x). \tag{1}$$

The distribution of x in universe of discourse U is defined as a cloud, and each x is called a cloud drop.

Three numerical characteristics Ex, En, and He are employed to reflect the property of concept in the normal cloud model. Expectation Ex is the mathematical expectation of the points belonging to a qualitative concept in the universe. Entropy En represents the uncertainty measurement of a certain concept. Hyper entropy He is the uncertain degree of entropy (Li 2000).

Definition 2 (Li et al. 2011) If $x \sim N(Ex, En'^2)$, where $En'^2 \sim N(En, He^2)$, the membership degree of the qualitative concept C satisfies:

$$\mu_C(x) = e^{\frac{-(x-Ex)^2}{2En'^2}}. \tag{2}$$

The distribution of x in universe of discourse U is called as a normal cloud.

As similar as the normal distribution, the droplets that have significant contributions for the qualitative concept are mostly in the range: $[Ex - 3En, Ex + 3En]$. The cloud droplets that are located outside of $[Ex - 3En, Ex + 3En]$ are called the small probability event. It does not affect the overall characteristics of the cloud model if we do not take them into account. This is the "$3En$"rules of normal cloud (Li and Du 2007).

The transformation between a qualitative concept and its quantitative instantiations can be realized by the forward cloud generator (FCG) and the backward cloud generator (BCG) (Li and Du 2007). In this paper, BCG effectively transforms the accurate data of certain number into the qualitative concept by digital characteristics Ex, En, and He. A new backward cloud algorithm of the cloud X information was proposed (Liu et al. 2004). The expression is as follows:

Step 1 For the sample points: $x_i (i = 1, 2, \ldots, n)$, get its sample average $\bar{X} = \frac{1}{n}\sum_{i=1}^{n} x_i$, a first-order sample absolute

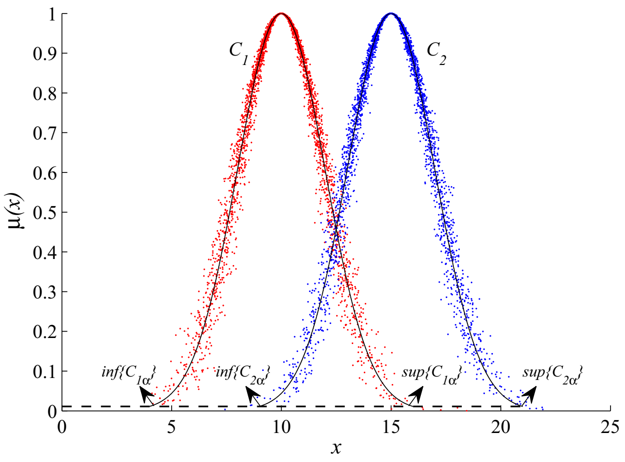

Fig. 1 Supremum and infimum of clouds C_1 and C_2

central moment $\frac{1}{n}\sum_{i=1}^{n}|x_i - \bar{X}|$, and sample variance $S^2 = \frac{1}{n-1}\sum_{i=1}^{n}|x_i - \bar{X}|^2$.

Step 2 Obtain the expectation: $Ex = \bar{X}$.

Step 3 Calculate the entropy:

$$En = \sqrt{\pi/2} \times \frac{1}{n}\sum_{i=1}^{n}|x_i - Ex|.$$

Step 4 Calculate the hyper entropy:

$$He = \sqrt{S^2 - En^2}.$$

3.2 Overlap-based expectation curve of cloud model

The OECM algorithm as a measurement method is used to measure similarity degree of cloud models. In this algorithm, overlap degree is proposed to describe the overlapping part of two clouds and converted to similarity degree. The algorithm can be described as follows.

Definition 3 Assume that there are two clouds C_1 and C_2 in universe of discourse U (Sun et al. 1964), which are shown in Fig. 1. The overlap degree is defined as:

$$ol(C_1, C_2) = \frac{2(\sup\{C_{1\alpha}\} - \inf\{C_{2\alpha}\})}{(\sup\{C_{1\alpha}\} - \inf\{C_{1\alpha}\}) + (\sup\{C_{2\alpha}\} - \inf\{C_{2\alpha}\})}, \tag{3}$$

where $\sup\{C_{i\alpha}\}$ and $\inf\{C_{i\alpha}\}(i = 1, 2)$ are supremum and infimum of cloud C_i expectation curve, respectively. The similarity degree of C_1 and C_2 is defined as:

$$Sim(C_1, C_2) = \frac{\mu - \alpha}{1 - \alpha} \cdot ol, \tag{4}$$

where μ represents the certainty degree of the intersection of the C_1 and C_2, and α is the certainty degree of cloud model "$3En$"rules.

3.3 Information fusion similarity measurement

In the previous similarity measurement, the variation tendency of sequence is not taken into consideration, resulting in the incompleteness of information in the process of measurement. In this paper, information fusion similarity measurement based on cloud model is proposed. The definition can be described as follows.

Definition 4 Assuming that there are series $X = \{x_1, x_2, \ldots, x_n\}$ and $Y = \{y_1, y_2, \ldots, y_n\}$, the trend sequences of X and Y are $T_X = \{x_2 - x_1, x_3 - x_2, \ldots, x_n - x_{n-1}\}$ and $T_Y = \{y_2 - y_1, y_3 - y_2, \ldots, y_n - y_{n-1}\}$, which can be qualitatively represented as shown in Fig. 2. The degree of information fusion similarity between X and Y, $Sim_{IF}(X, Y)$, is defined as:

$$Sim_{IF}(X, Y) = \text{avg}\{Sim_g(X, Y), Sim_l(T_X, T_Y)\}, \tag{5}$$

where avg means the average and $Sim_g(X, Y)$ is the degree of global similarity between X and Y, which reflects the global similarity level of time series. The computational formula is as follows:

$$Sim_g(X, Y) = \min\{Sim(X_1, Y_1), Sim(X_2, Y_2), \ldots, Sim(X_m, Y_m)\}, \tag{6}$$

where min means the minimum and $Sim(X_i, Y_i), i = 1, 2, \ldots, m$ represents the degree of basic similarity between the ith segment of X and Y. Similarly, $Sim_l(T_X, T_Y)$ is the degree of local trend similarity, which reflects the variation trend. The computational formula is as follows:

$$Sim_l(T_X, T_Y) = min\{Sim(T_{X_1}, T_{Y_1}), Sim(T_{X_2}, T_{Y_2}), \ldots, Sim(T_{X_m}, T_{Y_m})\}, \tag{7}$$

where $Sim(T_{X_i}, T_{Y_i}), i = 1, 2, \ldots, m$ represents the degree of basic similarity between the ith segment of T_X and T_Y.

