Scalable Computing: Practice and Experience, ISSN 1895-1767, http://www.scpe.org
© 2024 SCPE. Volume 25, Issues 6, pp. 5161–5170, DOI 10.12694/scpe.v25i6.3259

# THE DEFECT IDENTIFICATION SYSTEM OF ELECTROMECHANICAL EQUIPMENT ON THE EDGE SIDE OF THE POWER GRID UNDER EDGE COMPUTING

HAIAN HAN*AND FAN HU†

**Abstract.** With the development of the industrial Internet of Things, modern industrial systems have developed towards intelligence. Electromechanical Equipment (EE) is essential, and its defect identification is fundamental. Firstly, this research introduces the basic content of Gated Recurrent Unit (GRU), Variational Auto-Encoder (VAE) in Deep Learning (DL), and Edge Computing (EC) to explore the construction of a defect identification system for EE on the edge side of the power grid. Secondly, combined with the advantages of GRU and VAE, a GRU-VAE defect recognition model is proposed. Then, the EC architecture is introduced, and the EE defect identification system based on the GRU-VAE algorithm is constructed. The EC intelligent EE defect identification service system is designed with this as the core. Finally, simulation experiments are carried out using different data sets to verify the performance of the GRU-VAE model. The results show that the GRU-VAE model has higher precision and recall than the separate GRU model and VAE model, and the corresponding F1 value is also higher. The F1 value can reach 0.997 on aperiodic data and 0.966 on periodic data. In addition, the optimal thresholds of different datasets are analyzed, and the relationship between the length of the time window and the model's performance is studied. When the time window length is 15, the model performance is optimal. This research on the defect identification system of EE on the edge side of the power grid based on EC and DL can provide a new path and inject new vitality into the defect detection of EE.

**Key words:** maintenance of the electromechanical equipment, edge computing, gated recurrent unit, variational auto-encoder, defect detection

**1. Introduction.** Electromechanical Equipment (EE) is an integral part of Industrial Internet of Things (IIoT) production activities. With the development of the IIoT, the development of EE also tends to be intelligent. Transportation, intelligent appliances, and computers have become indispensable EE products in people's lives. EE is widely used in intelligent manufacturing, transportation, smart city, and other fields [1]. However, failures due to long-term operation and lack of maintenance of EE often occur. Therefore, people pay more and more attention to the defect identification of EE. However, the complexity of electromechanical systems makes maintenance increasingly expensive. Machine Operation and Maintenance (O&M) gradually shifts from manual O&M to intelligent O&M [2]. Currently, most O&M data relies on Cloud Computing (CLO) platforms. In the IIoT, intelligent terminal application scenarios pay more attention to real-time service feedback. CLO has gradually exposed the shortcomings of high data transmission delay, untimely processing, and data privacy leakage. Then, Edge Computing (EC) emerged.

The combination of IIoT and intelligent O&M has become a development trend. Scholars have done much research in this regard. Based on the fifth-generation communication technology analysis and IoT technology, Liu et al. (2020) proposed the reference architecture of smart factories and its application path for traditional manufacturing enterprises in China. They combined the IoT with big data and EC to design a real-time tracking and monitoring system for intelligent workshop products [3]. Pivoto et al. (2021) investigated the major cyber-physical system architecture models available in industrial settings, highlighting their key features, technologies, and correlations. They pointed out the goals, advantages, and contributions of introducing IIoT in Industry 4.0 and identified the leading technologies adopted in current state-of-the-art cyber-physical systems and IIoT technologies [4]. Zhu et al. (2020) introduced the automation transformation of a sewage treatment plant based on practical engineering experience. They built a remote measurement and control system for sewage treatment based on the IIoT to realize wide-area monitoring and control of sewage treatment [5]. These studies combine IIoT with different modern technologies to explore the advantages and intelligence of IIoT from

