ORIGINAL RESEARCH PAPER

A simple gated recurrent network for detection of power quality disturbances

Xiangrong Zu  |  Kai Wei

Correspondence
Kai Wei, School of Control and Computer Engineering, North China Electric Power University, Beijing, 102206, China.
Email: weikaiv@outlook.com

Abstract
This paper presents a new concise deep learning–based sequence model to detect the power quality disturbances (PQD), which only uses original signals and does not require pre-processing and complex artificial feature extraction process. A simple gated recurrent network (SGRN) with a new recurrent cell structure is developed, which consists of only two gates: forget gate and input gate, and two weight matrices. Compared with the standard Recurrent Neural Network (RNN) model, the training process of the proposed method is more stable and the prediction accuracy is higher. In addition, this special structure retains basic non-linearity and long-term memory, while enabling the simple gated recurrent network model to be superior to Long Short-Term Memory (LSTM) Network and Gated Recurrent Unit (GRU) Network in terms of the number of parameters (i.e. memory cost) and detection speed. In the light of the experimental results, the simple gated recurrent network algorithm can achieve 99.07% detection accuracy, and contains only 18,959 parameters, which indicates that our proposed method is easier to deploy in resource-constrained internet of things (IoT) micro-controllers.

INTRODUCTION

Power quality analysis is currently at the core of energy management in power plants and utilities. Continuous and stable power supply can assist businesses in reducing expenses and increasing operational efficiency. Power quality disturbances (PQD) may cause many problems such as instrument metering errors, motor stalls, device overheating, and protection failures, which can damage power electronic equipment and reduce the service life of components. This will not only cause harm to the equipment itself, but also interfere with communications and even cause a large-scale power outage, which will seriously affect the normal work of the power distribution system [1]. With the steady development of smart grid, energy interconnection, and integrated energy systems, the power supply and load conditions of power systems have changed. Large-scale wind power and photovoltaic energy integrate into the power grid, and a great number of non-linear power components and non-linear electronic loads are connected to the smart grid. These phenomena make the power quality events more and more frequent and complicated, which has attracted the interests of many scholars and research groups in recent years.

Monitoring the PQD signals and accurately identifying the type of disturbances can facilitate the formulation of subsequent effective measures, thereby reducing the negative impact of PQD. Traditional detection schemes usually combine signal processing technologies (SPT) and machine learning (ML) algorithms. As shown in Figure 1, the disturbance signals are input to the signal processing module to analyse them in time and frequency domains by signal processing methods [2, 3] such as Fourier transform (FT), wavelet transform (WT), Hilbert transform (HT), and S transform (ST). Then features are selected from the feature set extracted by these technologies, such as energy, entropy, root mean square, total harmonic distortion, magnitude of fundamental, magnitude of harmonic [4, 5]. Finally, the selected features are transmitted to the classification module for classification and recognition. The classifiers often use ML algorithms [6–8] such as decision trees (DT), support vector machines (SVM), artificial neural networks (ANN), and extreme learning machine (ELM).

Nevertheless, there are some inherent disadvantages of the transformation methods. For example, FT is not suitable for non-stationary signals, the performance of WT depends on the choice of mother wavelet, and the calculation of ST is complicated. In addition, the effect of PQD classification heavily depends on the pre-designed feature set based on expert knowledge.

This is an open access article under the terms of the Creative Commons Attribution License, which permits use, distribution and reproduction in any medium, provided the original work is properly cited.

© 2020 The Authors. IET Generation, Transmission & Distribution published by John Wiley & Sons Ltd on behalf of The Institution of Engineering and Technology
Recently, a lot of new methods also have been presented. The work in [9] uses an improved double-resolution ST to extract features, which reduces the computational complexity of ST while obtaining better time-frequency localization, but it is difficult to find the best value of parameters. In [10], a three-layer Bayesian network for PQD detection is proposed. This method adds historical information and circumstance factors on the basis of existing feature extraction methods, which expands the feature set and improves the accuracy of combination disturbances, but the calculation is complicated. In [11], a new method for classification using the information provided by a phasor measurement unit (PMU) scheme and the homogeneity index is proposed. Only if-then-else is used for classification, which greatly reduces the computational burden. However, the classification effect of multiple compound disturbances cannot be determined.

As an end-to-end classification method, deep learning provides a new idea for simplifying PQD detection and avoiding empirical feature extraction. The work in [12] takes advantage of the superior performance of convolutional neural network (CNN) in the field of image classification. The signals are converted to image format by Wigner–Ville distribution, and given as inputs to CNN for automatic classification. Due to the typical temporal features of PQD signals, some scholars use deep learning algorithms, which are more suitable for sequences, to detect them. A hybrid architecture of CNN and LSTM is put forward in [13, 14] to automatically extract features and detect PQD signals. Document [15] uses a sequence-to-sequence deep learning model for PQD detection. At the same time, time positioning is performed based on the moment when the type of disturbance is changed. However, these network models often have a lot of parameters and require a lot of calculation and storage costs.