The steps of employing the cloud model method to calculate the degree of information fusion similarity between X and Y are as follows:

Step 1 Calculate the trend sequence T_X, T_Y of X and Y, respectively.

Step 2 Divide the sequence of X, Y, T_X, and T_Y into m segments.

Step 3 Calculate the digital characteristics of all segments through BCG algorithm.

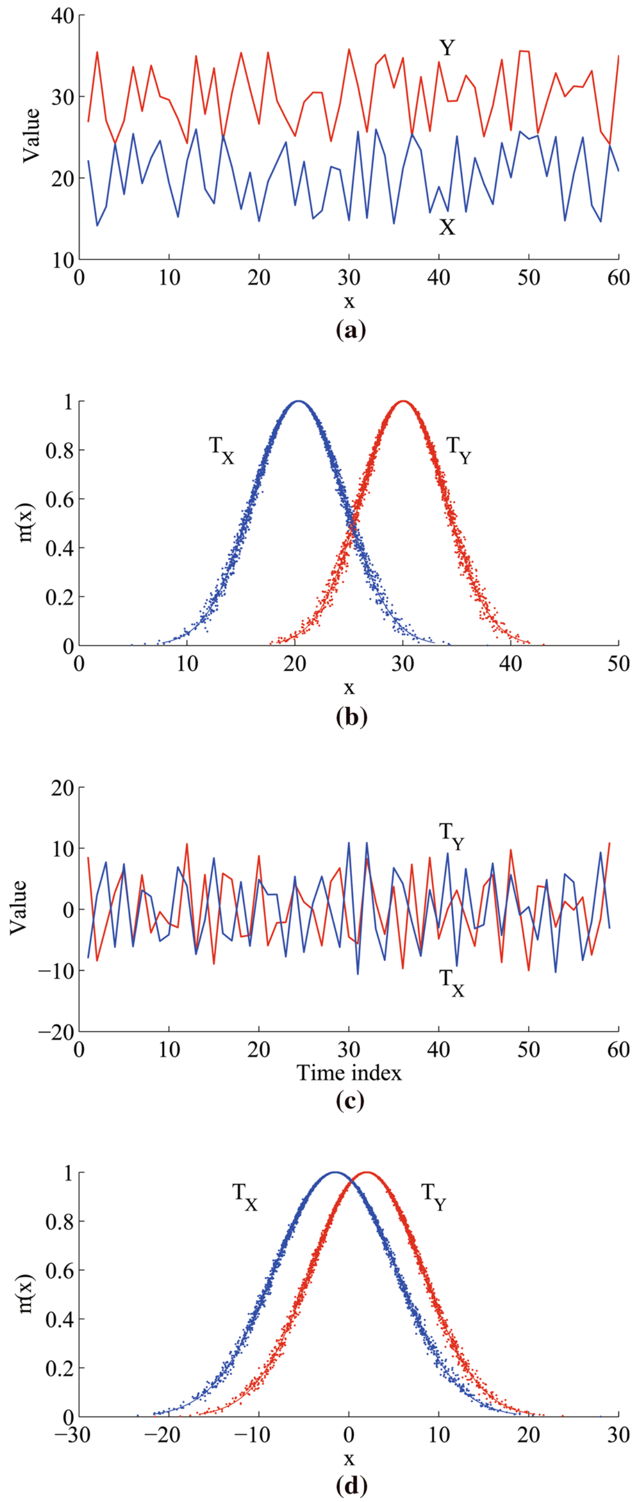

Fig. 2 Representation of cloud model for time series. **a** Original time series. **b** The qualitative representation of X and Y. **c** Trend time series. **d** The qualitative representation of T_X and T_Y

Step 4 Calculate the $Sim_g(X, Y)$ and $Sim_l(T_X, T_Y)$ using OECM algorithm.

Step 5 To obtain the degree of information fusion similarity according to Eq. (5).

Table 1 Description of training set

Output	Input
$\hat{y}_{r+h}$	$y_r, y_{r-1}, \ldots, y_2, y_1$
$\hat{y}_{r+h+1}$	$y_{r+1}, y_r, \ldots, y_3, y_2$
$\ldots$	$\ldots$
$\hat{y}_{t-1}$	$y_{t-h-1}, y_{t-h-2}, \ldots, y_{t-h-r+1}, y_{t-h-r}$
$\hat{y}_t$	$y_{t-h}, y_{t-h-1}, \ldots, y_{t-h-r+2}, y_{t-h-r+1}$

4 Prediction model based on similarity measurement

4.1 Sample construction

Assume that the time series $\{y_1, y_2, \ldots, y_{t-1}, y_t\}$ are collected at equally spaced time, where t represents the current moment. The objective of time series forecasting is to estimate the value of future time $t + h$, $\hat{y}_{t+h}$ as formula (8):

$$\hat{y}_{t+h} = f(y_t, y_{t-1}, \ldots, y_{t-r+1}), \tag{8}$$

where f is the prediction model, h represents the horizon of prediction, and r is the number of time lags. If $h = 1$, we call it as the single-step prediction, and if $h > 1$, we call it as the multi-step prediction (Chen and Lee 2015).

The term current vector is called query sequence q:

$$q = \{y_t, y_{t-1}, \ldots, y_{t-r+2}, y_{t-r+1}\}. \tag{9}$$

In order to obtain $\hat{y}_{t+h}$, the corresponding training set of query sequence is constructed, where the proper lags r are determined. The training set is presented in Table 1.

4.2 Neighbor selection

In neighbor selection, our goal is to select the neighbor set of query sequences through measuring and ranking the similarity degrees between query sequence and each of training set.

Following the information fusion similarity measurement, we can proceed to calculate the similarity degrees of q and each of training set. The similarity degrees are sorted in descending order, and the first k sequences are selected as the neighbor set of q.