*State Grid Shanxi Electric Power Company, Taiyuan, 030021, Shanxi, China (Corresponding author, HaianHan@126.com)
†State Grid Shanxi Electric Power Company, Taiyuan, 030021, Shanxi, China (FanHu926@163.com)

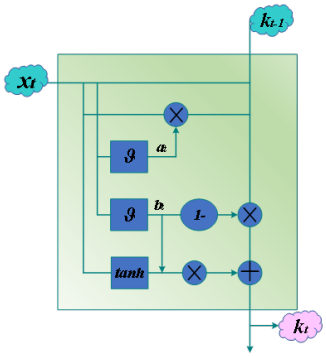


Fig. 2.1: GRU structure

different perspectives, but few studies consider defect and fault identification of EE.

Based on the above background, starting from the theoretical knowledge of Deep Learning (DL) and EC, the structure and workflow of Gated Recurrent Unit (GRU) and Variational Auto-Encoder (VAE) and the basic content of EC are briefly introduced. The GRU and VAE are fused. The EC architecture is introduced to build a defect detection algorithm model. The defect identification system of EC intelligent EE is designed, taking the defect detection model of GRU-VAE as the core. Meanwhile, the architecture of the system and its core part, the EE defect detection service system, are introduced. This research can provide ideas for establishing a defect identification system for EE on the edge side of the power grid.

## 2. EC EE defect detection system.

**2.1. Theoretical basis of DL and EC.** DL models often consist of artificial neural networks known as Deep Neural Networks (DNNs). It can simulate complex functions, learn the underlying laws of data through network structure, and achieve excellent performance in image classification, natural language processing, face recognition, and other fields [6]. Common DNNs include Convolutional Neural Networks (CNNs), Recurrent Neural Networks (RNNs), and generative neural networks. RNNs have the problem of gradient vanishing and explosions. Scholars optimized it and proposed the GRU to solve this problem [7]. GRU is a variant of RNN, a gating mechanism unit. Its structure is shown in Figure 2.1 [8].

In Figure 2.1, GRU mainly comprises update and reset gates. The update gate controls the effect of the previous moment state information on the current moment state, and the reset gate controls the degree of ignoring the previous moment state information to obtain a long time memory capability. The reset gate vector can be calculated according to Eqaution 2.1.

$$a_t = \theta(Q_a x_t + W_a k_{t-1} + \beta_a) \tag{2.1}$$

In Equation 2.1, $x_t$ is the input data at time t, $a_t$ represents the reset gate vector. $k_{t-1}$ represents state information at the previous moment of time t. $\theta$ stands for sigmoid function, whose output value is between zero and one, which is used to choose how much information is left. Q and W are the weight matrices, and $\beta_a$ is the reset gate parameters. The calculation process of the update gate vector is similar to that of the reset gate, as expressed in Eqaution 2.2.

$$b_t = \theta(Q_b x_t + W_b k_{t-1} + \beta_b) \tag{2.2}$$

In Eqaution 2.2, $b_t$ is the update gate vector, and $\beta_b$ is the update gate parameter. The updated value $\widetilde{k_t}$ is jointly determined by the reset gate vector $a_t$, the output $k_{t-1}$ at the previous moment, and the input $x_t$ at that moment, as given in Eqaution 2.3.

$$\widetilde{k_t} = tanh[Qx_t, W(a_t \cdot k_{t-1})] \tag{2.3}$$

The final output at the current moment is determined by the updated value $\widetilde{k_t}$, the update gate $b_t$, and the previous moment's input $k_{t-1}$ together, as shown in Eqaution 2.4.

$$k_t = b_t \cdot k_{t-1} + (1 - b_t) \cdot \widetilde{k_t} \tag{2.4}$$

In generative neural networks, VAE is a generative model based on an encoder-decoder framework [9]. The encoder effectively encodes the input data and obtains the characteristics of the input data. The decoder is connected to the encoder, and the data encoded by the encoder is restored to new data similar to the input data after the encoder. The encoder-decoder performs lossy compression and decompression of the data, which has the role of noise reduction and extraction features. The encoding process of VAE mainly models the structure of existing data sets. It captures the relationships between different dimensions of time series data and learns the distribution of low-dimensional hidden variables. The decoding process of VAE generates new data that conforms to the input data distribution by adding white noise [10].

