In-depth basic data detection device based on Internet of Things technology

Shanyi Xie\(^1\), Ziying Zhang\(^1\), Chen Cheng\(^1\)
†, Jian Wang\(^2\), Chen Lian\(^2\)

\(^1\) Electric Power Research Institute of Electrical Guangdong Power Grid Co., Ltd. Guangzhou, 510080, China
\(^2\) China Southern Power Grid Digital Grid Research Institute Co., Ltd, Guangzhou, 510060, China

Abstract
Due to the limited computing power of the perception layer of the Internet of Things (IoT), the ability to analyse and process the collected complex object information data is insufficient, and it is also difficult to complete the storage of a large amount of collected data. Through convolutional neural network-simple recurrent unit (CNN-SRU) deep learning, we preprocess a large amount of complex data in the perception layer. The data collected by the perception layer are first transmitted to the CNN for simple category screening and analysis, and then they reach the SRU link, which is updated and optimised again, to improve the integrity and accuracy of IoT information collection. The results show that the accuracy of gated recurrent unit (GRU), long–short-term memory (LSTM) and SRU algorithms shows a downward trend under the three error evaluation standards of root mean squared error (RMSE), mean absolute error (MAE) and relative error (RE), from 0.034 to 0.015, 0.028 to 0.012 and 0.024 to 0.013, respectively; in terms of training time, the SRU algorithm is increased by 54.52%; the maximum SRU in terms of data storage is increased to 33.22%; and the maximum SRU reduction in data mining energy consumption is 11.45%. This meets the requirements of IoT applications in big data mining.

Keywords: Internet of Things, perception layer, data mining, SRU, CNN

1 Introduction
The rapid development of the Internet of Things (IoT) benefits from the rapid development and popularisation of the computer Internet \([1, 2]\). The realisation of IoT mainly includes the cooperation of the perception, network and application layers. The perception layer generally perceives related objects through sensors, intelligently recognises objects and then connects sensory sensing devices through ubiquitous network processing to realise intelligent object recognition. The IoT unites all walks of life through Internet technology, so as to realise the interconnection of objects in unlimited time, place, and provides people with a large amount of accurate and reliable information \([3, 4]\). Users can view the information and status of related objects in real time through

\(^{†}\)Corresponding author.
Email address: ttstugzvip@126.com

the IoT client. For the realisation of IoT, first it needs to collect object information through a large number of sensing technologies, and update the information to the network in real time; after the IoT receives the collected information, it is transmitted, and finally the data are processed by some information data and processed into types that meet the user needs [5, 6]. At present, the most widely used information collection in the perception layer is mainly of two types: wireless sensor network (WSN) and radio-frequency identification (RFID) [7, 8] technologies. The network resources of the sensing layer are limited, and it is difficult to guarantee the privacy protection of data due to multi-hop forwarding. The energy of the perception layer is limited, and it is difficult to meet the protection strategy of high traffic volume. Second, the internal nodes of the perception layer have limited memory and computing power, so they cannot store much data, and also it is difficult to process complex data [9]; Therefore, when collecting different object information, the relevant electrical signal frequencies are also different. How to quickly screen and classify the collected information, and transmit the data confidentially after simple processing is of concern. The IoT WSN consists of multiple autonomous wireless nodes or terminals, which cooperate with each other to complete network tasks. Different WSN structures correspond to different privacy security and life cycles. The structure of WSN mainly includes clustering and distributed. As shown in Figure 1, the core of clustering is to divide network nodes into clusters; the nodes in the cluster send data to the cluster head node, and then send it out after a series of fusions. The clustering network can reduce the number of network communications and save energy; in addition, the nodes are ordinary sensors with simple functions, which greatly reduces the communication volume; the clustering type is easy to manage and is suitable for large-scale networks. The distributed structure is relatively simple, all nodes are equal and the entire network has only one level. The distributed structure is simple, the error of a single node has little impact on the entire network, all nodes have equal status, it is not easy to cause bottleneck effects and it has high robustness [10–12].