4.3 Prediction model

In this subsection, a brief description of LSSVM (Suykens et al. 2002) is given, which is used to predict the future values. As the modification of standard SVM formulation, it is usually available to solve the function estimation and nonlinear regression problems.

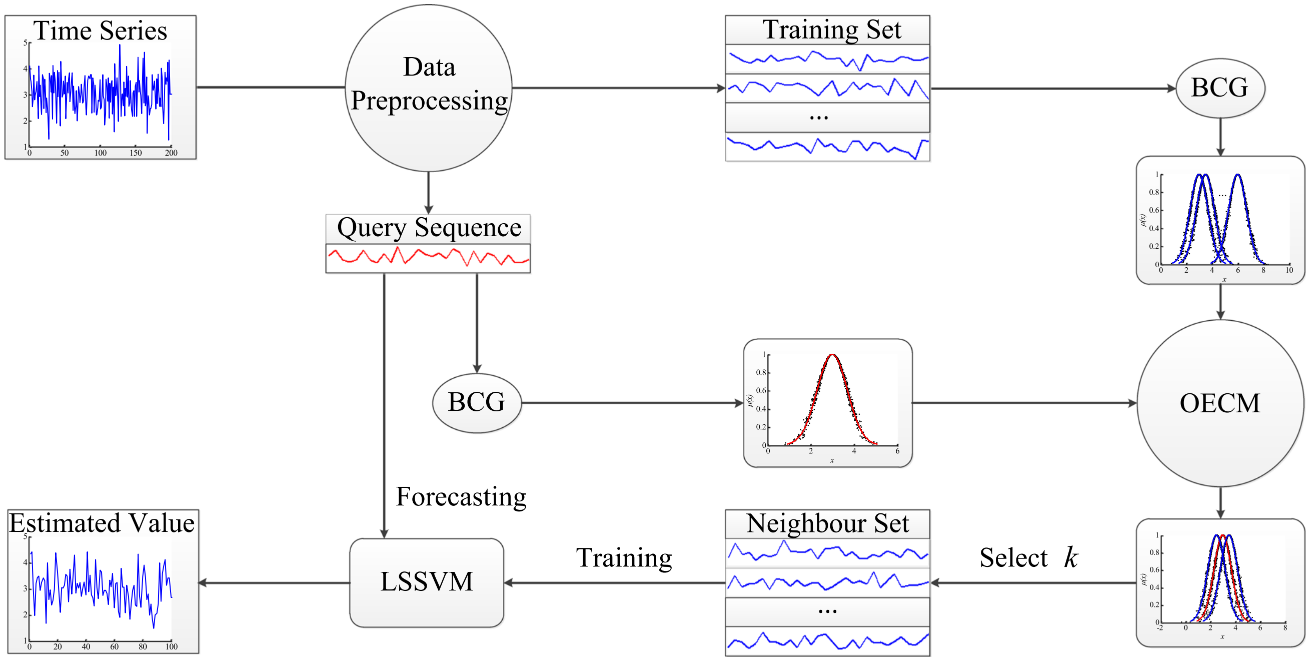

Fig. 3 Flowchart of proposed time series prediction method

A linear function in primal weight space can be expressed as:

$$y(x) = \omega^T \phi(x) + b, \tag{10}$$

where $x, y \in R$. $\phi(\cdot)$ is the nonlinear mapping function from the input space to a high-dimensional feature. ω is the weight vector, and b is a bias term.

Considering a training set $(x_i, y_i), i = 1, 2, \ldots, n$, the objective function of LSSVM in the primal weight space can be described as follows:

$$\min J(\omega, e) = \frac{1}{2}\omega^T \omega + \frac{1}{2}\gamma \sum_{i=1}^{n} e_i^2, \tag{11}$$

which is subject to the constraints:

$$y_i = \omega^T \phi(x_i) + b + e_i, i = 1, \ldots, n, \tag{12}$$

where γ is the regularization factor and e_i represents regression error of the ith sample. The Lagrangian function for Eqs. (11) and (12) is as follows:

$$L = J - \sum_{i=1}^{n} \alpha_i \{\omega^T \phi(x_i) + b + e_i - y_i\}, \tag{13}$$

where α_i is the Lagrange multipliers. Then, this optimization problem can be transformed into solving linear equations:

$$\begin{bmatrix} 0 & {l_n}^T \\ l_n & K + \gamma^{-1} l_n \end{bmatrix} \begin{bmatrix} b \\ a \end{bmatrix} = \begin{bmatrix} 0 \\ y \end{bmatrix}, \tag{14}$$

where $l_n = [1, 1, \ldots, 1]^T$, $y = [y_1, y_2, \ldots, y_n]^T$, $a = [a_1, a_2, \ldots, a_n]^T$, and K is a kernel matrix defined as $K_{ij} = K(x_i, x_j)$. In order to avoid dimension disaster in the high-dimensional feature space, the radial basis function (RBF) is used as the kernel function, and it is given by

$$K(x, x_i) = e^{-\frac{\|x - x_i\|^2}{\sigma^2}}, \tag{15}$$

where σ is the width of the RBF. Finally, the estimating function of LSSVM is represented as:

$$y(x) = \sum_{i=1}^{n} \alpha_k K(x, x_i) + b. \tag{16}$$

The learning and generalization ability of LSSVM is influenced greatly by regularization parameter γ and kernel function width σ^2, which are usually determined by using grid search algorithm with K-fold cross-validation (Arlot and Celisse 2010).

In this paper, the LSSVM is trained by adopting the idea of local model (McNames 1998). The proposed prediction model is divided into three steps: sample construction, neighbor selection, and query prediction. The frame structure of prediction model is shown in Fig. 3. The specific algorithm flow is described in Algorithm 1.