In the actual power system, with the new development of the internet of things (IoT) platform and the deployment of large number of edge devices, the need to collect data on the edge devices of IoT and make prediction at the edge becomes more urgent [16]. This requires us to develop methods with fewer parameters and a lower computational cost to apply to resource-constrained devices [17, 18]. It is obvious that PQD signals exactly have time series characteristics, and as a special type of deep learning model, recurrent network contains a hidden layer to process time series data. However, many recurrent network models with gating mechanisms such as LSTM, GRU are computationally complex, and are not suitable for resource-constrained devices. Therefore, this paper aims to propose a simple gated recurrent network (SGRN) with smaller model size and less calculation burden for PQD detection.

We mainly study a new deep learning method SGRN to automatically detecting PQD signals. A simplified gated recurrent cell is designed by reusing the RNN matrices and does not add additional scalar parameters, which can approach the naive RNN model in terms of the number of parameters and calculation cost. At the same time, the long-range dependence of the LSTM/GRU model and its gate structure are retained by forget gate and input gate. This makes our proposed deep learning model more accurate than traditional RNN and is small enough to provide the possibility of deployment on micro-controllers with limited resources.

The main contributions of this research are:

(i) The method adopted here does not require expert knowledge or prior in-depth understanding of PQD features, and avoids complex feature extraction or insufficient extracted features.

(ii) The proposed method adds two gates on the basis of standard RNN, and preserves the long-term memory through this structure. The problem of disappearing gradient of RNN is solved, the training process is more stable, and the prediction accuracy is also improved.

(iii) The SGRN has developed a simplified gated recurring cell, which contains only two weight matrices, and the parameters of the model is less than half of LSTM and GRU. This allows it to match the state-of-the-art prediction accuracy of some reported methods, while having smaller memory cost and computing time.

(iv) We propose a feasible edge-to-centre power quality monitoring system, which combines deep learning and edge computing, in Section 4.4. It makes it possible to realize edge detection by using the micro-controllers of power quality sensors at the intelligent IoT edge and reduce the transmission pressure and the transmission time by only transmitting the data related to disturbances.

2 | DEEP LEARNING–BASED TRADITIONAL SEQUENCE MODELS

The detailed structures of the recurrent neural networks relevant to our research are analyzed in this section.

2.1 | Notation

The given sequence $X = [x_1, x_2, ..., x_T]$ is used to refer to the input sequence, where $x_t \in \mathbb{R}^{D}$ denotes the input for the
2.2 Background methods

RNN is a type of neural network for processing time series tasks. It can correlate the information of the previous and subsequent sequences. The hidden state of its recurrent cell is as follows:

\[ h_t = \text{tanh}(W x_t + U h_{t-1} + b), \]

where \( x_t \) is the current input and \( h_{t-1} \) is the hidden state of the previous time step. \( W \) and \( U \) are parameters that need to be optimized and trained by stochastic gradient descent or other methods. \( \text{tanh}(\cdot) \) as a non-linear activation function maps the output of the hidden state between \(-1 \) and \( 1 \).

Some scholars have proposed LSTM algorithm to improve RNN [19]. It adds a complex gated structure as shown in Figure 2, thus making the training process of the network more stable. This method can maintain long-term and short-term memory, but to a certain extent, it increases the prediction accuracy at the expense of computing costs. LSTM contains cell state \( c_t \) and hidden state \( h_t \), and its calculation method is:

\[
\begin{align*}
    f_t &= \sigma(W_f x_t + U_f h_{t-1} + b_f), \\
    i_t &= \sigma(W_i x_t + U_i h_{t-1} + b_i), \\
    \tilde{c}_t &= \text{tanh}(W_c x_t + U_c h_{t-1} + b_c), \\
    c_t &= f_t \odot c_{t-1} + i_t \odot \tilde{c}_t, \\
    o_t &= \sigma(W_o x_t + U_o h_{t-1} + b_o), \\
    h_t &= o_t \odot \text{tanh}(c_t),
\end{align*}
\]

where \( f_t \) is the forget gate, determining how much content that relevant to the previous hidden state to be preserved. \( i_t, \tilde{c}_t \) make up the input gate and determine what current input information needs to be added. \( o_t \) constitutes the output gate, which determines the output of the current hidden state. The cell state and hidden state of LSTM respectively remain the long term and short term memory of the input sequence. It should be noted that the amount of model parameters of LSTM is four times that of RNN.