EC is relative to CLO. CLO is a virtual and manageable computing and storage capacity driven by economies of scale. It is also a large-scale distributed computing model delivered to external users through the Internet according to user needs [11]. CLO can provide services to users anytime, anywhere, using resources such as shared computing facilities and applications on demand. However, the upper-layer computing applications of EE are more demanding to run. CLO can no longer meet the requirements of faster real-time and higher interactivity of new services. To solve this problem, scholars have proposed EC. The main computing nodes and applications of EC are distributed in data centers close to terminals, which makes service response performance and reliability higher than the traditional centralized CLO concept. The act of collecting and analyzing data occurs in local devices and networks close to where the data is generated, without having to transfer the data to the cloud, where computing resources are centralized for processing [12]. Different application scenarios subdivide EC into mobile EC, micro CLO, and fog computing [13]. When EC is interactively fused with Artificial Intelligence (AI), the AI frontier is pushed to the edge of the network to unlock the potential of big data at the edge. Intelligent decision-making capabilities are provided to end devices at the data source, resulting in intelligence at the edge [14]. Edge intelligence can empower edge devices with environmental awareness and data analysis, thereby improving the service quality of EC.

**2.2. Defect detection algorithm based on GRU-VAE.** GRU and VAE were used to integrate the two to construct a defect detection model of EE based on GRU-VAE. Therefore, the GRU-VAE model has not only the noise reduction and defect detection functions of the VAE model but also the long-term correlation of the GRU model to capture EE data. The role of long-term prediction of time series data can solve the problems of spatial dimension and time dimension of EE data. Its structure is demonstrated in Figure 2.2.

Figure 2.2(a) shows that the GRU-VAE defect detection model is divided into three parts: VAE encoder based on the CNN network layer, prediction module based on GRU, and VAE decoder based on the CNN network layer. The GRU module is embedded between the encoder and decoder of the VAE model. The preprocessed time series data $X_t$ is fed into the VAE encoder, which is transformed into a low-dimensional hidden variable $h_t$. The GRU prediction module learns the transformation trend of time series data and predicts the hidden variable $\hat{h}_{t+1}$ of the next time series. The hidden predicted variable $\hat{h}_{t+1}$ is connected to the VAE decoder, and the new $\hat{x_{t+1}}$ data is obtained by decoding the predicted hidden variable $\hat{h}_{t+1}$. Finally, the defect detection of EE is realized by calculating the reconstruction error. The detailed process is as follows.

First, the dataset $X_t = \{x_{t-l-1}, x_{t-l}, \cdots, x_t\}$ represents the window vector of l consecutive time series data in front of the t moment, and $X_t$ is the input of the VAE encoder. $X_t$ passes through the convolution layer of the VAE encoder and performs vector operations with the convolution kernel to complete the sampling and feature extraction of the $X_t$ window vector. The encoder encodes $X_t$ into the hidden variable $h_t$ in the potential space by Equation 2.5.

$$h_t = encoder(X_t) \tag{2.5}$$

The hidden variable is connected with the GRU time series hidden variable prediction model. The structure of the GRU time series hidden variable prediction model is shown in Figure 2.2(b) [15]. The hidden variable $h_t$ is used as input to the GRU module, and GRU predicts the final output $\hat{h}_{t+1}$, as shown in Eq. 2.6-Eq. 2.7.

