Mobile Networks and Applications (2022) 27:371–381
https://doi.org/10.1007/s11036-021-01837-y

An Adaptive IoT Network Security Situation Prediction Model

Hongyu Yang[1] · **Le Zhang**[1] · **Xugao Zhang**[1] · **Jiyong Zhang**[2]

Accepted: 30 January 2021 / Published online: 27 October 2021

Abstract
With the rapid development of the Internet of things (IoT) technology, how to effectively predict the network security situation of the IoT has become particularly important. It is difficult to quantify the IoT network situation due to a large number of historical data dimensions, and there are also has the problem of low accuracy for IoT network security situation prediction with multi-peak changes. To solve the above problems, this paper proposed an adaptive IoT network security situation prediction model, which makes the IoT network security situation prediction accuracy higher. Firstly, the paper used the entropy correlation method to calculate the network security situation value sequence in each quantization period according to Alarm Frequency (AF), Alarm Criticality (AC), and Alarm Severity (AS). Then, the security situation values arranged in time series are fragmented through the sliding window mechanism, and then the adaptive cubic exponential smoothing method is used to initially generate the IoT network security situation prediction results. Finally, the paper built the time-varying weighted Markov chain to predict the error value and modify the initial predicted value based on the error state. The experimental results show that the model has a better fitting effect and higher prediction accuracy than other models, and this model's determination coefficient is 0.811. Compared with the other two models, the sum of squared errors in this model is reduced by 78 %-82 %. The model can better reflect the changes in the IoT network security situation over a while.

Keywords Network security situation prediction · Internet of Things · Alarm element · Entropy correlation · Cubic exponential smoothing · Time-varying weighted Markov chain

1 Introduction

Today, although the Internet of things (IoT) has brought people convenience, also brings people security problems, such as communication interruption, privacy disclosure, information tampering, unsafe driving, etc. To solve these problems, Tang et al. [1] propose a new framework called Unmanned Aerial Vehicles (UAV) enabled social internet of vehicles. FLAUZAC et al. [2] proposed a network security framework for the IoT based on software-defined networking (SDN), which can well solve the security problems of wired and wireless networks. Therefore, how to effectively resolve the IoT security problems is necessary. To avoid losses in all aspects, the situation prediction of IoT network security is particularly important. According to network architecture, to complete the prediction and evaluation of the network security situation has become a research hotspot.

The IoT network security situation prediction forms a nonlinear time series through the factors affecting IoT network security. According to historical data and network status, the IoT security situation is predicted in a future period through a specific mathematical model, which facilitates network management personnel to detect threats in time and take corresponding protective measures.

The current IoT network security prediction methods include the gray prediction method, the prediction based on time series, and prediction based on the neural network [3]. Zhang et al. [4] proposed a network security situation prediction model based on multi swarm chaotic particle optimization. The key parameters of the grey neural network are optimized by using multi-population chaotic particles to improve the prediction effect of network conditions. Xiao et al. [5] proposed a network security situation prediction method based on MEA-BP. The method uses the Mind Evolution

✉ Hongyu Yang
yhyxlx@hotmail.com

[1] School of Computer Science and Technology, Civil Aviation University of China, Tianjin 300300, China

[2] School of Computer and Communication Science, Swiss Federal Institute of Technology in Lausanne, CH-1015 Lausanne, Switzerland

Algorithm (MEA) to optimize the weight and threshold of the BP neural network to improve the prediction accuracy and efficiency of the security situation, but the standardization of historical data is not perfect. Sun et al. [6] proposed a Markov prediction model based on complex networks. The model constructs the transformation relationship of network security status into a complex network and uses the weighted Markov chain to predict the security situation, which can reflect the security status of the network to a certain extent, but the state transition probability matrix constructed by the multi-state network is too large. Zhou et al. [7] proposed a network multi-node security situation prediction model based on an improved G-K algorithm. The model extracts the main factors affecting network security by the grey entropy correlation method. Based on this, the Kalman filter is established to improve the accuracy of security situation prediction. Zhang et al. [8] reduced the training complexity of the neural network situation prediction model by improving the convolutional neural network and improved the efficiency of network security situation prediction, but the quality of extracted features needs to be improved.

Because of the uneven quality of the historical data in the above-mentioned IoT network security situation prediction methods, and the deficiency of the IoT network security situation prediction accuracy with multi-peak changes, this paper proposes an adaptive IoT network security situation prediction model to improve the accuracy of network security situation prediction.

1.1 Proposed Approach

The network security situation prediction model for the IoT proposed in this paper is shown in Fig. 1.

The network security situation prediction for the IoT process is designed as follows:

Step 1. Generating a non-linear time series of security situation values by using an entropy correlation method based on the network alarm information;
Step 2. Using a sliding window to divide the IoT network security situation value sequence segment, and each time an IoT network security situation value is updated, the sliding window slides backward by one unit;
Step 3. Establish a three-dimensional exponential smoothing prediction model based on the security situation sequence in the sliding window, and adaptively adjust the static smoothing coefficient α to improve the prediction accuracy of the module;
Step 4. Calculate the error between the security situation predicted and the security situation actual value in the sliding window, and divide the error into n error intervals, which are recorded as n error states. Using a time-varying weighted Markov chain to predict the error value and correct the situation predictor;
Step 5. Check the error. If the threshold condition is not met, return to step 4 and divide the error state into $n+1$; if the threshold condition is met, obtain the next cycle security situation value according to steps 1–4.