Algorithm 1 The proposed prediction algorithm

```
Input: A time series $\mathbf{y} = \{y_1, y_2, \ldots, y_r, y_{r+1}, \ldots, y_t\}$.
Output: The estimated value $\hat{y}_{t+h}$.
Step 1. Preprocess the dataset $\mathbf{y}$ and obtain query sequence $q = \{y_t, y_{t-1}, \ldots, y_{t-r}, y_{t-r+1}\}$, training set
$s_1 = \{y_r, y_{r-1}, \ldots, y_2, y_1\}$,
$s_2 = \{y_{r+1}, y_r, \ldots, y_3, y_2\}$,
$\ldots$,
$s_p = \{y_{t-h}, y_{t-h-1}, \ldots, y_{t-h-r+2}, y_{t-h-r+1}\}$,
$(p = t - h - r + 1)$.
Step 2. Calculate $q'$ and $s'$ (the digital characteristics of $q$ and $s$).
for $i = 1, 2, \ldots, N_t$ do
    Step 3. Calculate the similarity degree $Sim^i_{IF}$ of $q'$ and $s'$.
end for
Step 4. Sort $Sim_{IF}$ in descending order.
Step 5. Select the first $k$ sequence from $s$ to form neighbor set.
Step 6. Train LSSVM with the neighbor set.
Step 7. Obtain $\hat{y}_{t+h}$ by putting $q$ into LSSVM.
```

In this algorithm, N_t represents the number of forecasting points. For each input q corresponding to the forecasting point, the neighbor set is selected from training set and used to train a LSSVM model.

5 Experimental results

The experimental results of four time series data are presented, aiming to evaluate the effectiveness and feasibility of the proposed method. All the time series data are obtained from the Time Series Data Library (TSDL) (Hyndman 2013), which is an open repository of time series datasets. These four time series data are also used in Wu and Lee (2015) and Veloz et al. (2016). The four time series data are: (a) the Laser dataset, which is from the fluctuations of a far-infrared laser and measured in a physics laboratory experiment; (b) Sunspots, which contain the annual amount of observed sunspots from 1700 to 1987; (c) ESTSP2007 dataset, which is from the European Symposium on Time Series Prediction competition 2007; (d) Poland electricity load, which presents the electricity load values of Poland from 1990s. Table 2 lists the total size, training size, test size, minimum, maximum, and mean of four time series datasets.

To evaluate the similarity between the k sequences of the nearest neighbor subset and the query sequence, the unitary evaluation index is employed.

Definition 5 Suppose that the nearest neighbor subset of the query sequence is selected from the training set, and the match degree is defined as:

$$\text{Match} = 1 - \frac{p}{k} \cdot \frac{\sum_{i=1}^{k} \text{var}(s_i - q)}{\sum_{i=1}^{p} \text{var}(s_i - q)}, \tag{17}$$

where var means the variance and k and p are the number of neighbor subset and training set, respectively. From the definition, the higher the match degree is, the higher the similarity degree is.

Experiments verify the effectiveness of the proposed method in different prediction horizons. In single-step prediction, the impact of different nearest neighbor set size k on the prediction accuracy is analyzed. Considering the influence of time complexity, the experiment compares the computational time of several measurement methods. In addition, the different prediction models combined with CMSM method are compared. In multi-step prediction, we obtained the prediction errors of four datasets in five steps, ten steps, and 15 steps, respectively. Different similarity measurement methods are contrasted in multi-step prediction as well.

Evaluating the accuracy and reliability of the prediction results is an important part of forecasting analysis. In this paper, four error indices are considered: root-mean-squared error (RMSE), mean absolute error (MAE), normalized root-mean-square error (NRMSE) (Wu and Lee 2015), and normalized mean square error (NMSE) (Veloz et al. 2016). These error indices are shown as follows:

$$\text{MAE} = \frac{1}{N_t} \sum_{i=1}^{N_t} |y_i - \hat{y}_i|, \tag{18}$$

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^{N_t} (y_i - \hat{y}_i)^2}{N_t}}, \tag{19}$$

$$\text{NRMSE} = \frac{\text{RMSE}}{y_{\max} - y_{\min}}, \tag{20}$$

$$\text{NMSE} = \frac{1}{\text{var}(y_i) \times N_t} \sum_{i=1}^{N_t} (y_i - \hat{y}_i)^2, \tag{21}$$

where $\hat{y}_i$ is the predicted value of corresponding observation y_i and N_t is the total number of the test set. $y_{\max}$ and $y_{\min}$

Table 2 Description of experimental datasets

Time series	Total size	Training set size	Test set size	Minimum	Maximum	Mean
Laser	10,093	5600	100	2	255	59.82
Sunspot	288	221	67	0	269.3	78.53
Poland	1601	1500	101	0.618	1.349	0.966
ESTSP2007	875	800	75	18.9	29.20	23.13

Table 3 Match degree obtained from different measurements

Dataset	Method	Query Sequence			
		1	2	3	4
Laser	DTW	0.936	0.973	0.716	0.231
	HD	0.956	0.972	0.971	0.984
	MSD	0.957	0.973	0.972	0.984
	VC	0.957	0.973	0.972	0.983
	CMSM	0.934	0.969	0.962	0.956
Sunspot	DTW	0.633	0.703	0.754	0.702
	HD	0.834	0.829	0.822	0.822
	MSD	0.836	0.826	0.820	0.823
	VC	0.838	0.833	0.824	0.818
	CMSM	0.782	0.769	0.725	0.751

Table 4 Errors of single-step prediction for two datasets

Model	Laser		Sunspot	
	MAE	RMSE	MAE	RMSE
DTW	1.185	2.449	19.76	25.57
HD	1.412	3.328	19.59	25.34
MSD	1.519	3.795	19.22	25.56
VC	1.281	3.045	18.31	24.56
CMSM	**0.987**	**1.554**	**17.53**	**24.04**

are the maximum and minimum values of the dataset. These four indices are very effective to evaluate absolute forecasting error.

5.1 Single-step prediction

The Laser and Sunspot datasets are used in single-step experiments. To verify the effectiveness of the CMSM, four existing methods are considered for comparison: Hybrid Distance (HD) (Wu and Lee 2015), Morphology Similarity Distance (MSD) (Men et al. 2015), DTW, and Vector Cosine (VC).

Experimental parameters of Laser and Sunspot datasets are set as $r = 10$ and $k = 50$. Table 3 shows the match degrees of four query sequences in Laser and Sunspot datasets, which illustrates the validity of searching the nearest neighbor set. The similarity measurement results are very similar among all the methods. Therefore, we can get better prediction results by modeling with the neighbor set, because the selected neighbor set and the test sequences are close to each other in numerical and trend.