$$k_t = GRU(h_t) \tag{2.6}$$

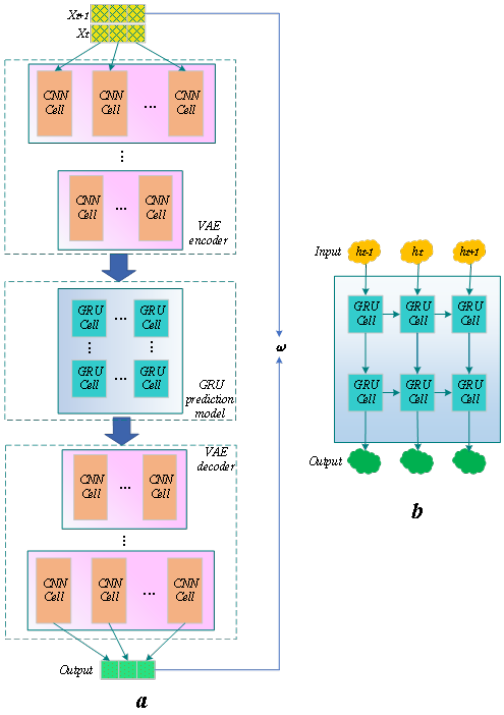


Fig. 2.2: GRU-VAE defect detection model (a is the GRU-VAE defect detection model; b is a GRU-based time series hidden variable prediction model)

$$\hat{h}_{t+1} = k_t \tag{2.7}$$

The predicted hidden variable $\hat{h}_{t+1}$ is reconstructed by the VAE decoder into the time series window $\hat{X_{t+1}}$ at the next moment. Window $\hat{X}_{t+1}$ of the VAE model reconstruction does not contain defect data and noise. The location of the window defect is determined by calculating the reconstruction error $\omega_{t+1}$ with the real-time series window $X_{t+1}$, as shown in Eq. 2.8-Eq. 2.9.

$$\hat{X}_{t+1} = decoder(\hat{h}_{t+1}) \tag{2.8}$$

$$\omega_{t+1} = ||\hat{X}_{t+1} - X_{t+1}||_2 \tag{2.9}$$

**2.3. EC EE defect detection service system architecture.** The IoT and intelligent monitoring technology of EE are combined, and the design concept of EC architecture is introduced to build a defect detection service system for EE based on EC. The computing and storage capabilities of the cloud center are deployed to edge nodes. The EE defect detection model based on DL is directly deployed at the edge node, which can realize real-time processing and rapid response to defect detection and provide edge intelligent services. The overall architecture of the EE defect detection service system based on edge intelligence is displayed in Figure 2.3.

From Figure 2.3, the EC intelligent EE defect detection service platform is divided into the terminal device layer, edge node layer, and cloud center layer. The terminal device layer is responsible for sensing EE data in the EC intelligent platform and transmitting data through the network. Meanwhile, the terminal device receives and executes the control command from the edge node to control the EE in real-time. As an extension of the cloud hub on the data source side, the edge node has the cloud center's computing, storage, and communication capabilities. It also has functions such as protocol conversion, data collection, data storage, data transmission,