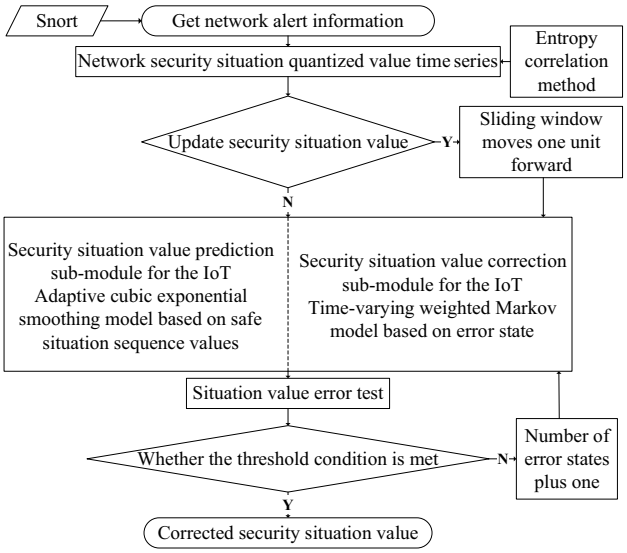

Fig. 1 Network security situation prediction model

1.2 Contribution

The contribution of this paper is described as follows:

(1) The author uses the sliding window mechanism to limit the length of the historical data sequence on which the cubic exponential smoothing prediction is based, because of the failure problem of the exponential smoothing prediction method in the long time series;
(2) In this paper, the error between the predicted value and the actual value is divided. The time-varying weighted Markov chain is used to predict the error of the next cycle according to the known error state. The predicted value is corrected by the error obtained from the prediction.

1.3 Organization

This paper is organized as follows. Section 2 describes the quantification of network security situation assessment for the IoT. Section 3 describes the network security situation prediction sub-module for the IoT. Section 4 describes the predictive value correction sub-module. Section 5 analyzes the experimental results. Section 6 describes the conclusion.

2 Quantification of IoT Network Security Situation Assessment

Firstly, the alarm information is acquired based on the Snort intrusion detection system. Then, according to the entropy correlation method, the IoT network security situation value in each quantization period is calculated. The specific design method is as follows:

The IoT network security situation quantified value is determined by Alarm Frequency (AF), Alarm Criticality (AC), and Alarm Severity (AS). Referring to the alarming quality quantification method [9], the observation vector obtained by this method based on the alarming quality can effectively improve the data source and the evaluation accuracy. Therefore, the alarm with the highest quality quantization value is selected as the basis for quantifying the IoT network security situation in each cycle.

Assuming a total of T quantization periods, let the IoT network security situation quantization value of the ith cycle is V_i ($i=1, 2,\ldots, T$), and let the alarm with the highest quality quantization value of the ith cycle is H_i, then define $V_i = V_i$ (AF_{Hi}, AC_{Hi}, AS_{Hi}) ($i=1, 2,\ldots, T$), where,

$$AF_{Hi} = \frac{H_i \ number \ of \ alarms \ in \ T_i}{Number \ of \ all \ alarms \ in \ T_i} \tag{1}$$

AC_{Hi} is the critical degree of the alarm H_i, indicating the possibility that the occurrence of the alarm H_i causes a change in the network security state. When AC_{Hi} is higher, it means that the IoT network security state is more likely to change. If the alarm AC_{Hi} is an alarm that has occurred in the period i, its priority is set to 1; If the alarm is generated in the period i-N to the period i-1, the priority is 2; If there is no alarm in the AC_{Hi} from period i-N to period i-1, the priority is set to 3. According to the intrusion detection alarm aggregation association algorithm [10], this paper takes $N=2$.

AS_{Hi} is the severity of the alarm, indicating the negative impact of the alarm on the network. The greater the severity, the greater the impact of H_i on the IoT network security status. In this paper, the severity of the alarm is divided into low, general, and high, and the corresponding priority values are 1, 2, and 3.

To quantify the IoT network security situation of the period i, the comment support matrix P is set as shown in Table1. Let $X_1 = AF_{Hi}$, $X_2 = AC_{Hi}$, $X_3 = AS_{Hi}$, then X_1, X_2, X_3 correspond to the three quantitative indicators of the alarm H_i with the highest alarm quality in period i, namely alarm frequency, alarm criticality, and alarm severity; p_{ij} indicates the support degree of the ith index for the jth comment ($i, j=1, 2, 3$)

To distinguish the severity of the network security threat from indicator X_1, the alarm occurrence rate interval c_j is set as shown in Table 2. Calculate the correlation between X_1 and the interval c_j ($j=3,2,1$) at the current time based on the distance between X_1 and the endpoint of the alarm occurrence interval. This correlation is the correlation between X_1 and the evaluations in Table 1.

Let $X_1 = x$, then formula (2) is used to calculate the support degree of the index for each comment:

$$P_{ij} = \frac{1 - \left|x - \left((a_j + b_j)/2\right)\right| + (b_j - a_j)/2}{\sum_{j=1}^{3}\left[1 - \left|x - \left((a_j + b_j)/2\right)\right| + (b_j - a_j)/2\right]} \tag{2}$$

where, a_j and b_j respectively correspond to the lower endpoint and the upper endpoint of the interval c_j, $j=1, 2, 3$.

The support for X_2 and X_3 for each comment is shown in Table 3.

In Table 3, the higher the priority, the greater the threat degree of the index. Therefore, when the correlation degree of high risk, medium risk and low risk is evaluated with low priority, the correlation degree increases in turn. On the contrary, it decreases. When the priority of the indicator X_i ($i=2,3$) is j, the correlation degree in the same row of Table 3 and j is taken as the correlation degree corresponding to X_i ($i=2,3$) in Table 1. Use formula (3) to calculate the absolute entropy of each indicator of the alarm:

$$H_i = -\sum_{j=1}^{n} p_{ij} \ln p_{ij} \tag{3}$$

when $p_{i1} = p_{i2} = \ldots = p_{in}$, $H_{max} = \ln n$, the relative entropy values of the alarm indicators are:

Table 1 Comment support matrix

Index	Low	General	High
X_1	P_{11}	P_{12}	P_{13}
X_2	P_{21}	P_{22}	P_{23}
X_3	P_{31}	P_{32}	P_{33}