GRU is a type of further simplification of LSTM [20]. It discards the cell state of LSTM and reduces its recurrent cell to two gates (see Figure 2). The calculation formula is:

\[
\begin{align*}
    z_t &= \sigma(W_z x_t + U_z h_{t-1} + b_z), \\
    r_t &= \sigma(W_r x_t + U_r h_{t-1} + b_r), \\
    \tilde{h}_t &= \text{tanh}(W_{\tilde{h}} x_t + U_{\tilde{h}}(r_t \odot h_{t-1})), \\
    h_t &= z_t \odot h_{t-1} + (1 - z_t) \odot \tilde{h}_t,
\end{align*}
\]

where \( r_t \) is the reset gate and \( z_t \) is the update gate, both of that update the hidden state from the historical record and the current input, respectively. The amount of model parameters of GRU is reduced compared to LSTM and is three times that of RNN.

3 THE PROPOSED SGRN METHOD

If the detection model is directly integrated into the power quality sensor, the device can complete the self-detection at the edge. Our goal is to design a more concise deep learning model on the basis of ensuring accuracy and make it easier to be deployed on IoT micro-controllers with limited resources, which are so small that they are not suitable for storing other complex recurrent network models. Taking the standard RNN, LSTM, and GRU models as the benchmark, and combining the
3.1 | Detail schematic of SGRN cell

In order to improve the training stability and detection accuracy of the model, based on the standard RNN, this paper designs a new recurrent cell with two gated structures: forget gate and input gate. Meanwhile, in order to reduce unnecessary matrix parameter operations in LSTM and GRU, we let the input components of the two gate structures share two identical matrices \( W \) and \( U \). The calculation process of the recurrent cell is as follows:

\[
\begin{align*}
    f_t &= \sigma(Wx_t + Ub_{t-1} + b), \\
    i_t &= 1 - f_t, \\
    h_t &= \tanh(i_t \circ Wx_t + f_t \circ b_{t-1}).
\end{align*}
\]

In this formula, the current hidden state is generated by directly weighting the current input and the previous hidden state, which can help us obtain a better gradient to make the training process more stable, and we also retain the non-linearity of the standard RNN. The new recurrent cell we designed, as shown in Figure 3, consists of forget gate and input gate. \( f_t \) is the forget gate, which determines the historical information related to the previous state that needs to be retained or forgotten through the \( \sigma(\cdot) \) function. \( i_t \) is an input gate, which is used to control the added information from the current input state. The non-linear activation function \( \sigma(\cdot) \) selected in SGRN is the sigmoid function, and the sigmoid function plays the role of control information in the recurrent cell. The value range of the sigmoid function is between 0 and 1, thus controlling the opening and closing of the door, which will cause the information to be forgotten or remembered. In addition, the temporality of the model is reflected in the information that can be stored and accessed for a long time.

**3.1.1 | Analysis of SGRN structure**

In the design process, we get inspiration from the ATR structure proposed in [21], and add a mechanism similar to self-attention to the proposed recurrent cell. We will show the analysis of SGRN by decomposing the recurrent structure, which can be explained by expanding Equation (4). The expansion results are as follows:

\[
\begin{align*}
    \bar{h}_t &= i_t \circ Wx_t + f_t \circ b_{t-1} \\
    &= i_t \circ Wx_t + f_t \circ i_{t-1} \circ Wx_{t-1} \\
    &\quad + f_t \circ f_{t-1} \circ b_{t-2}, \\
    &= i_t \circ Wx_t + \sum_{k=1}^{t-1} i_k \circ \left( \prod_{l=1}^{t-k} f_{k+l} \right) \circ Wx_k, \\
    \approx \sum_{k=1}^{t} m_k \circ Wx_k,
\end{align*}
\]

where the \( t \)-th hidden state \( h_t \) is related to the first to \( t \)-th input \( x_1, x_2, \ldots, x_t \). This not only makes each hidden state related to the weighted sum of the input components, but also establishes the necessary dependency between each step of input and the current hidden state. \( m_k \) can be regarded as the approximate weight assigned to the \( k \)-th input when updating the hidden state. This form of SGRN that repeated manipulation of the weight assigned to each step information is similar to the self-attention network structure. But unlike the self-attention network, the weights in SGRN span all channels and are unnormalized.

In addition, we can also analyse the memory in SGRN through the expansion process shown in Equation (5). In the input gate, \( i_t \) is used to control SGRN to access the \( t \)-step input \( x_t \), and its value reflects the amount of information allowed to be passed by the current step input. When the amount of information that the input gate allows to transfer is larger, the input signal will be stronger, and the proportion of short-term memory will be larger. At this time, the historical memory information controlled by the forget gate \( f_t \) will decrease. Then, the input information is gradually forgotten through the transmission of the forgetting chain \( \prod_{k=1}^{t-k} f_{k+l} \), and thus becomes a long-term memory, which is gradually replaced by a new input.