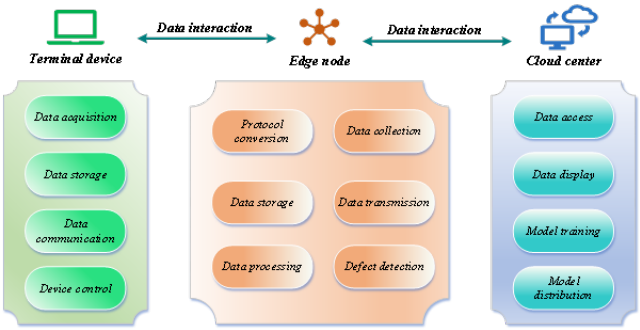


Fig. 2.3: EC EE defect detection service system architecture

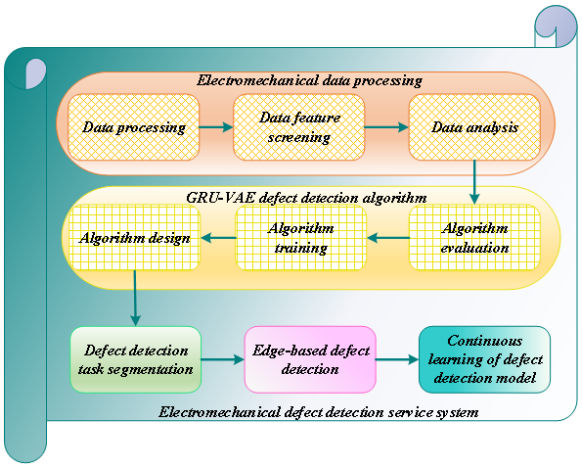


Fig. 2.4: EE defect detection service system

data processing, and defect detection. In addition, the edge node is the bridge of the EC intelligence platform. The edge node parses the EE data collected by the terminal device, obtains the original data of the EE and transmits it to the cloud center, and establishes a database to store the original data. Edge nodes receive real-time data from EE and merge them with historical data. After data processing, the DL defect detection model is used to monitor the EE in real-time. The cloud center layer has the functions of data access and storage of EE. It analyzes and mines EE data with powerful computing power. The cloud center builds an intelligent O&M system for EE. According to the geographical location, monitoring status, historical status, and other factors of the EE, the functions of data monitoring, equipment management, data analysis, and other functions are uniformly managed.

The core of the intelligent EE defect detection service platform based on EC is the EE defect detection service. The EC intelligent platform provides data collection, interaction, storage, and edge device access functions for EE defect detection. It combines EC, CLO, and DL-based defect detection, trains the GRU-VAE EE defect detection model with the powerful computing resources of the cloud center, and deploys the GRU-VAE defect detection model trained by the cloud center on edge nodes. The EE defect detection service system is built. Figure 2.4 displays its structure.

From Figure 2.4, the EE defect detection service system is divided into five parts: EE data processing, GRU-VAE defect detection algorithm, defect detection task segmentation, edge-based defect detection, and continuous learning of the defect detection model. EE data processing is divided into three sub-parts: data

analysis, feature screening, and data processing. They are mainly used for analyzing EE data to discover specific data patterns, mine the potential value of data, screen features to extract dimensional features related to EE defects, process and delete redundant data, fill in missing data, and standardize data. The GRU-VAE defect detection algorithm is the core of the entire EE defect detection service system. The standard data after the data processing of EE is divided into the training set, verification set, and test set. The training set enters the GRU-VAE defect detection model with a fixed window length and continuously trains and updates the GRU-VAE model parameters. After model training and verification, the GRU-VAE model detects the test set of EE data and evaluates the defect detection performance of the model. Defect detection task segmentation uses EC to fuse with the GRU-VAE model to segment the inspection task. GRU-VAE model training is placed in the cloud center, and GRU-VAE model inference is placed in edge nodes. Edge-based defect detection refers to cloud center training models and delivering model files to edge nodes. Edge nodes detect EE data in real-time close to the data source. The continuous learning of the defect detection model is aimed at the actual use scenarios of EE, prevents the static model phenomenon of the GRU-VAE model, and continuously learns the model regularly to improve the model detection performance.

**2.4. GRU-VAE model evaluation method.** The performance of the GRU-VAE model is verified using precision, recall, and harmonic mean F1 values [16]. Precision indicates the proportion of true defect samples to total defect samples in defect detection. It is obtained according to Eq. 2.10.

$$Pre = \frac{TP}{TP + FP} \tag{2.10}$$

In Eq. 2.10, TP means that the true value of the sample is positive, the model's predicted value is also positive, and the correct sample is judged. FP means that the true value of the sample is negative, the model's predicted value is positive, and the correct sample is wrong. The recall rate represents the proportion of true defect samples to true samples in the detection, calculated as shown in Eq. 2.11.

$$Rec = \frac{TP}{TP + FN} \tag{2.11}$$

In Eq. 2.11, FN means that the true value of the sample is negative, the model's predicted value is positive, and the correct sample is wrong. The F1 value is the harmonic mean of recall and precision, which is the final evaluation term of the experiment, and the performance of the model detection is comprehensively evaluated with the F1 value. F1 value can be calculated according to Eq. 2.12.

$$F1 = \frac{2 * Pre * Rec}{Pre + Rec} \tag{2.12}$$

In the calculation of F1 values, precision and recall are weighted equally.