Table 2 X_1 comment interval

Comment	Low	General	High
Frequency interval c_j	[0,0.3)	[0.3,0.7)	[0.7,1]

Table 3 X_2, X_3 Comment support scale

Priority	Low	General	High
1	0.5	0.333	0.167
2	0.25	0.5	0.25
3	0.167	0.333	0.5

$$\mu_i = -\frac{1}{\ln n}\sum_{j=1}^{n} p_{ij}\ln p_{ij} \tag{4}$$

If the relative entropy value of an indicator is larger, it means that the indicator has less influence on the quantified value of the alarm. Let (1-μ_i) denote the weight of the corresponding indicator, namely:

$$\tau_i = \frac{1}{n-\sum_{i=1}^{n}\mu_i}(1-\mu_i) \tag{5}$$

Where, $\tau_i \in [0, 1]$ and $\tau_1+\ldots+\tau_n = 1$. τ_i is the entropy weight coefficient of the index X_i. The vector of the comment weight is $\boldsymbol{W} = (w_{low}, w_{normal}, w_{high}) = (1/5, 1/3, 7/15)$[11]. In the ith cycle, the quantitative result of network security situation is calculated[12] as follows:

$$V_i = \mu \cdot \tau \cdot \mathbf{P} \cdot \mathbf{W}^T \tag{6}$$

Where μ is the correction factor. To facilitate data processing, this paper takes $\mu = 10{,}000$. If the situation quantitative value is higher, the network security situation is worse.

3 Network Security Situation Prediction Sub-module for the IoT

3.1 Sliding Window Mechanism

Given the failure of the exponential smoothing prediction method under long time series, this paper limits the length of historical data sequence based on the sliding window mechanism.

Let the sliding window width be L (L is a positive integer), and the current IoT network security situation values are arranged in chronological order as V_1, V_2,…, V_m (m is a positive integer), then the sliding window mechanism is designed as follows:

(1) If the number of security situation values in the sliding window is k ($1 \le k \le m$), the sequence of security situation values in the width of the sliding window is V_1', V_2',…, V_k'. If $k+1 \le L$, the window does not move, predicting the $m+1$th security situation value and waiting for a new security situation value to enter the window.
(2) If $k+1 > L$, the sliding window moves forward by one unit when a new security situation value is added to the sequence. Based on the sequence value in the new window, the security situation value of the m + 1th cycle is predicted.

The schematic diagram of the sliding window mechanism is shown in Fig. 2. The mechanism ensures that the length of time series based on the cubic exponential smoothing method does not exceed L. When the new security situation value is added to the historical sequence, the mechanism ensures that the cubic exponential smoothing method can still predict normally. It can improve the accuracy and dynamic of security situation value prediction.

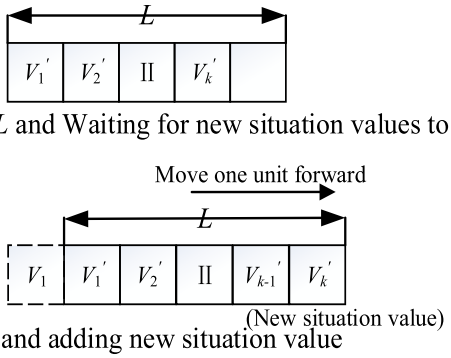

Fig. 2 The sliding window mechanism schema

3.2 Adaptive Cubic Exponential Smoothing Model

Let the IoT network security situation value of m period currently have V_1, V_2, …, V_m, and there are k security situation values in the sliding window width. If $m \le L$, then $V_1' = V_1$ and $V_k' = V_m$; if $t > L$, then $V_1' = V_{m-L+1}$, and $V_k' = V_m$. The calculation steps of the model are as follows:

$$V_{t+T}^1 = a_t + b_t T + c_t T^2 \tag{7}$$

where the quantitative prediction result of the security situation of period $t+T$ is $V^1{}_{t+T}$. The prediction period advance quantity is T, and a_t, b_t and c_t are the prediction coefficients of the tth period.

$$a_t = 3s_t^{(1)} - 3s_t^{(2)} + s_t^{(3)} \tag{8}$$

where, $s_t{}^{(1)}$, $s_t{}^{(2)}$ and $s_t{}^{(3)}$ are the 1, 2, and 3 adjustment coefficients of period t respectively.

$$b_t = \frac{\alpha}{2(1-\alpha)^2}\Big[(6-5\alpha)s_t^{(1)} - 2(5-4\alpha)s_t^{(2)} + (4-3\alpha)s_t^{(3)}\Big] \tag{9}$$

$$c_t = \frac{\alpha^2}{2(1-\alpha)^2}\left(s_t^{(1)} - 2s_t^{(2)} + s_t^{(3)}\right) \tag{10}$$

Where, $\alpha \in [0, 1]$ is the static adjustment coefficient.

Let $s_{t-1}{}^{(1)}$, $s_{t-1}{}^{(2)}$ and $s_{t-1}{}^{(3)}$ be the initial values of the first, second and third exponential smoothing of the tth period, then

$$s_t^{(1)} = \alpha X_t + (1-\alpha)s_{t-1}^{(1)} \tag{11}$$

$$s_t^{(2)} = \alpha s_t^{(1)} + (1-\alpha)s_{t-1}^{(2)} \tag{12}$$

$$s_t^{(3)} = \alpha s_t^{(2)} + (1-\alpha)s_{t-1}^{(3)} \tag{13}$$

where X_t is the actual situation value of period t.

The initial value of the smoothing index is $s_0^{(1)}=s_0^{(2)}=s_0^{(3)} = (V_1'+ V_2'+ V_3')/3$; α is the static smoothing coefficient, and $\alpha\in[0, 1]$. Its value indirectly affects the final prediction accuracy.