The MAE and RMSE of prediction results are presented in Table 4, where bold values indicate the best results of all methods. The results show that the CMSM method achieves the minimum error when compared with other methods in single-step prediction. Figure 4 shows the corresponding forecasting results of the two datasets. All of these methods have a good approximation to the real values, which verify the effectiveness of local model in single-step prediction. Combined the error histogram in the details of Fig. 4 with Table 3, it can be seen that the similarity measurement proposed in this paper can get better prediction results based on local modeling among all these approaches. In addition, the prediction error (MAE, RMSE, NRMSE, NMSE) distribution obtained by modeling the Laser dataset using five different methods, respectively, is shown in Fig. 5. It also indicates that the CMSM method has the best prediction performance among the five methods.

The prediction performance is affected by the value of k. Figure 6 shows the MAE obtained by different measurement methods in Laser dataset, with k increasing from 30 to 150. It can be observed that the prediction errors of these methods have a decreasing trend with the increase of k, but the process is fluctuated. The CMSM method shows a good accuracy among all methods when k is small. Moreover, the MAE of the CMSM method owns the smallest variation with the increase of k, and thus its performance is stable. Although the errors of all methods are decreasing, the time complexity increases with the increase of k. Therefore, the relationship between the prediction accuracy and the time complexity should be considered in the selection of parameter k.

The prediction results of ANFIS, optimally pruned extreme learning machine (OPELM) (Grigorievskiy et al. 2014), back-propagation neural network (BPNN), SVM, and LSSVM combined with the CMSM method, respectively, are displayed in Fig. 7 in terms of MAE and RMSE in Laser dataset. This bar diagram shows that the LSSVM model achieves the minimum MAE and RMSE in Laser dataset. It is clear that the combination of LSSVM and CMSM is a good choice among all models.

5.2 Multi-step prediction

There are three main multi-step prediction strategies: direct strategy, recursive strategy, and combination strategy (Grigorievskiy et al. 2014). In recursive strategy, errors accumulate with the increase of the prediction step since each prediction has some error. Direct strategy based on true values of time series is more accurate than recursive. Combination strategy coincides with the direct strategy in the first step. But the dimension of the model increased because all predicted values act as new input. Therefore, direct strategy is selected in multi-step prediction experiments.

Laser and Sunspot datasets are used in the multi-step experiment, and the results are compared with HD. In Laser dataset, the sample size is $k = 150$, and in Sunspot dataset, the sample size is $k = 90$. The comparison results are shown in Table 5. As the table shows, the CMSM method outperforms HD in terms of RMSE. Figure 8 shows errors change of two datasets in different steps, and the errors become big-

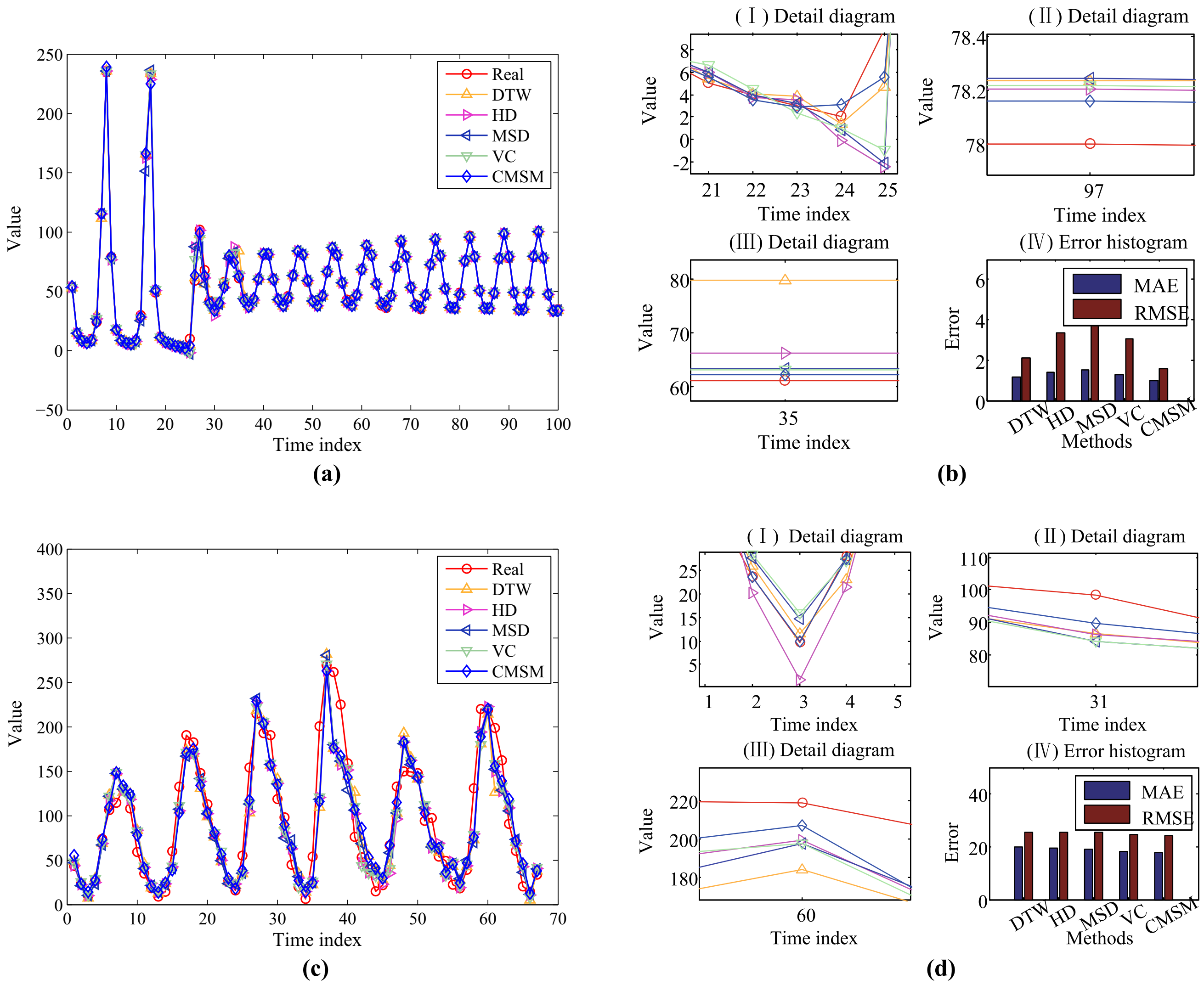

Fig. 4 Comparison of single-step prediction using the different methods in Laser and Sunspot datasets. **a** Laser result. **b** Laser details. **c** Sunspot result. **d** Sunspot details