**3.1.2 | Analysis of gradient**

The gated recurrent cell of SGRN gives it the ability to handle long-term memory as well as gradient vanishing and explosion. From [22] we can understand that the gradient of the \( t \)-th time step in the RNN model largely depends on Equation (6):

\[
\frac{\partial h_t}{\partial h_{t-1}} = U^T \frac{\partial}{\partial h} g_t,
\]

where \( g_t \) respects the derivative of the Tanh function calculated element-wise. \( \frac{\partial h_t}{\partial h_{t-1}} \) is in the form of a Jacobian matrix, and due

---

**FIGURE 3** Detail schematic of the proposed simple gated recurrent network cell
to the chain rule, it will be multiplied repeatedly as the time step increases. Therefore, when the norm of the weight matrix $U$ is too large or small, this repeated multiplication will cause the gradient explosion or vanishing.

SGRN replaces the single weight matrix $U$ in Equation (6) by the underlined part in Equation (7):

$$\frac{\partial h_t}{\partial h_{t-1}} = (f_t + h_{t-1} \odot \sigma_f + \sigma_i \odot Wx_t)g',$$

where $\sigma_f$ and $\sigma_i$, respectively, denote the derivation of $f_t$ and $i_t$ in Equation (4). Compared with Equation (6), this equation adds a weighted structure of the previous hidden state and the current input in the underlined part, which becomes a multi-addition form. And the norm of this part is related to the current input and varies dynamically along different positions of the sequence, thus providing SGRN with the ability to avoid the explosion and vanishing of the gradient.

3.1.3 Comparison with other recurrent cells in terms of the number of weight matrices and parameters

In the actual training and testing processes, the number of weight matrices and the parameters of the model will affect the running speed and the amount of space occupied. In this regard, we made a systematic comparison of standard RNN, LSTM, GRU, and the methods mentioned in the paper, and summarized in Table 1.

RNN does not contain the gating structure, only has two weight matrices, and the number of its parameter is $n(n + d + 1)$. LSTM, GRU, and SGRN all contain gating structures. LSTM contains four different neural network layers with eight weight matrices, and the parameter quantity is $4[n(n + d + 1)]$. GRU simplifies LSTM, contains three neural layers with six weight matrices, and the parameter quantity is $3[n(n + d + 1)]$. SGRN only contains two weight matrices, and the parameter quantity is $n(n + d + 1)$. Among them, $n$ is the number of hidden units, and $d$ is the input dimension. Therefore, when the hyper-parameter settings are the same, the weight matrix and the number of parameters of SGRN are the least, and the running time and memory cost are smaller.

3.2 Structure of the proposed model

Figure 4 demonstrates how to build a model to implement PQD detection through the proposed recurrent cell. First, the original signals need to be sampled, labelled, and batch processed. $X = [X_1, X_2, ..., X_n]$ is the input data set of the model, $n$ represents how many batches the data set is divided into, and the information of each $X_i$ contains [batch size, time steps, input dimension]. Then, input a batch of data $X_i$ into the SGRN layer. As shown in Figure 4, we use one sequence in the batch as an example to illustrate the data processing flow. In the SGRN layer, we can obtain the output features $[o_1, o_2, ..., o_k]$ that retain long- and short-term memory and put them into the fully connected layer. The fully connected layer maps an input vector of length $k$ (i.e. the output of the SGRN hidden layer) to an output $\zeta = [z_1, z_2, ..., z_m]$ of length $m$, where $z_j$ is equivalent to

### Table 1

Comparison with RNN, LSTM, GRU in terms of the number of weight matrices (WM) and parameters (Param)

| Model  | RNN  | LSTM | GRU  | SGRN |
|--------|------|------|------|------|
| WM     | 2    | 8    | 6    | 2    |
| Param  | $n(n + d + 1)$ | $4[n(n + d + 1)]$ | $3[n(n + d + 1)]$ | $n(n + d + 1)$ |

FIGURE 4 The proposed signal sequence detection model
The score of the $j$-th category, and $m$ is the total number of categories. Then, use Softmax to map the output of the fully connected layer to an output vector $\mathbf{y} = [y_1, \ldots, y_m]$ composed of $m$ probabilities, where $y_j$ is the probability of being classified into the $j$-th category, its value is between 0 and 1, and $\sum_{j=1}^{m} y_j = 1$. The calculation process of Softmax function is shown in Equation (8). And, we use the Argmax function (see Equation 9) to calculate the index of the maximum value in the vector $\mathbf{y} = [y_1, \ldots, y_m]$. The result is the final prediction category, that is, the probability that the current input belongs to this category is the highest.