Generally, when the actual value sequence shows a horizontal trend, $\alpha\in[0.05, 0.2]$; when the actual value sequence fluctuates, but the long-term fluctuation is small, $\alpha\in[0.3,0.5]$; When the actual value sequence fluctuates greatly, showing an obvious upward or downward trend, $\alpha\in[0.6,0.8]$. The larger the value of α, the greater the impact of long-term data on the predicted value. In this paper, we propose to minimize the sum of absolute errors between predicted and actual values, and then obtain the optimal dynamic solution of α. The optimal dynamic solution process α is designed as follows:

Step 1. It is assumed that the kth IoT network security situation actual values in the current sliding window constitute a vector $V'= (V_1', V_2',\ldots, V_k')$. The initial value of static smoothing coefficient α is 0;
Step 2. It is known that $s_0^{(1)}=s_0^{(2)}=s_0^{(3)}= (V_1'+ V_2'+ V_3')/3$, and $X_1=V_1'$. From formulas (11)–(13), $s_t^{(1)}$, $s_t^{(2)}$ and $s_t^{(3)}$ $(t=0,1,\ldots,k)$ are obtained. From the formulas (8)–(10) ,a_t, b_t and c_t are obtained $(t=0, 1,\ldots, k)$;
Step 3. In this step, $t=0, 1,\ldots,k$-1, and the prediction period T is 1. According to Eq. (7), the predicted value sequence $V_1=(V_1^1, V_2^1,\ldots, V_k^1)$ based on the current static smoothing coefficient is obtained;
Step 4. The sum of the absolute values of the error of the predicted value sequence and the actual value sequence is $E=\sum_{i=1}^{k}\left|V_i^1 - V_i'\right|$, $\alpha=\alpha+0.001$;
Step 5. Repeat steps 1–4 to $\alpha=1$, and record the absolute error generated by each cycle as E_j $(j=0, 1,\ldots, 1000)$, and obtain min $\{ E_j \}$ $(j=0, 1,\ldots, 1000)$ .The corresponding α value is taken as the optimal dynamic solution of the static smoothing coefficient under the current sliding window and is denoted as α_{best};
Step 6. Let $t=k=m$, $\alpha=\alpha_{best}$, $T=1$. The security situation value of the m + 1th cycle can be obtained by formula (7)–(13).

4 Predictive Value Correction Sub-module

Through investigation, it is found that trust plays an important role in the security of the Internet of Things [13, 14]. Therefore, in order to make the predicted value more reliable, the error correction strategy is adopted in this paper. According to the theoretical analysis, there is an error between the known initial prediction value and the known security situation actual value in the same window. And the error is related to the fluctuation of the security situation in the sliding window. To reduce the error between the actual value and the predicted value of the situation, this paper proposes a time-varying weighted Markov correction model based on the error state.

4.1 Error State Division

The vulnerabilities and threats in the network at different times will change. The possible situations are as follows:

(1) In a short period of time, the network is attacked intensively, which leads to the great fluctuation of its security situation. The distance between the upper limit and lower limit of the error between the predicted value and the actual value of the security situation is large;
(2) The network is faced with conventional vulnerabilities, so its security situation sometimes will be relatively gentle or fluctuate slightly. The distance between the upper and lower limits of the error between the predicted value and the actual value of the security situation is small.

With the addition of new quantitative value of security situation, the sliding window moves. The volatility of network security situation sequence contained in the window will change. The distance between the upper and lower limits of the error between the actual value and the predicted value will also change. There are k known security situation values in the current sliding window, taking $V=\{V_i'|i=1,2,3,\ldots,k\}$, the corresponding security situation prediction value is $V^1=\{V_i^1|i=1,2,3,\ldots,k\}$, the lower limit of the error is F^L=min$\{ V_i^1- V_i'| i=1,2,3,\ldots,k \}$, and the upper limit of the error is F^U= max$\{ V_i^1- V_i'| i=1,2,3,\ldots,k \}$.The distance between the upper and lower limits of the error is denoted as $FL=F^U- F^L$. The process of dividing the error state is designed as follows:

Step 1. The upper and lower limits of the error are divided into n intervals. The interval length is FL/n .The interval range is $[F^L, F^L + FL/n)$, $[F^L + FL/n, F^L + 2FL/n)$, …, $F^L +(n\text{-}1) \bullet FL/n, F^U]$;

Step 2. The sequence of error values in the current sliding window is $F=\{F_i=V_i^1-V_i' | i=2,3,\ldots,k\}$. If $F_i \in [F^L+(j-1)\bullet FL/n, F^L+j\bullet FL/n)$, then the error F_i is in the error state j, where $j\in\{1,2,\ldots,n\}$. In particular, when $F_i=F^U$, F_i is considered to be in state n;
Step 3. If the predicted value does not satisfy the threshold test requirement after the error correction, the number of error states needs to be increased, that is, $n=n+1$ to refine the error correction result.

4.2 Time-varying Weighted Markov Chain Based on Error State

Based on the sequence of error states in the current sliding window, this paper uses the time-varying weighted Markov chain to predict the error value. The error prediction process is designed as follows:

Step 1. Determine an error state transition probability matrix. There are currently n error states. The current period is t. Let the error state of adjacent time be $f_{t-1}f_t$. If the error state after q cycles is recorded as f_{t+q}, then.

$$p_{ijr}=p\{f_{t+q}=r|f_{t-1}=i,f_t=j\},i,j,r\in\{1,2,\ldots,n\}.$$

Where p_{ijr} represents the error state of the period t-1 as i. The error state of the period t is j. The error state is r after q cycles. The probability is obtained by a statistical method. When the initial value of the error state number n is 3, the q-order error state transition probability matrix is

$$\boldsymbol{P}^q_{(n\times n)\times n}=\begin{pmatrix} p_{111} & p_{112} & \cdots & p_{11n} \\ p_{121} & p_{122} & \cdots & p_{12n} \\ \vdots & \vdots & \ddots & \vdots \\ p_{nn1} & p_{nn2} & \cdots & p_{nnn} \end{pmatrix} \tag{14}$$

where, $q=1,2,\ldots,\beta$. In this paper, $\beta_0=[L/3]$, L is the sliding window width, and the β value adjustment is determined by step 3;