![WSN clustered and distributed structure diagram](image)

**Fig. 1** WSN clustered and distributed structure diagram. WSN, wireless sensor network

The security data of IoT mainly include static data [13, 14] and dynamic data [15, 16]. Static data (device-related data, vulnerability, security measure data) does not change with time, while dynamic data evolves with time, including traffic data, log data and information data collected by the perception layer. For the changes in static data, statistical methods can be used. Machine learning can be used for data processing and prediction for dynamic data, and abnormal data can be monitored and analysed [17]. There are many methods for abnormal data detection at home and abroad [18, 19]. The most common method is to directly apply the detection of Internet technology to IoT. In the research work of Vijayakumar and Shiny Angel [20], they studied abnormal surveillance access found in IoT smart cities, wherein the monitored access distribution exploits the communication characteristics of user applications and their synchronisation with user devices. Anomalies linking user devices, applications and authentication are observed in backpropagation (BP) learning. BP learns to reduce the assigned weights according to the anomalies trained during the visit assignment process. Kayode Saheed et al. [21] proposed an efficient supervised machine learning intrusion detection system (IDS) to utilise min–max normalisation method for feature scaling to limit information leakage on the test set for detecting IoT attacks with detection accuracy of up to 99.99% and Matthews correlation coefficient (MCC) up to 99.97%.
In-depth basic data detection device based on IoT technology

Compared with other proposed models and state-of-the-art methods, XgBoost provides excellent accuracy, precision, F1-score and MCC. Lin [22] proposed an automatic image algorithm for tourism scene anomalies. On the basis of traditional image segmentation technology, the dynamic characteristics of continuous frames were added according to the neighbourhood correlation characteristics of Markov random fields and reconstruct the Gibbs energy function. The location of the abnormal situation is determined by coding, the spatial information of each pixel and adjacent points is considered, and the time information of consecutive frames is also added to accumulate the energy values of all pixels in the entire image while using the energy values to compare data for analysis. It can accurately and efficiently identify the abnormal situation images of tourist attractions. Chen [23] proposed a multi-objective evolutionary convolutional neural network (CNN) for the IDS. The method uses CNN as a classifier to detect intrusions, and uses a decomposition-based multi-objective evolutionary algorithm (MOEA/D). The algorithm is improved to evolve the CNN model, which greatly simplifies the parameter tuning process. Lu and Chen [24] modelled the risk analysis of IoT through convolution, and established the return matrix of the regulatory risk model. The CNN used will pool the training data to the maximum value, and set the local corresponding normalisation layer, which shortens the mining time and contrasting methods by 6–13 S. From the analysis of software-based IDS, Violettas et al. [25] introduced a new IDS for RPL with multiple profiles to solve the above problems and at least 13 attacks were mitigated. At the same time, other solutions can be up to 8, while the overhead it supports for different modes of operation is lower (i.e. 6.28% on average) compared with other solutions by up to 30%. Power consumption remains at acceptable levels (from 0.18% to over 1.54%).

There are many IoT devices, and their security protection is fragile [9, 26]. All nodes in the current WSN have routing functions, which can route to other nodes by themselves, and send relevant data information to adjacent nodes. The WSN sensor network has five characteristics: limited resources, self-organisation, data-centric, multi-hop routing and security. Nodes have limited computing power, and the deployment and operation of the network does not require any other network infrastructure, and can automatically forward data. Due to the limited computing power of the perception layer, the ability to analyse and process the collected complex object information data is insufficient, and it is difficult to store a large amount of collected data. Therefore, in this paper, we use convolutional neural network-simple recurrent unit (CNN-SRU) deep learning to preprocess a large amount of complex data in the perception layer. The data collected by the perception layer are first transmitted to CNN for simple classification analysis, and then reach the SRU link, to update and optimise the screening again to improve the integrity and accuracy of IoT information collection.

2 In-depth basic data detection based on CNN network application networking technology

The data collection, storage and display of IoT are continuously and deeply excavated. However, there will be abnormal data at the same time, which will reduce the quality of the data, thus affecting the effect of in-depth data mining. In addition, abnormal data may contain potential hazards such as smart home node failures that cause fires and medical equipment data loss. In view of the above problems, this paper adopts CNN to strengthen data mining to reduce abnormal data. Combined with the linear structure of SRU, it can easily and accurately capture the semantic sequence information from characters to text sequences.

2.1 Structure of CNN network

For the different types of data in this paper, we choose a 2D CNN network to track electronic data. 2D CNN can not only extract features, data reconstruction and other preprocessing methods for a large number of images/texts. At the same time, each of its calculation layers has the same weight for sharing links, which greatly reduces the calculation cost and calculation time, and avoids too many parameters affecting the clarity of the design drawings.
CNN mainly includes the convolutional layer, pooling layer, fully connected layer and softmax classification output layer. In the middle, dropout, regularisation and orthogonalisation can be used to improve the network computing performance to prevent the calculation results from overfitting or underfitting the dataset [27, 28]. A schematic diagram of the general structure of the CNN network is shown in Figure 2.