$$y_j = \text{Softmax}(z_j) = \frac{e^{z_j}}{\sum_{j=1}^{m} e^{z_j}}.$$ \hfill (8)

$$\text{Argmax}(\mathbf{y}) = \{ j \mid \forall i \in (1, \ldots, m) : y_j \geq y_i \}.$$ \hfill (9)

In order to verify the classification ability of the features automatically extracted by the SGRN hidden layer, we perform dimensionality reduction and visualization on the features of different disturbances. We select 15 types of disturbances, and 100 samples for each type of disturbances. The dimensionality reduction visualization is performed by T-SNE algorithm. Figure 5(a) is the result of visualizing the original data, and Figure 5(b) is the result of visualizing the characteristics of the fully connected layer.

The 15 different coloured clusters in the picture represent 15 different power quality signals. It can be clearly seen from the figure that compared with the original disturbance data, the features automatically extracted by the SGRN model can well separate 15 types of disturbances.

### Table 2: Classes of PQD events

| PQD event                                      | Class label |
|-----------------------------------------------|-------------|
| Normal                                        | C1          |
| Sag                                           | C2          |
| Swell                                         | C3          |
| Interruption                                  | C4          |
| Transient/impulsive/spike                     | C5          |
| Oscillatory transient                         | C6          |
| Harmonic                                      | C7          |
| Flicker                                       | C8          |
| Notch                                         | C9          |
| Flicker with sag                               | C10         |
| Flicker with swell                             | C11         |
| Sag with oscillatory transient                 | C12         |
| Swell with oscillatory transient               | C13         |
| Harmonic with sag with oscillatory transient   | C14         |
| Harmonic with swell with oscillatory transient | C15         |

### 4 | EXPERIMENTS

#### 4.1 | Data set description

In order to test the effectiveness of the proposed method, we consider the normal signals, and 14 types of single and compound disturbances, as given in Table 2. According to the mathematical models given in [23] and [24], we generate a total of 12,000 samples through MATLAB R2017b, and the corresponding variations in the parameters are compliant with the IEEE 1159 standards. In addition, in order to make the test effect more reliable, all parameters of the numerical model are allowed to change randomly when we simulate the signals. The PQD signals we simulated have a basic frequency of 50 Hz. We set the sampling frequency to 6.4 kHz and set the simulation period to 10 cycles. Each cycle has 128 sampling points and a total of 1280 sampling points. We use 10,500 of the generated samples for training and 1500 for testing.

#### 4.2 | Hyper-parameter settings

Deep learning models usually need appropriate parameter selection to get satisfactory results. The input of the proposed model contains three parameters: batch size, time steps, and input dim.
We sample the single-phase voltage, and each sample contains a total of 1280 sampling points, so the dimension of the input data should be \([\text{batch size}, 1280, 1]\). Considering that too long time steps will produce higher time cost, we adopt the method of splitting the long samples into sub-sequences. We comprehensively consider the running time and accuracy rate, split the 1280×1 input sequence and reshape it into 320×4, that is, 4 input dim and 320 time steps, which is regarded as the best choice. For the batch size, we set its size to 32, 64, 96 and 128, respectively. Figure 6 shows the change in classification accuracy of different batch sizes. According to the results shown in the figure, selecting the batch size of 64 can get the highest accuracy.

For the number of hidden units, we set it to 64, 96, and 128, respectively, and test the classification accuracy as shown in Figure 7. The figure shows that when the number of hidden units is 128, the accuracy rate curve is more stable and the maximum classification accuracy can be obtained. We use the decay method to set the learning rate. Set a higher learning rate in the initial stage to make the loss drop faster, and then decay the learning rate to avoid falling into the local optimum. Through experimental verification, we set the initial learning rate to 0.005 and decayed by a factor of 0.8 times every 200 epochs. We choose the Adam optimizer to optimize the network. The total training epoch is set to 600, and achieve the highest accuracy at the 570 epoch.

TensorFlow framework is used for our program development. All experiments are implemented in Python environment on i7-10750H 2.6GHz CPU with 16 GB memory.

### 4.3 Results and discussion

Table 3 summarizes the detection accuracy of each type of PQD. In this table, the normal, sag, swell, interruption, transient/impulsive/spike, oscillatory transient, harmonic, flicker, notch, flicker with sag, flicker with swell, sag with oscillatory transient, swell with oscillatory transient, harmonic with sag with oscillatory transient and harmonic with swell with oscillatory transient signals are, respectively, indicated by C1 to C15. It can be observed from this table that most PQD signals have been correctly identified, and the detection rate of each kind of disturbance is above 96%, and the average accuracy of all PQD signals is 99.07%. However, some PQD signals will also be recognized incorrectly. C10 is the most likely to be misidentified, followed by C4 and C14. When the amplitude is close to 0.1 pu, C2 and C4 are relatively similar, so the disturbances containing the sag component is easily confused with C4. When the parameters of C5 and C9 are set to be small, it is easier to confuse them with the normal signals in the recognition process, and this phenomenon is easy to be ignored [25, 26].