Step 2. Calculate the weight of each order error state transition probability matrix. First, calculate the correlation coefficient η_q between $f_{t-1}f_t$ and f_{t+q}

$$\eta_q=\frac{\sum_{t=1}^{n-q}(y_{t-1}+y_t-2\bar{y})(y_{t+q}-\bar{y})}{\sqrt{\sum_{t=1}^{n-q}(y_{t-1}+y_t-2\bar{y})^2\sum_{t=1}^{n-q}(y_{t+q}-\bar{y})^2}} \tag{15}$$

where, $q=1,2,\ldots,\beta$; y_{t-1}, y_t, y_{t-q} respectively means the error value of period t-1, period t, and period $t+q$ in the original error sequence in the current window. $\bar{y}$represents the average of the original error sequence in the current window.

Then the q-order error state transition probability matrix weight w_q is

$$w_q=\frac{|\eta_q|}{\sum_{q=1}^{\beta}|\eta_q|},\ q=1,2,\ldots\beta \tag{16}$$

Step 3. Adjust the value of β. Check the value of w_β. The weight threshold of the error state transition probability matrix is 0.05[15]. If $w_\beta<0.05$, it is shown that the prediction effect of the A-Order error state transition probability matrix can be ignored. This step discards the matrix. Let $\beta=\beta$-1, this step recalculate the value of w_β until $w_\beta\geq 0.05$, at this time $q_{max}=\beta$;
Step 4. The error of the predicted value of the security situation in the current window is predicted. The probability that the error value of the period $t+1$ is in the error state r ($r=1,2,\ldots,n$) is $p_{r(t+1)}$

$$p_{r(t+1)}=\sum_{q=1}^{\beta}p_{ijr}^{(q)}\cdot w_q \tag{17}$$

where, $q=1,2,\ldots,\beta$;$i,j\in\{1,2,\ldots,n\}$,$p_{ijr}^{(q)}$ is taken from the q-order error state transition probability matrix $\boldsymbol{P}^q$, which represents probability of the adjacent error state $f_{t-q}=i$, $f_{t-q+1}=j$ steering error state $f_{t+1}=r$. w_q is the q-order error state transition probability matrix weight. The error state probability distribution vector of period $t+1$ is $\boldsymbol{P}_{r(t+1)}=\{p_{1(t+1)},p_{2(t+1)},\ldots,p_{n(t+1)}\}$.

Let the error median vector composed of the median values of each error interval be

$$\mathbf{F_{mid}}=\{[F^L+(F^L+FL/n)]/2,[F^L+FL/n+(F^L+2FL/n)]/2,\ldots,[F^L+(n-1)\cdot FL/n+F^U]/2\}$$

Then the error prediction value operator at time $t+1$ is

$$F'_{t+1}=\mathbf{P}_{r(t+1)}\cdot\mathbf{F}_{mid} \tag{18}$$

At the time of $t+1$, the correction result of the predicted value is

$$V_{c(t+1)}=V^1_{(t+1)}-F'_{(t+1)} \tag{19}$$

where $V_{(t+1)}{}^1$ is the uncorrected security situation prediction value based on the sub module of network security situation prediction.

4.3 Threshold Test

Firstly, the proximity between the actual value and the predicted value is analyzed. Then, it is judged whether the number of partition n of error state is enough. It is known that the sequence of corrected security situation predictors and the actual value sequence in a window is as shown in Table 4.

Table 4 Sequence of predicted and actual values

Correction value	$V_{C(2)}$	$V_{C(3)}$	…	$V_{C(k)}$
Actual value	V_2	V_3	…	V_k

The methods for judging the accuracy of prediction in this paper are:

(1) Post-test difference test: The author records R_i as residual, which is the difference between the actual situation value V_i and the revised predicted situation value $V_{c(i)}$ in a certain period of time. R_i=V_i-$V_{c(i)}$, i=2,3,…,k. The security situation value variance ${S_1}^2$ in the current security situation sequence segment is

$${S_1}^2=\frac{1}{k}\sum_{i=2}^{n}(V_i-\frac{1}{k}\sum_{i=2}^{k}V_i)^2 \tag{20}$$

The residual sequence variance ${S_2}^2$ is calculated by the formula (21):

$${S_2}^2=\frac{1}{k}\sum_{i=2}^{k}(R_i-\frac{1}{k}\sum_{i=2}^{k}R_i)^2 \tag{21}$$

Then, the posterior difference ratio $c=S_2/S_1$. The smaller the value of c, the better the prediction accuracy.

(2) Small probability test: A small probability test result P is obtained from formula (22). The larger the value of P, the better the prediction accuracy.

$$P=P(|R_i-\frac{1}{k}\sum_{i=2}^{k}R_i|<0.6745S_1) \tag{22}$$

According to c and P, comparing with the prediction accuracy grade table (as shown in Table 5) to determine whether the number of error state divisions should be increased. If the prediction result of the model is first-level prediction accuracy or second-level prediction accuracy, there is no need to increase the number of error state division. Otherwise, the number of error state division is $n+1$.

Table 5 Rank of prediction accuracy

Prediction accuracy level	c	P
First level	<0.35	>0.95
Second level	<0.50	>0.80
Third level	<0.65	>0.70
Fourth level	≥0.70	≥0.65

5 Experimental Results and Analysis

The predictive validity of the model is verified using Lincoln Laboratory's standard dataset LL_DOS_1.0. This data set is the most comprehensive attack test data set, and also serves as the common and widely used benchmark data set in the research field. Therefore, the data set is relatively complete and rich, which is suitable for most networks. The LL_DOS_1.0 attack process is:

1-70 min: The attacker installs the relevant attack software and scans the experimental network topology through IP Sweep to find the currently active host;
71-125 min: Use Sadmind Ping to find hosts with Sadmind vulnerabilities;
126-240 min: The attacker uses Sadmind Exploit to attack the three hosts locked in step 2, Pascal, Mill, and Locke until they invade each host system;
241-319 min: The attacker installs the DDOS Trojan on the three hosts that were attacked by the intruder;
After 320 min: The attacker launched a DDOS attack on the remote server.