## 3. Simulation of experimental design.

**3.1. Simulation experiment environment and data set.** This simulation experiment is based on a server with a high-performance graphics processor. The Central Processing Unit (CPU) is Inter(R) Core(TM) i7-8700K CPU@3.70GHz. The graphics processor is the NVIDIA GeFoece RTX 2080. Table 3.1 demonstrates the parameters of the GRU-VAE model.

In this simulation experiment, an EE data set of the highway power grid is collected. Two public datasets are collected: Machine Internal Temperature (MT) data set and the Electrocardiogram (ECG) data set. The data set is divided into the training set, validation set, and test set according to the ratio of 5:1:4. The training and validation sets include only normal data. The test set includes normal data and defect data The EE dataset is divided into two dimensions: current I and temperature T. The current data set EE-I is aperiodic, and the temperature data set EE-T is periodic. Among the two public datasets, the MT dataset is aperiodic, and the ECG dataset is periodic.

Table 3.1: GRU-VAE model parameters

| Time window size | | 14 |
|---|---|---|
| Batch size | GRU | 32 |
| | VAE | 32 |
| Number of training | GRU | 55 |
| | VAE | 55 |
| Learning rate | GRU | 0.0003 |
| | VAE | 0.0005 |

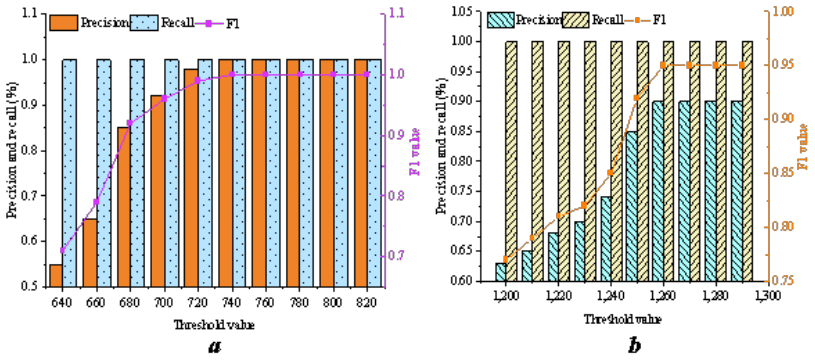


Fig. 3.1: Defect detection performance of EE in EE data sets under different thresholds (a is the defect detection performance of EE under EE-I data sets; b is the defect detection performance of EE under EE-T dataset)

**3.2. Dynamic threshold selection model performance simulation experimental analysis.** The GRU-VAE defect detection algorithm detects defects in EE data through reconstruction error. The standard for defect definition is that the reconstruction error is not less than the threshold $\omega$. The current data is marked as defective, and the opposite is normal. If the threshold is too small, the GRU-VAE model is more sensitive to defect data, and some normal data are mistakenly detected as defect data. If the threshold is too large, the model appears less sensitive. Defective data is misdetected as normal data, and the model's usefulness is reduced. The dynamic threshold selection algorithm is used to select the threshold of four sets of data sets, EE-I, EE-T, MT, and ECG. The defect detection performance of EE of EE data under different thresholds is shown in Figure 3.1.

From Figure 3.1(a), the F1 value of the EE-I dataset increases with the increase of the threshold and reaches stability at the threshold of 740, which is 1. So, the optimal threshold of the EE-I dataset is 740. From Figure 3.1(b), the F1 value of the EE-T dataset increases with the increase of the threshold and reaches a stable level of 0.95 when the threshold is 1,260. So, the optimal threshold of the EE-T dataset is 1,260. The defect detection performance of EE under different thresholds of MT and ECG data sets is shown in Figure 3.2.