### 4.4 Comparison with other methods

In order to further illustrate the effectiveness of the algorithm proposed here, we compare the proposed method with the traditional approaches that are composed of SPT and ML and some methods based on deep learning.

In Table 4, we enumerate the performance comparison of our proposed method with some reported methods in the number of detected PQD types, the number of feature selections, the classification accuracy, and the classification time. As can be seen from the table, some methods have higher accuracy, but they usually need to select more features as support [5, 28, 30] or can only detect fewer disturbance types [9, 12]. Such as [28], the accuracy rate is as high as 99.09%, but 45 features need to be designed as the input of the classifier, which undoubtedly greatly increases the calculation cost. The method used in [12] has an accuracy rate of 99.67%, but it can only identify nine types of disturbances. Our proposed method completely avoids artificial feature selection and feature optimization, which greatly reduces manual participation and computational burden, while being able to detect 15 types of single and compound disturbances.

In the comparison between SGRN and some deep learning models that automatically extract features, it can be seen that the detection effect of the method proposed here is better than the LSTM, GRU, and CNN + LSTM implemented in [13] and the FTSI + CNN method proposed in [31]. Although the
accuracy rate is lower than [12], it can detect more compound disturbances.

We also summarized some articles that mentioned the test time, such as [4, 6 9, 27 29, 30]. Although the test times in some of these articles are shorter than the test time reported here, it cannot be ignored that these test times are often affected by the factors such as the number of sample points, the number of disturbances in the test set, and the processor speed. Our proposed method requires 0.2104 s to test the entire test set (1,500 samples), that is, only 0.00014 s to detect one sample. The sampling time of each sample is 200 ms, so we can complete the detection before the arrival of the next set of sampling.
### 4.5 Analysis of practical data test

In order to verify the detection effect of the proposed method in actual situation, we use two groups of practical data for testing. The first group of verification data is the “Real-life power quality sags” data set provided by IEEE DataPort [32]. The data set provides signals recordings from the power network of the University of Cádiz during the last 5 years, including 26 different voltage sag events. There is noise interference in the collection process of this data set, and the sampling frequency is 20 kHz, which is different from the samples simulated according to the mathematical model. Therefore, before the verification experiment, we must filter out the noise in the real-life signals. After that, the samples are resampled to ensure that the number of sampling points per cycle is 128. In addition, since the sampling period of the data set is relatively long, in order to be consistent with the simulated signals, the real signals are truncated in units of 10 cycles (1280 sampling points). Finally, the data is normalized to $[-1, 1]$ and input them to the trained model for testing. The test result is shown in Table 5.

The second group of data is provided by the laboratory of the School of Electrical Engineering, North China Electric Power University, and the disturbance signals are generated by chroma 61830. Chroma 61830 is a recycled grid analog power supply designed according to the IEEE 1547 / IEC 61000-3-15 / IEC 62116 standard. It is a powerful equipment which can simulate a variety of abnormal and disturbance states of the power grid to meet the test requirements. The laboratory uses chroma 61830 to generate six different types of disturbance signals, the basic frequency of the signals is 50 Hz, and the sampling frequency is 10 kHz. Before the verification experiment, we re-sample and truncate the signals. Then, we normalize the data to $[-1, 1]$, and mark its true label. Finally, the labelled samples are input into the trained model for testing. The test result is shown in Table 6.

Table 5 and Table 6 show the detection effect of the proposed algorithm on two groups of practical data. It can be seen that the recognition accuracy of sags in the first group is 92.31%, and the average accuracy of the six types of disturbances in the second group is 96.25%. Due to the noises in the real-life data, the practical data detection accuracy is not as good as the simulation data. But it can be seen that our proposed method still has a good performance under practical PQD signals.

### 4.6 A feasible edge-to-centre power quality monitoring system

At present, IoT platforms and various edge IoT devices are being widely used in many fields. Some fault detection systems combining edge computing and deep learning and some biological sensor that can monitor the medication process of patients have been proposed. With the development of smart grids and smart cities, micro-controllers and signal processing systems are gradually popularizing in the power sector, aiming at improving the quality of people’s production and life. In order to adapt to the new development environment, we have designed a feasible edge-to-centre power quality monitoring system (as shown in Figure 8). This system combines edge computing with deep learning to achieve edge PQD signals detection.