5.1 Experimental Data Processing

Under the Ubuntu 16.04 operating system, the LL_DOS_1.0 packet is replayed using the Tcpreplay technology. The Snort intrusion detection system is used to generate an alarm log for the replay traffic under the Windows 10 operating system.

According to Section 2, the IoT network security situation is quantified. The quantization period T is set to 4 min. 90 security situation values in the interval [2800, 4000] are generated in 1-360 min. In this paper, 10 security situation values within 1-40 min are taken as the actual security situation sequence. The actual value of 80 network security situation and the corresponding network security situation prediction were compared in 41-360 min to test the prediction effect of the model.

The model prediction process is illustrated by taking 10 security situation values within 40 min of 41-360 min.

The preferred alarm H, the alarm frequency AF_H, the alarm criticality AC_H level, and the alarm severity AS_H level in a certain quantization period T=4 min are shown in Table 6.

According to formula (2) and Tables 1, 2 and 3, a list of comment support matrices is obtained (as shown in Table 7)

The network security situation quantization value of the period calculated by the formulas (2)–(6) is: V=3504. The

Table 6 Alert property sample

H	AF_H	AC_H	AS_H
Sensitive data Email address	0.25	3	2

Table 7 Matrix of comment support

Index	Low	General	High
AF_H	0.41	0.37	0.22
AC_H	0.167	0.333	0.5
AS_H	0.25	0.5	0.25

value correction sub-module, when the number of divided error states n is equal to 8 and β is equal to 4, the initial prediction value is modified, and its value can meet the posterior difference test and small probability test. The error status interval is shown in Table 11.

Table 8 Security situation sequence

T_i	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}
V_i	3504	3485	3285	2919	3582	3306	2921	3070	3321	2926

Table 9 Comparison between the initial predicted value and actual value

T_i	T_1	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}
V_i	3504	3485	3285	2919	3582	3306	2921	3070	3321	2926
V_{i1}	-	3454.7	3469.9	3405.4	3218.3	3329.3	3310.1	3151.2	3089.7	3141.2

Table 10 Error sequence

f_i	f_2	f_3	f_4	f_5	f_6	f_7	f_8	f_9	f_{10}
V_i	-30.3	184.9	486.4	-363.7	23.3	389.1	81.2	-231.3	215.2

Table 11 Error state interval partition

Error state i	1	2	3	4
Error value interval	[-363.7,-257.4)	[-257.4,-151.1)	[-151.1,-44.8)	[-44.8,61.5)
Error state i	5	6	7	8
Error value interval	[61.5,167.8)	[167.8,274.1)	[274.1,380.4)	[380.4,486.4]

Table 12 Comparison of predicted correction value and actual value

T_i	T_2	T_3	T_4	T_5	T_6	T_7	T_8	T_9	T_{10}
V_i	3485	3285	2919	3582	3306	2921	3070	3321	2926
$V_{c(i)}$	3467	3302.1	3142	3443.9	3320.6	3036	3099.7	3239.6	2993.3

quantization process of other quantization periods will not be described here. The sequence of 10 security situations in the 40 min period is shown in Table 8.

5.2 Security Situation Value Prediction and Correction

According to the method in Secs. 4, the security situation value of T_2-T_{10} is predicted. Let the width of the sliding window be $L=10$, and the optimal solution of the static smoothing coefficient α_{best} under the current situation value sequence is 0.126. The initial predicted value and the actual value pair are as shown in Table 9. The value of $V_{11}{}^1$ is calculated to be 3115. The T_2-T_{10} period error sequence is shown in Table 10.

The error interval $[F^L,F^U]$ is equal to [-363.7,486.4]. According to the Sec. 4 situation

The corrected security situation predicted values in the T_2-T_{10} interval are calculated by formulas (14)–(19). The error state initial probability distribution vector is determined by the security situation value of 10 quantization periods before the T_1 period. The comparison between the corrected value of security situation prediction and the actual value is shown in Table 12.

The posterior difference ratio c is equal to 0.42. The value P of small probability test result is equal to 0.89. Therefore, the prediction accuracy of this paper is two-level. The threshold test condition is satisfied. The predicted security situation of T_{11} is $V_{c(11)}=V_{11}{}^1-F'_{(11)}=3115-141.6=2973.4$.

The relative error of the security situation value $V_{11}=2920$ of this period in the original situation sequence is 1.8 %, indicating that the prediction accuracy is high.

The prediction of the security situation values for other periods is the same as the procedure of Section 5.1 and Section 5.2. A total of 80 security situation prediction values are generated.

5.3 Sliding Window Width Selection Experiment

Different sliding window width L will have different influence on the final prediction results. In this paper, the sliding window width is selected to generate a higher precision prediction value sequence before the prediction value is corrected. This can reduce the load of predictive value correction sub module. This paper takes $L=5,10,15$, and the comparison between the predicted value of network security situation and the actual value is shown in Fig. 3.

It is known from Fig. 3 that when the sliding window width $L=10$, the error between the security situation predictive value generated by the network security situation prediction sub-module and the original security situation value is smaller. This is because:

(1) When the width of the sliding window is short, the time span is small, and the forward data has less influence on the prediction of the security situation value than the recent data;
(2) When the width of the sliding window is large and one or several security situation values in the window fluctuate greatly, the value of static smoothing coefficient α will be determined by other data with small fluctuation in the window. It will cause the situation prediction sub-module to reduce the adaptability to abnormal situation fluctuations;
(3) When the width of the sliding window is moderate, the effect of the recent data and the long-term data on the prediction result will be balanced. The difference in the number of situation values with large fluctuations and small fluctuations in the window will be reduced, and the accuracy of the preliminary prediction of the security situation is improved.