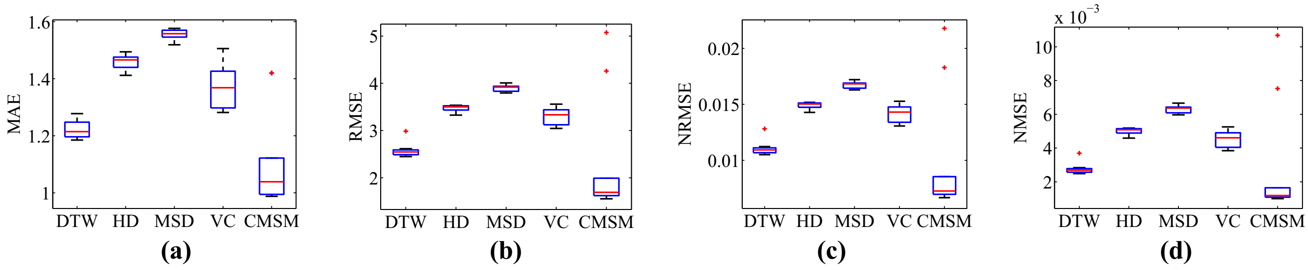

Fig. 5 Comparison of the error distribution with five methods using the box plot in Laser dataset. **a** MAE. **b** RMSE. **c** NRMSE. **d** NMSE

ger with the step increasing. CMSM method obtains a good prediction accuracy in small step. However, it cannot avoid accuracy decreasing when the step increases.

Table 6 shows multi-step prediction errors of four datasets in other steps. The bold values represent the best results. The error indices MAE and RMSE are used to evaluate prediction performances of different methods. The CMSM method significantly outperforms other methods in terms of both MAE and RMSE. The experimental results demonstrate the availability of CMSM method. Unlike CMSM time series forecasting method, these four methods are incapable of

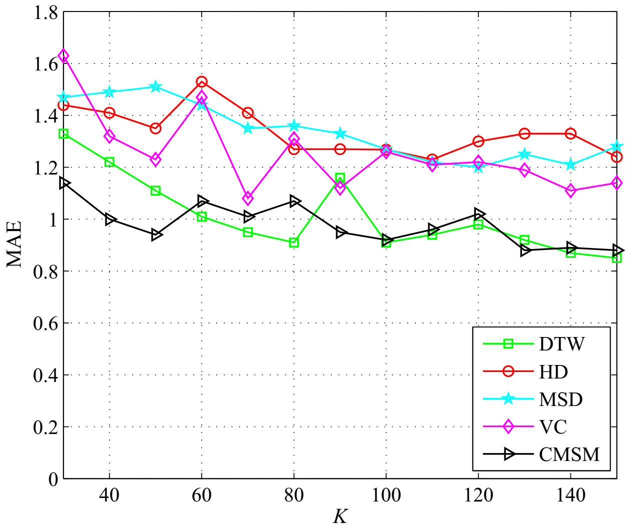

Fig. 6 Changes of MAE in Laser dataset with different *k*

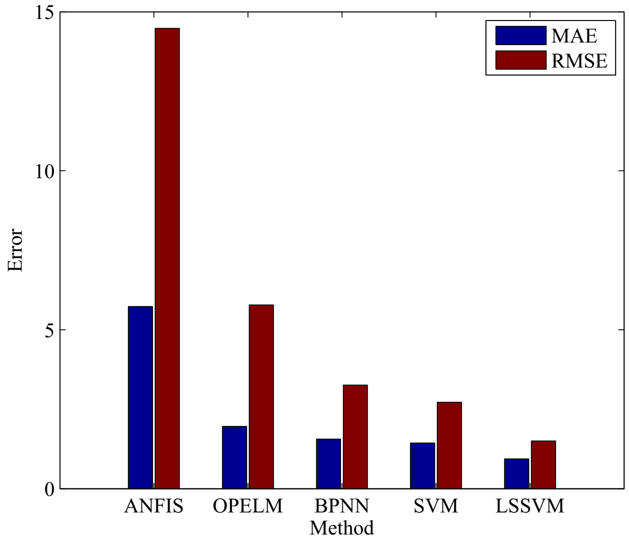

Fig. 7 Errors of Laser dataset obtained from various prediction models

dealing with the vague and missing data under uncertain circumstances, which leads to poor performance.

The proposed method based on CMSM, which uses the three characteristics to represent time series from the overall level and local trend, can avoid the influence of outliers to a certain extent. Therefore, there is a high degree of similarity between the neighbor set and query sequence. Figure 9 shows the average computation time in three different steps of four datasets for each method. On the one hand, computation time is related to the size of datasets. On the other hand, it is related to their own characteristics for all measurement methods. The sequence is segmented to measure similarity in the CMSM method. It results in slight increase of the computation time. However, it is still faster than the DTW method.

Table 5 RMSE performance comparison of multi-step prediction between HD and CMSM

Step	Laser		Sunspot	
	HD	CMSM	HD	CMSM
2	8.627	3.188	41.10	39.24
3	12.27	6.629	48.31	43.76
4	10.90	8.197	47.85	45.75
5	14.04	8.248	48.50	47.13
6	13.97	6.097	49.25	54.13
7	11.18	6.199	49.07	58.01
8	13.87	5.591	53.29	49.28
9	12.22	6.199	48.93	49.25
10	21.53	6.831	50.06	50.49
11	13.43	7.084	51.87	56.12
12	17.99	14.45	58.73	61.16