The edge nodes of the system are composed of IoT devices, and various loads are monitored through IoT-based power quality sensors. The edge devices download the trained network model from the data centre before it starts to work, and then use local calculation ability to perform PQD detection when it starts working. The data centre is responsible for model training, which can continuously receive new sample data from the edge devices. Until the model reaches a higher accuracy test result, the trained network model and parameters will be deployed to the edge devices to realize detection at the edge. Finally, IoT integrates data from multiple sources, excludes normal signals, and only sends data related to PQD to the data centre for further training and decision making. This type of data is much less than the original data collected by the edge devices. In this way (as shown in Figure 8), the edge nodes constantly interact with the data centre and exchange information.

### 4.7 Analysis of the advantages of proposed method

In this section, we analyse the advantages of the proposed method in the following four aspects.

(i) This paper presents a new concise deep learning–based sequence model and uses it for end-to-end PQD detection. This method avoids the experience-based feature extraction process of traditional strategies (signal processing, feature extraction, classification), and realizes automatic feature extraction and selection, which can reduce considerable workload and model complexity. And to a certain extent, the inherent defects of various signal transform methods are avoided.

(ii) The method proposed here has tested 15 types of single and compound disturbances, and has a high accuracy. Its accuracy is higher than most traditional methods and deep learning methods listed in Table 4. Although its accuracy is lower than [9, 12], and [28], [9] and [12] only consider nine types of disturbances, and [28] needs to extract 45 empirical features.
Considering comprehensively, the method proposed here is more suitable for PQD detection.

(iii) Compared with the LSTM and GRU models, the SGRN model has obvious advantages in terms of number of parameters (i.e. memory cost) and calculation speed. The parameters of the SGRN model used here is less than half of LSTM and GRU, so there is no need to study other neural network compression algorithms, and it is easier to deploy in resource-constrained IoT devices.

(iv) We test the SGRN model with the “Real-life power quality sags” data set, the test result shows that only two samples were identified incorrectly, indicating that the proposed model has good generalization ability and is suitable for PQD detection problems.

5 | CONCLUSION AND FUTURE WORK

In order to detect the PQD signals in the IoT environment, this paper has proposed a new deep learning–based sequence model SGRN and a feasible edge-to-centre power quality monitoring system. In particular, the recurrent cell of this model contains two gates: forget gate and input gate, which makes the model be able to retain long-term memory and retain better gradients to avoid the problem of gradient vanishing. The recurrent cell of this method only contains two weight matrices, and the number of parameters is less than half of LSTM and GRU. This makes the proposed model more smaller and more easier to be deployed in resource-constrained edge devices. In order to evaluate the effectiveness of SGRN, 15 types of single and compound disturbances have been tested. Experimental results have shown that, compared with traditional signal processing-related methods, the proposed method can avoid artificial feature selection while ensuring a higher accuracy. Compared with other detection methods based on deep learning, the proposed method also has a higher accuracy rate. However, due to the complexity of the actual working environment and the existence of noise interference, in the future work, we will try to stack sparse denoising autoencoder on the basis of the existing model to improve the robustness of the detection model and try to deploy it in the edge devices.