From Figure 3.2(a), the F1 value of the MT dataset decreases with the increase of the threshold and peaks at 0.96 at the threshold of 6,400. Then, the F1 value decreases significantly. So, the optimal threshold of the MT dataset is 6,400. From Figure 3.2(b), the F1 value of the ECG dataset decreases with increasing threshold, peaking at 460 at 0.93. After that, the F1 value decreases significantly. Therefore, the optimal threshold of the ECG dataset is 460.

**3.3. Time series window performance simulation experimental analysis.** This simulation experiment analyzes the relationship between the time series window size of the GRU prediction module and the performance of GRU-VAE defect detection to verify the influence of time window length on local defects and time-dependent defects. Besides, 15 data lengths are a time window, and the number of time windows is entered

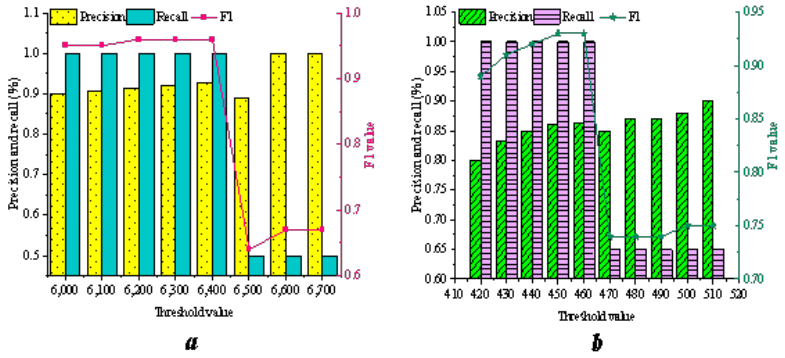


Fig. 3.2: Defect detection performance of EE in MT and ECG datasets under different thresholds (a is the defect detection performance of EE under the MT dataset; b is the defect detection performance of EE under the ECG dataset)

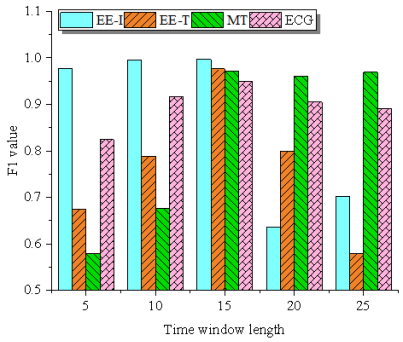


Fig. 3.3: Time series window performance experiment

into the GRU model with different time windows. The influence of 5 to 25-time window lengths on the F1 value of GRU-VAE is evaluated. Figure 3.3 reveals the results.

From Figure 3.3, with the increase of the length of the time window, the F1 value on the four datasets first increases, peaks when the time window length is 15, and begins to gradually decrease after reaching the peak. The time window length directly affects the correlation before and after the time series data. It determines the input time window length of the GRU prediction module to predict the next time window. When the length of the time window is too small, the GRU prediction module is not enough to obtain the distribution law of the time series of the time window. It is difficult to support the time series data for long-term prediction, so the F1 value of the model is low. When the time window is too large, defect data is diluted, resulting in degraded model performance. Therefore, when the other experimental conditions are unchanged, the time window length is 15, the GRU-VAE model fits the time series data best.

**3.4. EE defect detection model performance simulation experimental analysis.** When performing model performance comparison experiments, the dataset is experimented with in batches according to periodic and aperiodic periods. Both the separate GRU and VAE models are compared. The performance test results of the aperiodic data are provided in Figure 3.4.