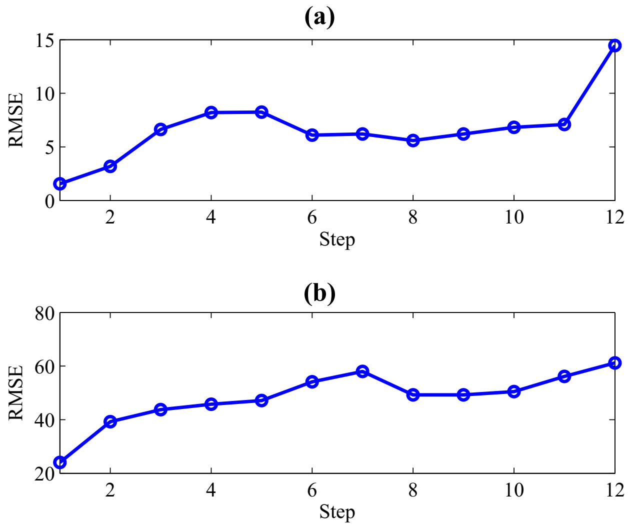

Fig. 8 Changes in RMSE for Laser and Sunspot datasets with increasing step in multi-step prediction

6 Conclusions

A time series forecasting method based on cloud model similarity measurement has been proposed in this paper. The raw and difference sequences of query sequence and training set are transformed into digital characteristic sequences by using the BCG algorithm, respectively. Then, OECM algorithm is employed to calculate their similarity, and the neighbor set of query sequence is selected. Finally, the LSSVM model is trained with the neighbor set and used for forecasting. The novelty in this method is the use of the cloud model and the obtainment of additional information from the raw series. Cloud model effectively solves the uncertainty of measuring similarity by transforming quantitative data into qualitative series. Two-tier features of series are extracted from the general perspective and local trend. This

Table 6 Errors of multi-step prediction for four datasets

Dataset	Step	Error	DTW	HD	MSD	VC	CMSM
Laser	5	MAE	2.945	2.652	3.154	3.030	**2.401**
		RMSE	10.40	8.619	10.23	8.932	**7.611**
	10	MAE	**2.860**	3.502	3.205	3.191	2.944
		RMSE	7.039	7.683	**6.650**	7.071	6.775
	15	MAE	4.669	4.745	4.871	5.332	**4.084**
		RMSE	9.735	11.15	11.26	10.86	**9.514**
Sunspot	5	MAE	36.65	34.39	32.46	33.38	**31.59**
		RMSE	53.42	49.42	47.91	49.49	**46.21**
	10	MAE	35.22	34.47	**34.20**	34.32	35.70
		RMSE	51.63	50.29	**49.75**	50.70	51.00
	15	MAE	51.73	53.27	51.07	50.77	**49.52**
		RMSE	73.13	73.04	70.39	69.66	**68.98**
Poland	5	MAE	0.029	0.026	0.026	**0.025**	0.027
		RMSE	0.043	0.041	0.040	**0.039**	0.042
	10	MAE	0.040	0.038	0.039	**0.034**	0.036
		RMSE	0.058	0.052	0.054	0.051	**0.050**
	15	MAE	0.041	0.043	0.046	0.043	**0.039**
		RMSE	0.059	0.060	0.064	0.063	**0.055**
ESTSP2007	5	MAE	0.776	**0.775**	0.806	0.812	0.792
		RMSE	1.037	0.990	1.014	1.037	**0.961**
	10	MAE	1.264	1.168	1.181	1.166	**1.120**
		RMSE	1.684	1.557	1.517	1.506	**1.406**
	15	MAE	1.606	1.453	1.474	1.426	**1.282**
		RMSE	2.168	2.023	1.967	1.976	**1.681**

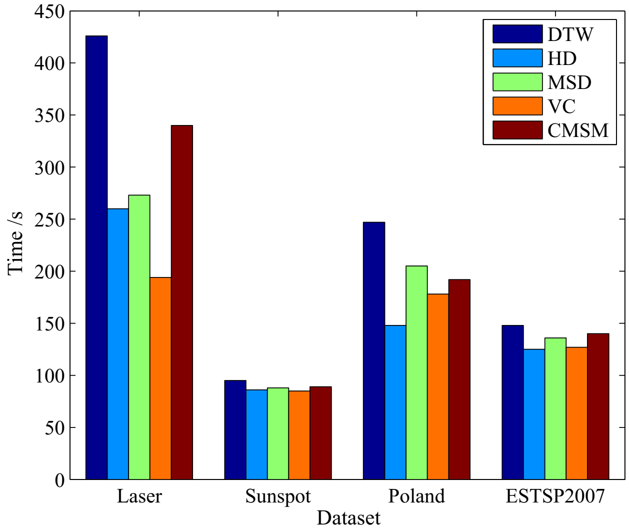

Fig. 9 Computation time comparison of various methods in all datasets

improves the measurement accuracy, which is favorable in subsequent prediction. The comparison of prediction results with different modeling methods shows that the proposed method is an excellent local prediction model. Meanwhile, the OECM algorithm and LSSVM are utilized for similarity measurement and prediction, respectively, and their combination obtains a good performance. Experimental results of the four time series have verified that the proposed method can achieve credible prediction accuracies in single-step and multi-step prediction.

In the proposed prediction model, the selection of time lags and the number of the nearest neighbor samples is a challenging task. Therefore, the next step we will focus on is selecting number of neighbor samples. In addition, the time complexity increases with the introduction of a variety of optimization methods. How to balance the relationship of forecast accuracy, forecast stability, and time complexity is worthy of further consideration.

Acknowledgements This work was supported by National Natural Science Foundation of China (61450011) and Natural Science Foundation of Shanxi (2015011052).