REFERENCES

1. Mishra, M.: Power quality disturbance detection and classification using signal processing and soft computing techniques: A comprehensive review. Int. Trans. Electr. Energ. Syst. 29(8), 1–42 (2019)
2. Granados-Lieberman, D., et al.: Techniques and methodologies for power quality analysis and disturbances classification in power systems: a review. IET Gener. Transm. Distrib. 5(4), 519–529 (2011)
3. Urbina-Salas, I., et al.: Instantaneous power quality indices based on single-sideband modulation and wavelet packet-hilbert transform. IEEE Trans. Instrum. Meas. 66(5), 1021–1031 (2017)
4. Manimala, K., et al.: Optimization techniques for improving power quality data mining using wavelet packet based support vector machine. Neurocomputing 77(1), 36–47 (2012)
5. Erşişi, H., et al.: Optimal feature selection for classification of the power quality events using wavelet transform and least squares support vector machines. Int. J. Elect. Power Energy Syst. 49, 95–103 (2013)
6. Thirumala, K., et al.: A classification method for multiple power quality disturbances using EWT based adaptive filtering and multiclass SVM. Neurocomputing 334, 265–274 (2019)
7. Borrás, M.D., et al.: Disturbance ratio for optimal multi-event classification in power distribution networks. IEEE Trans. Ind. Electron. 63(5), 3117–3124 (2016)
8. Zhao, C., et al.: Novel method based on variational mode decomposition and a random discriminative projection extreme learning machine for multiple power quality disturbance recognition. IEEE Trans. Ind. Inf. 15(5), 2915–2926 (2019)
9. Li, J., et al.: Detection and classification of power quality disturbances using double resolution S-transform and DAG-SVMs. IEEE Trans. Instrum. Meas. 65(10), 2302–2312 (2016)
10. Luo, Y., et al.: Three-layer Bayesian network for classification of complex power quality disturbances. IEEE Trans. Ind. Inf. 14(9), 3397–4006 (2018)
11. Razo-Hernandez, J.R., et al.: Homogeneity-PMU-based method for detection and classification of power quality disturbances. Electronics 7(12), 433 (2018)
12. Cai, K., et al.: Classification of power quality disturbances using Wigner-Ville distribution and deep convolutional neural networks. IEEE Access 7, 119099–119109 (2019)
13. Mohan, N., et al.: Deep power: Deep learning architectures for power quality disturbances classification. In: Proceedings of the International Conference on Technological Advancements Power Energy, Kollam, India, 1–6 (2017)
14. Rodrigues Junior, W.L., et al.: Classification of power quality disturbances using convolutional network and long short-term memory network. In: Proceedings of the International Conference on International Joint Conference on Neural Networks (IJCNN), Budapest, Hungary, 1–6 (2019)
15. Deng, Y.P., et al.: A sequence-to-sequence deep learning architecture based on bidirectional GRU for type recognition and time location of combined power quality disturbance. IEEE Trans. Ind. Inf. 15(8), 4481–4493 (2019)
16. Medina-Gracia, R., et al.: Power quality sensor for smart appliance’s self-diagnosing functionality. IEEE Sens. 20(19), 9486–9495 (2019)
17. Chen, Z., et al.: Deep neural network acceleration based on low-rank approximated channel pruning. IEEE Trans. Circuits Syst. I 67(4), 1232–1244 (2020)
18. Gupta, C., et al.: ProtoNN: Compressed and accurate kNN for resource-scarce devices. In: Proceedings of the International Conference on International Conference on Machine Learning, Sydney, Australia, 1331–1340 (2017)
19. Hochreiter, S., Schmidhuber, J.: Long short-term memory. Neural Comput. 9(8), 1735–1780 (1997)
20. Zhao, R., et al.: Machine health monitoring using local feature-based gated recurrent unit networks. IEEE Trans. Ind. Electron. 65(2), 1539–1548 (2018)
21. Zhang, B., et al.: Simplifying neural machine translation with addition-subtraction twin-gated recurrent networks. In: Proceedings of the International Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, 4273–4283 (2018)
22. Pascaru, R., et al.: On the difficulty of training recurrent neural networks. In: Proceedings of the International Conference on International Conference on Machine Learning, Georgia, USA, 1310–1318 (2013)
23. Markovska, M., et al.: Real-time implementation of optimized power quality events classifier. IEEE Trans. Ind. Appl. 56(4), 3431–3442 (2020)
24. Lgual, R., et al.: Integral mathematical model of power quality disturbances. In: Proceedings of the 18th International Conference on Harmonics and Quality of Power (ICHQP), Ljubljana, Slovenia, 1–6 (2018)
25. Zhong, T., et al.: Power-quality disturbance recognition based on time-frequency analysis and decision tree. IET Gener. Transm. Distrib. 12(18), 4153–4162 (2018)
26. Huang, J.S., et al.: SVM-based PQ disturbance recognition system. IET Gener. Transm. Distrib. 12(2), 328–334 (2018)
27. Ahila, R., et al.: An integrated PSO for parameter determination and feature selection of ELM and its application in classification of power system disturbances. Appl. Soft Comput. 32, 23–37 (2015)
28. Dalai, S., et al.: Cross-spectrum analysis-based scheme for multiple power quality disturbance sensing device. IEEE Sens. 15(7), 3989–3997 (2015)
29. Thirumala, K., et al.: Tunable-Q wavelet transform and dual multiclass SVM for online automatic detection of power quality disturbances. IEEE Trans. Smart Grid 9(4), 3018–3028 (2018)
30. Sahani, M., Dash, P.K.: Automatic power quality events recognition based on Hilbert Huang transform and weighted bidirectional extreme learning machine. IEEE Trans. Ind. Inf. 14(9), 3849–3858 (2018)
31. Ahajjam, M.A., et al.: Electric Power Quality Disturbances Classification based on Temporal-Spectral Images and Deep Convolutional Neural Networks. In: Proceedings of the International Conference on International Wireless Communications and Mobile Computing (IWCMC), Limassol, Cyprus, 1701–1706 (2020)
32. Florencias-Oliveros, O., et al.: Real-life power quality sags. IEEE Dataport, 2017, Available: https://doi.org/10.21227/H2K88D.

How to cite this article: Zu X, Wei K. A simple gated recurrent network for detection of power quality disturbances. IET Gener Transm Distrib. 2021;15:751–761. https://doi.org/10.1049/gtd2.12056