Therefore, the width of the sliding window is $L=10$.

5.4 Experimental Comparison and Analysis

The experimental data set is the LL_DOS_1.0 data set. The model of this paper, the traditional Markov prediction model, and the improved Convolutional Neural Network (ICNN) prediction model [8] are used to obtain the network security situation prediction value. The results of the three methods are compared. Through experiments, the network security situation prediction value sequence of three methods (as shown in Fig. 4) and the security situation prediction value absolute error sequence (as shown in Fig. 5) are obtained.

According to Figs. 4 and 5, Compared with other models, the network security situation prediction results obtained by this model have a better fitting degree with the original situation value. The reasons are as follows:

(1) The prediction of traditional Markov model depends on the state transition probability matrix, but the state transition probability matrix lacks dynamic adjustment after being determined. Therefore, the absolute error of the prediction results is large and the error presents a periodic trend;
(2) Although increasing the depth of convolutional neural network can improve the accuracy of security situation prediction to a certain extent, the setting of super parameters in neural network model training is affected by prior

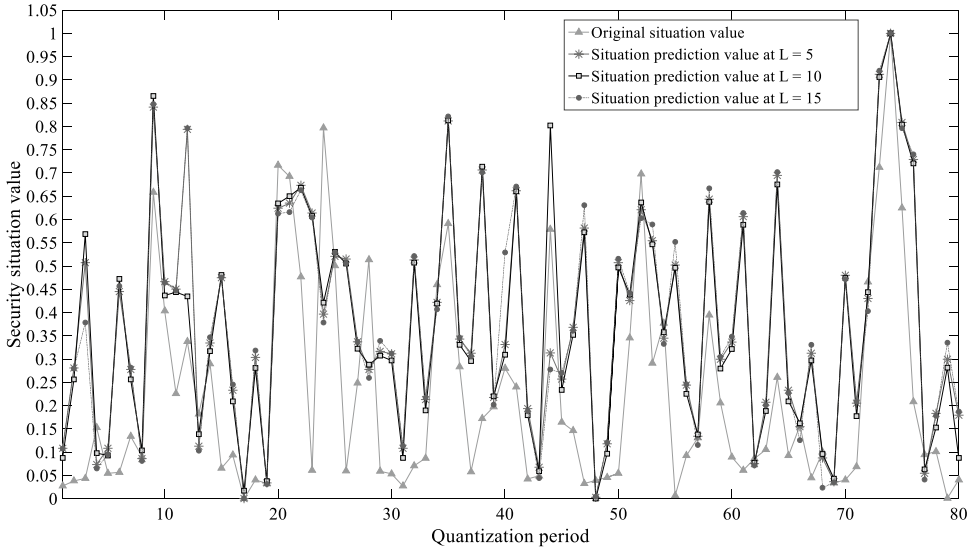

Fig. 3 Comparison of predicted values at different window widths

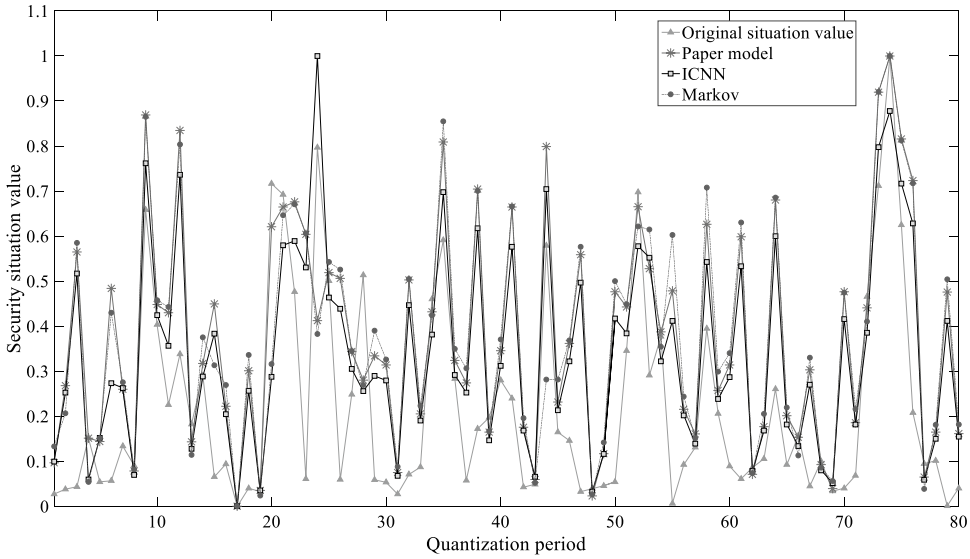

Fig. 4 Network security situation prediction value sequence

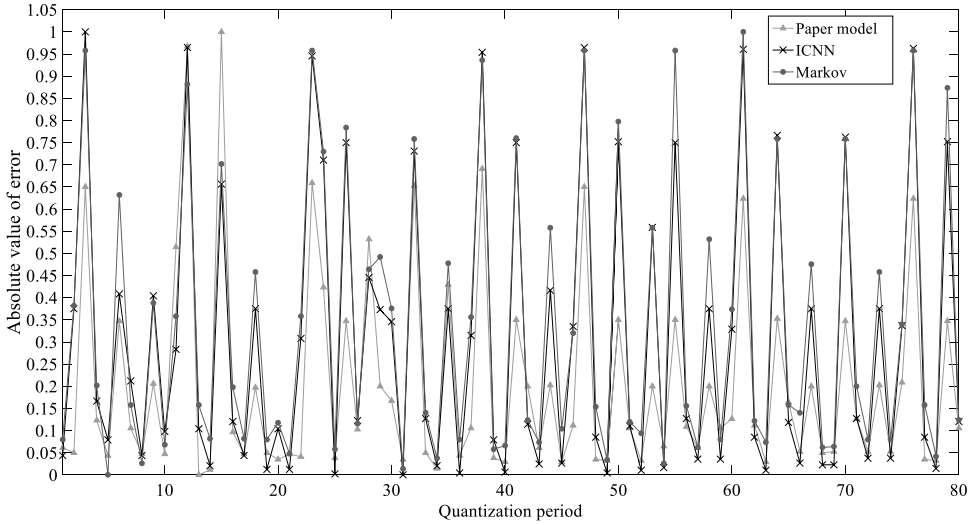

Fig. 5 Predicted value absolute error sequence

experience. This will lead to the deviation between the predicted result and the actual value;