References

Adwan S, Arof H (2012) On improving dynamic time warping for pattern matching. Measurement 45(6):1690–1620

Arlot S, Celisse A (2010) A survey of cross-validation procedures for model selection. Stat Surv 4:40–79

Bao Y, Xiong T, Hu Z (2014) Multi-step-ahead time series prediction using multiple-output support vector regression. Neurocomputing 129:482–493

Bernecker T, Emrich T, Kriegel HP, Mamoulis N, Renz M, Zufle A (2011) A novel probabilistic pruning approach to speed up similarity queries in uncertain databases. In: 2011 IEEE 27th international conference on data engineering (ICDE). IEEE, pp 339–350

Bhardwaj S, Srivastava S, Gupta JRP (2013) Pattern-similarity-based model for time series prediction. Comput Intell 31(1):106–131

Cai SB, Fang W, Zhao J, Zhao YL, Gao ZG (2011) Research of interval-based cloud similarity comparison algorithm. J Chin Comput Syst 32(12):2457–2460

Chen TT, Lee SJ (2015) A weighted LS-SVM based learning system for time series forecasting. Inf Sci 299:99–116

Cherif A, Bon R (2014) Recurrent neural networks for local model prediction. Advances in cognitive neurodynamics (II). Springer, Dordrecht, pp 621–628

Chiu B, Keogh E, Lonardi S (2003) Probablistic discovery of time series motifs. In: Proceedings of the ninth ACM SIGKDD international conference on Knowledge discovery and data mining. ACM, pp 493–498

Coyle D, Prasad G, McGinnity TM (2005) A time-series prediction approach for feature extraction in a brain–computer interface. IEEE Trans Neural Syst Rehabil Eng 13(4):461–467

De Gooijer JG, Hyndman RJ (2006) 25 years of time series forecasting. Int J Forecast 22(3):443–473

Giles CL, Lawrence S, Tsoi AC (2001) Noisy time series prediction using recurrent neural networks and grammatical inference. Mach Learn 44(1–2):161–183

Goel H, Melnyk I, Banerjee A (2017) R2N2: residual recurrent neural networks for multivariate time series forecasting arXiv:1709.03159

Grigorievskiy A, Miche Y, Ventela AM, Severin E, Lendasse A (2014) Long-term time series prediction using OP-ELM. Neural Netw 51:50–56

Hyndman RJ (2013) Time Series Data Library (TSDL). http://robjhyndman.com/TSDL/. (01-01-13)

Jayawardena AW, Li WK, Xu P (2002) Neighbourhood selection for local modelling and prediction of hydrological time series. J Hydrol 258(1):40–57

Jiang R, Li DY, Chen H (2000) Time-series prediction with cloud models in DMKD. Methodol Knowl Discov Data Min 1(5):13–18

Keogh E, Chakrabarti K, Pazzani M, Mehrotra S (2001) Dimensionality reduction for fast similarity search in large time series databases. Knowl Inf Syst 3(3):263–286

Khashei M, Bijari M (2010) An artificial neural network (p, d, q) model. Expert Syst Appl 37(1):479–489

Korn F, Muthukrishnan S (2000) Influence sets based on reverse nearest neighbor queries. ACM SIGMOD Rec 29(2):201–212

Li DY (2000) Uncertainty in knowledge representation. Eng Sci 10:73–79

Li DY, Du Y (2007) Artificial intelligence with uncertainty. CRC Press, Boca Raton

Li HL, Guo CH, Qiu WR (2011) Similarity measurement between normal cloud models. Dianzi Xuebao (Acta Electronica Sinica) 39(11):2562–2567

Lin J, Keogh E, Lonardi S (2003) A symbolic representation of time series, with implications for streaming algorithms. In: Proceedings of the 8th ACM SIGMOD workshop on research issues in data mining and knowledge discovery, pp 2–11

Liu CY, Feng M, Dai XJ, Li DY (2004) A new algorithm of backward cloud. J Syst Simul 16(11):2417–2420

Lu J, Qin SY (2014) Similarity measurement between cloud models based on close degree. Appl Res Comput 31(5):1308–1311

McNames J (1998) A nearest trajectory strategy for time series prediction. In: Proceedings of the international workshop on advanced black-box techniques for nonlinear modeling. KU Leuven, Belgium, pp 112–128

Men LS, Wei JF, Li Z (2015) Morphology similarity distance based time series similarity measurement. Comput Eng Appl 51(4):120–122

Miranian A, Abdollahzade M (2013) Developing a local least-squares support vector machines-based neuro-fuzzy model for nonlinear and chaotic time series prediction. IEEE Trans Neural Netw Learn Syst 24(2):207–218

Popivanov I, Miller RJ (2002) Similarity search over time-series data using wavelets. In: 18th international conference on data engineering, Proceedings. IEEE, pp 212–221

Ruta D, Gabrys B, Lemke C (2011) A generic multilevel architecture for time series prediction. IEEE Trans Knowl Data Eng 23(3):350–359

Saimurugan M, Ramachandran KI, Sugumaran V, Sakthivel NR (2011) Multi component fault diagnosis of rotational mechanical system based on decision tree and support vector machine. Expert Syst Appl 38(4):3819–3826

Sisman-Yilmaz NA, Alpaslan FN, Jain L (2004) ANFIS unfolded in time for multivariate time series forecasting. Neurocomputing 61:139–168

Sun NN, Chen ZH, Niu YG, Yan GW (2015) Similarity measurement between cloud models based on overlap degree. J Comput Appl 35(7):1955–1958, 1964

Suykens JA, Van Gestel T, De Moor B, Vandewalle J (2002) Basic methods of least squares support vector machines. Least Squares Support Vector Machines. World Scientific Publishing Co. Pte. Ltd, Singapore

Van Heeswijk M, Miche Y, Lindh-Knuutila T, Hilbers PA, Honkela T, Oja E, Lendasse A (2009) Adaptive ensemble models of extreme learning machines for time series prediction. In: Artificial Neural Networks-ICANN 2009. Springer, Berlin, pp 305–314

Veloz A, Salas R, Allende-Cid H, Allende H, Moraga C (2016) Identification of lags in nonlinear autoregressive time series using a flexible fuzzy model. Neural Process Lett 43(3):641–666

Wang X, Mueen A, Ding H, Trajcevski G, Scheuermann P, Keogh E (2013) Experimental comparison of representation methods and distance measures for time series data. Data Min Knowl Discov 26(2):275–309

Wu SF, Lee SJ (2015) Employing local modeling in machine learning based methods for time-series prediction. Expert Syst Appl 42:341–354

Xiong T, Bao Y, Hu Z (2013) Beyond one-step-ahead forecasting: evaluation of alter-native multi-step-ahead forecasting models for crude oil prices. Energy Econ 40:405–415

Zhang Y, Zhao D, Li DY (2004) The similar cloud and the measurement method. Inf Control 33(2):130–132

Zhang GW, Li DY, Li P, Kang JC, Chen GS (2007) A collaborative filtering recommendation algorithm based on cloud model. J Softw 18(10):2404–2411