![Fig. 2 Convolutional network structure](image)

### 2.2 Learning algorithm of CNN

The training process of CNN is mainly divided into forward propagation and BP [21]. First, by inputting data to the convolution layer, the feature extraction is performed on the convolution operation based on the filter and convolution kernel (Kernel), the feature map is obtained, the bias term is added to it and then the activation function (RELU, tanh, sigmoid, softmax) to calculate the output of the convolutional layer. The pooling layer samples the data processed by the receiving convolutional layer, and then converts the feature map into a vector by stretching, and sums the weighted biases. Finally, the class probability output is obtained through the activation function, and the calculation is repeated until the loss function is the minimum value.

#### 2.2.1 Calculation of the convolutional layer

The convolution formula is as shown in formula (1), and the total sum obtained by the summation is output through the nonlinear activation function $f$ (the specific process is shown in Figure 3 as

$$a_{d,i,j} = f \left( \sum_{d=0}^{D-1} \sum_{m=0}^{F-1} \sum_{n=0}^{F-1} w_{d,m,n} x_{d,j+m,j+n} + b \right)$$

where $D$ and $F$ are the number of filters and the size of the convolution kernel, respectively; $X$ is the input; $W$ is the weight and $b$ is the bias.

![Fig. 3 Convolution flowchart](image)

#### 2.2.2 Calculation of pooling layer and full connection

The pooling layer retains the main effective information by locally sampling the feature map and reduces the influence of unnecessary data on the calculation result. In addition, as long as there is a relative relationship with the main information of the feature map, no matter how large the proportion of abnormal data is, it cannot affect the accuracy of the results. After the data are passed through the pooling layer, the anomalies can be filtered out, the training accuracy can be improved and the error can be reduced. The average sampling method used in this
paper samples the feature map. In the traversed region, the average value is selected as the new feature of the region.

Due to the characteristics of CNN, under the action of the fully connected layer, the output of the convolutional layer is weighted $w^v$ summed, and then output through the activation function $f$, as shown in formula (2):

$$y = f(w \cdot x + b)$$  \hfill (2)

### 2.2.3 Calculation of softmax output layer

The activation function non-linearises the total number of weighted bias sums to solve multi-class problems, and its calculation formula is as follows:

$$y_k = \frac{e^{a_k}}{\sum_{i=1}^{n} e^{a_i}}$$  \hfill (3)

Here, softmax is used for multi-classification problems.

### 2.2.4 Backpropagation

Formulas (1)–(3) are forward propagation, and the error is calculated by the loss function for reverse propagation. The process is divided into three main parts:

$$\delta^{i,l} = -(y_{real}(i) - y_{predict}(i)) \ast \sigma(a^{i,l})$$  \hfill (4)

(b) The calculation error is passed in the reverse direction. The specific propagation formula of CNN is as follows:

Fully connected layer:

$$\delta^{i,l} = \left(w^{l+1}\right)^T \delta^{i,l+1} \ast \sigma(a^{i,l})$$  \hfill (5)

Convolutional layer:

$$\delta^{i,l} = \delta^{i,l+1} \ast \text{rot} 180 \left(w^{l+1}\right) \ast \sigma(a^{i,l})$$  \hfill (6)

Pooling layer:

$$\delta^{i,l} = \text{upsample} \left(\delta^{i,l+1}\right) \ast \sigma(a^{i,l})$$  \hfill (7)

where $l$ is the current layer and $\sigma$ is $(a^{i,l})$ the activation function.

(c) The final goal of reverse transfer is to update the weight $w$ and bias $b$, and the specific calculation is as follows:

Fully connected layer weight update:

$$w' = w - \alpha \sum_{i=1}^{m} \delta^{i,l} \left(a^{i,l-1}\right)^T$$  \hfill (8)

Fully connected layer bias update:

$$b' = b - \alpha \sum_{i=1}^{m} \delta^{i,l}$$  \hfill (9)
where $w_l$ and $b_l$ are the weight and bias of the $i$-th node of the fully connected layer, respectively.

Convolutional layer weight update:

$$w_l = w_l - \alpha \sum_{i=1}^{m} \alpha^{i,l-1} \delta_{i,l}$$  \hspace{1cm} (10)

Convolutional layer bias update:

$$b_l = b_l - \alpha \sum_{i=1}^{m} \sum_{u,v} \delta_{i,l}^{u,v}$$  \hspace{1cm} (11)

where $\alpha^{i,l-1}$ is the output of the $i$-th neuron in the $l-1$ layer.