(3) The model uses a sliding window mechanism to fragment a long nonlinear time series. The security situation value in the window is continuously updated so that the correlation coefficient can be adjusted adaptively and, dynamically and the situation prediction value with higher precision is corrected, and the accuracy of the situation prediction is improved.

6 Conclusions

This paper proposes an adaptive IoT network security situation prediction model. The model quantifies the network security situation values of several cycles by entropy correlation method and segments the security situation values arranged in time series based on the sliding window mechanism. The author uses the adaptive cubic exponential smoothing method to generate the initial prediction results. And the author uses the time-varying weighted Markov chain to predict the error and correct the security situation prediction value. Through the network situation prediction of the IoT, we can get the IoT network security situation for a period of time in the future, which will make people take corresponding protection measures for the system in time, and can effectively avoid certain property losses.

Because the relative importance of alert features varies in different network scenarios, future research will focus on the calculation and dynamic adjustment of alert feature weights. In addition, in order to improve the applicability of the prediction model to the abnormal fluctuations of network

security situation values, the linear correlation of network security situation values series will be mainly analyzed. Finally, the higher the trust of the data obtained from the IoT device [16], the more accurate the situation value is predicted by the reliable data. Therefore, future research will also focus on the influence of data trust on the experiment.

Acknowledgements This work was supported by the Civil Aviation Joint Research Fund Project of the National Natural Science Foundation of China under granted number U1833107. We are grateful for the support of this foundation project, as well as for the proofreading and valuable comments provided by all of our co-authors.

Author Contributions Hongyu Yang proposed research ideas and methods. Le Zhang designed experiments, analyzed results and wrote the manuscript. Xugao Zhang conducted theoretical and methodological research. Jiyong Zhang gave suggestions for revision of this paper.

Funding This work was supported by the Civil Aviation Joint Research Fund Project of the National Natural Science Foundation of China under granted number U1833107.

Data Availability The data set used in this study is Lincoln Laboratory's standard dataset LL_ DOS_ 1.0.

Code Availability Not applicable.

Declarations

Conflicts of Interest/Competing Interests The authors declare that there is no conflict of interests regarding the publication of this paper.

References

1. Tang C, WEI X LIUC et al (2020) UAV-enabled social internet of vehicles: roles, security issues and use case. In: 6th International Symposium on Security and Privacy in Social Networks and Big Data, pp 153–163
2. Olivier F, Carlos G, Florent N (2015) New security architecture for IoT network. Procedia Comput Sci 52:1028–1033
3. Leau YB, Manickam S (2015) Network security situation prediction: a review and discussion. information science and applications. Springer, Berlin
4. Zhang SB, Shen YJ, Zhang GD (2018) Network security situation prediction model based on multi-swarm chaotic particle optimization and optimized grey neural network. In: IEEE 9th International Conference on Software Engineering and Service Science, pp 426–429
5. Xiao P, Xian M, Wang HM (2017) Network security situation prediction method based on MEA-BP. In: 3rd International Conference on Computational Intelligence & Communication Technology, pp 1–5
6. Sun SX (2015) The research of the network security situation prediction mechanism based on the complex network. In: International Conference on Computational Intelligence and Communication Networks, pp 1183–1187
7. Zhou XW, Li XL (2018) Multi node network security situation prediction model based on improved G-K algorithm. Sci Technol Eng 18(25):72–77
8. Zhang RC, Liu YC, Liu J et al (2019) Network security situation prediction method using improved convolution neural network. Comput Eng Appl 55(6):86–93
9. Xi RR, Yun XC, Zhang YZ et al (2015) An improved quantitative evaluation method for network security. Chin J Comput 38(4):749–758
10. Debar H, Wespi A (2001) Aggregation and correlation of intrusion-detection alerts. In: International Symposium on Recent Advances in Intrusion Detection, Berlin, pp 85–103
11. Zhao DM, Zhang YQ, Ma JF (2004) Fuzzy risk assessment of entropy-weight coefficient method applied in network security. Comput Eng 30(18):21–23
12. Fu Y, Wu XP, Ye Q et al (2010) An approach for information systems security risk assessment on fuzzy set and entropy-weight. Acta Electron Sin 38(7):1489–1494
13. Huang S, Liu A, Zhang S, et al (2020) BD-VTE: a novel baseline data based verifiable trust evaluation scheme for smart network systems. IEEE Trans Netw Sci Eng. https://doi.org/10.1109/TNSE.2020.3014455
14. Jiang B, Huang G, Wang T et al (2020) Trust based Energy Efficient Data Collection With Unmanned Aerial Vehicle in Edge Network. Trans Emerg Telecommun Technol.. https://doi.org/10.1002/ett.3942
15. Wang X, Qi Y, Li QM (2017) Network anomaly detection model based on time-varying weighted Markov chain. Comput Sci 44(9):136-141 + 161
16. Li T, Liu W, Wang T et al (2020) Trust data collections via vehicles joint with unmanned aerial vehicles in the smart Internet of Things. Trans Emerg Telecommun Technol. https://doi.org/10.1002/ett.3956