In addition, this paper chooses the cross-entropy as the loss function, and the formula is as follows:

$$J_c = -\frac{1}{N} \sum_{i=1}^{k} \sum_{i=1}^{N} y(i) \log (y_c(i))$$  \hspace{1cm} (12)

where $y_c(i)$ is the predicted value, $y(i)$ is the true value and $N$ is the number of samples.

### 2.3 SRU network structure

In this paper, a variant of the RNN network, the SRU algorithm, is used to optimise the CNN network that processes large amounts of data. Compared with the commonly used long–short-term memory (LSTM) and gated recurrent unit (GRU) algorithms, the SRU network has a simpler structure, lower computational complexity and faster data training speed. A schematic diagram of the general structure of the SRU network is shown in Figure 4.

![SRU network structure diagram](image)

**Fig. 4** SRU data input diagram. SRU, simple recurrent unit

SRU is similar in function to LSTM and GRU. Its main purpose is to prevent problems such as gradient explosion or gradient disappearance caused by excessive time series data. It is mainly composed of output gate, input gate and forget gate [29, 30].

### 2.4 Learning algorithm of SRU network

The specific calculation process of the SRU algorithm is as follows
1. First, the input data needs to be weighted and linearised:

$$\tilde{x} = wx$$

(13)

Wherein, $k$ is the input data at time $k$, $w$ is the weight matrix of the input gate and $\tilde{x}$ are the weighted data.

2. Data weighted linearisation is conducted for the forget gate, and it is nonlinearily processed by the activation function:

$$f_k = \sigma (w_f \tilde{x} + b_f)$$

(14)

where $\sigma$ is the activation function, $b_f$ is the bias of the forget gate, $w_f$ is the weight matrix of the forget gate and $f_k$ is the output of the forget gate at time $k$.

3. Use the forget gate to modulate the state parameters inside the SRU to serve the state output obtained by parallel computing:

$$c_k = f_k \odot c_{k-1} + (1 - f_k) \odot \tilde{x}_k$$

(15)

where $c_k$ is the state quantity at time $k$.

4. The calculation method of the output gate is similar to that of the forget gate, and it is further processed nonlinearly as

$$r_k = \sigma (w_r \tilde{x} + b_r)$$

(16)

where $r_k$ is the output gate at time $k$, $w_r$ and $b_r$ are the weight matrix of the output gate and output gate bias, respectively.

5. Combine the operation results of Formulas (1)–(4) to update the output state quantity at time $k$, and pass it to the SRU layer at time $k+1$ for operation as

$$h_k = r_k \odot \text{ReLU} (c_k) + (1 + r_k) \odot x_k$$

(17)

where ReLU is one of the activation functions and $h_k$ is the state output at time $k$.

### 2.5 Data preprocessing and model hyperparameter determination

In this paper, the training length of this model algorithm is about $N = 168,360$ data points. In addition, because different communication methods correspond to different frequencies, in order to avoid excessive data influence errors caused by different orders of magnitude, this paper normalises the data. The purpose of processing is to normalise all data to the same interval and speed-up the solution of gradient descent. Therefore, we choose to select max–min normalisation for processing, and the calculation method is as follows:

$$\hat{x} = \frac{x - x_{\text{min}}}{x_{\text{max}} - x_{\text{min}}}$$

(18)

where $x_{\text{max}}$ is the maximum value in the dataset and $x_{\text{min}}$ is the minimum value in the dataset. After normalisation, the network training error is reduced.

In addition, in order to balance the training cost and training accuracy, the SRU network in this paper adopts a three-layer hidden layer structure, and the number of nodes in each layer is 64; the activation function is ReLU; and the number of training iterations epoch is set to 250; The learning rate is 0.0011; and the root mean square MSE function is used as the evaluation index; Adam is the network training optimiser.
3 Experimental verification and comparative analysis

3.1 Comparative analysis of abnormal data detection accuracy

In order to verify that the SRU network has better performance, this paper compares and analyses GRU and LSTM. To further analyse the performance of these three networks, we choose the following three evaluation metrics:

Root mean squared error (RMSE): a measure of the error between the observed value and the actual value, which is calculated as follows:

$$RMSE(X, h) = \sqrt{\frac{1}{N} \sum_{i=1}^{N} (h(x_i) - y_i)^2}$$  \hspace{2cm} (19)

Mean absolute error (MAE): the average value of the absolute error, which can better reflect the actual situation of the predicted value error, which is calculated as follows:

$$MAE(X, h) = \frac{1}{N} \sum_{i=1}^{N} |h(x_i) - y_i|$$  \hspace{2cm} (20)

Relative error (RE): the ratio of the difference between the observed value and the actual value to the actual value, which is calculated as follows:

$$RE(X, h) = \frac{h(x_i) - y_i}{y_i}$$  \hspace{2cm} (21)

Figure 5 gives the prediction evaluation index of the translation results of the three networks.

Specifically, compared with the actual data received and the compiled data at the frequency, the RMSE of the SRU algorithm is smaller than that of the GRU and LSTM algorithms, which are 0.015, 0.034 and 0.024, respectively; the SRU algorithm is improved by 56% and 37%, respectively; MAE by 0.012, 0.028 and 0.014, respectively; the SRU algorithm by 57% and 14%, respectively; the RE was 0.013, 0.032 and 0.024, respectively; and the SRU algorithm increased by 59% and 37%, respectively.

3.2 Comparative analysis of algorithm training speed

Algorithm training speed is also one of the criteria for neural network evaluation. Because in practical applications, the network should not only pay attention to the training ability of input and output, but also
consider the cost of implementing the operation. Therefore, this paper continues to compare the three algorithms of SRU, GRU and LSTM to test their speed performance, set the same hyperparameters as the number of iterations, mini-batch and so on.

The training time of SRU, GRU and LSTM under the condition of the same amount of data is within 80–85, 152–157 and 180–185 minutes, respectively. It can be seen that the SRU algorithm has a huge advantage over the other two algorithms compared with the GRU algorithm, and LSTM algorithms reduce by 46.28% and 54.52%, respectively, as shown in Figure 6. This has huge potential value for future IoT data mining applications.

![Fig. 6 Comparison of network model training time](image)

### 3.3 Comparative analysis of data storage capacity

The amount of data storage is one of the criteria for judging IoT technologies. Because in practical applications, a large amount of data need to be analysed in IoT, how to store the maximum amount of data and analyse it is particularly important. Therefore, this paper continues to compare the three algorithms of SRU, GRU and LSTM to test their data storage capacity, set the same hyperparameters as the number of iterations, mini-batch and so on.

For the data storage capacity of SRU, GRU and LSTM under the condition of the same amount of data, it can be seen that the SRU algorithm has a huge advantage over the other two algorithms, as shown in Table 1.

<table>
<thead>
<tr>
<th></th>
<th>Storage volume</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN-SRU</td>
<td>18,045,724</td>
</tr>
<tr>
<td>CNN-GRU</td>
<td>14,577,537</td>
</tr>
<tr>
<td>CNN-LSTM</td>
<td>13,545,767</td>
</tr>
</tbody>
</table>

<p>| | |</p>
<table>
<thead>
<tr>
<th></th>
<th></th>
</tr>
</thead>
</table>

### 3.4 Comparative analysis of data mining energy consumption

In practical applications, the energy consumption required for data mining has a considerable relationship with the economic cost. Therefore, this paper continues to compare the three algorithms of SRU, GRU and
LSTM to test their speed performance and set the same hyperparameters as the number of iterations, mini-batch and so on.

The energy consumption of SRU, GRU and LSTM under the condition of the same amount of data shows that the SRU algorithm has a huge advantage over the other two algorithms, as shown in Figure 7.

![Figure 7: Data mining energy consumption comparison](image)

**Fig. 7** Data mining energy consumption comparison

### 4 Conclusion

In this paper, we give a basic description of the concept of multi-communication framework and neural network algorithm, and introduce the structure and calculation process of SRU algorithm for wireless communication receiving module. We compared the three algorithms of SRU, GRU and LSTM, and found the hidden layer, number of nodes, number of iterations, learning rate and activation function in the network structure to be all the same. The results show the following:

1. The accuracy of GRU, LSTM and SRU algorithms shows a downward trend under the three error evaluation standards of RMSE, MAE and ME, from 0.034 to 0.015, 0.028 to 0.012 and 0.024 to 0.013, respectively.
2. The training time of GRU, LSTM and SRU algorithms is up to 54.52%, indicating the excellent performance of SRU deep learning in IoT data mining.
3. The GRU, LSTM and SRU algorithms have a maximum increase of 33.22% in data storage, indicating that SRU deep learning is not prone to jams and other phenomena in IoT data mining.
4. The GRU, LSTM and SRU algorithms can reduce the energy consumption of data mining by 11.45%, indicating that SRU deep learning brings huge economic benefits to IoT data mining.

**Topic information.** Research on security protection and detection technology of global Internet of Things. Subject Number: GDKJXM20201931
References