From Figure 3.4, in aperiodic data detection, the GRU-VAE model has a higher F1 value than both the

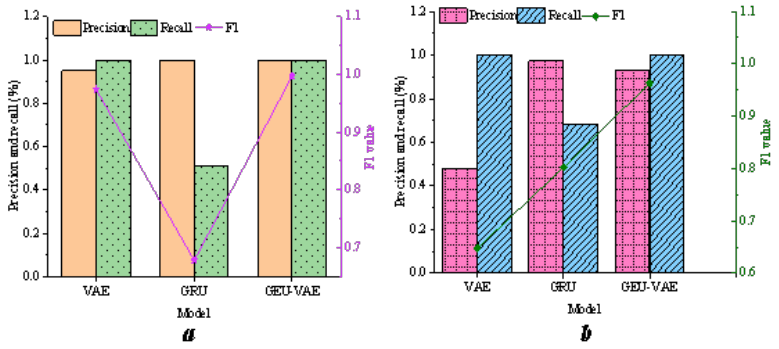


Fig. 3.4: Comparative experimental results of aperiodic data defect detection (a is the comparison result of the EE-I dataset; b is the comparison result of the MT dataset)

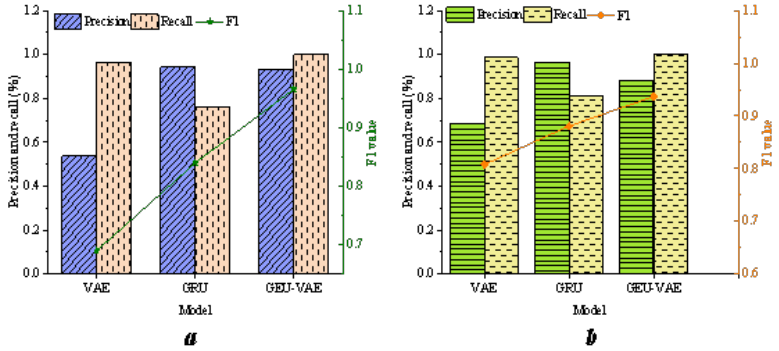


Fig. 3.5: Periodic data defect detection comparison experimental results (a is the comparison result of the EE-T dataset; b is the comparison result of the ECG dataset)

VAE and GRU models alone, and the precision and recall are also higher. In Figure 3.4(a), the F1 value of the EE-I dataset in the GRU-VAE model is 0.997, nearly 1. However, the GRU model alone performs poorly in the recall, indicating that the GRU model can recognize normal data but cannot detect defective data well. In Figure 3.4(b), the F1 value for the MT dataset in the GRU-VAE model is 0.963. The VAE model alone only has a higher recall, and the GRU model has a higher accuracy. Therefore, the F1 value of both is lower than that of the GRU-VAE model.

The detection results of periodic data are presented in Figure 3.5.

From Figure 3.5, in periodic data detection, although the recall rate of the VAE model is high, the accuracy is very low, indicating that the VAE model can detect the defect data of EE well during periodic data detection. Still, there is a defect detection error, and the defect data will be recognized as normal data. The GRU model has high precision but relatively low recall, indicating that the GRU model misses defect data when inspecting. The GRU-VAE model combines the high precision of GRU and the high recall of VAE. The F1 value is optimal compared to GRU and VAE alone, with 0.966 on the EE-T and 0.937 on the ECG datasets.

**4. Conclusion.** This research builds a GRU-VAE defect recognition algorithm model based on the VAE and GRU models in DL. It combines the advantages of both to study the defect identification of EE on the edge side of the power grid. Based on this, the EC architecture is introduced, and the EC intelligent EE defect identification system with the GRU-VAE model as the core is designed. Simulation experiments verify the

effectiveness of the model. It is found that the following conclusions: (1) The time window length is too large or too small will affect the model's performance. When the time window length is 15, the performance of the GRU-VAE model is optimal. (2) For periodic data, the recall rate of the VAE model alone is higher, and the precision of the GRU model is higher. The GRU-VAE model combines the advantages of both, with high precision and recall, and excellent F1 values up to 0.966. (3) For aperiodic data, the precision and recall of VAE on the EE-I dataset are relatively good, but there is still a certain gap compared with the GRU-VAE model. On the MT dataset, the precision of VAE and the recall of GRU are less than ideal. Therefore, the performance of the GRU-VAE model is better, and the F1 value can reach up to 0.997. However, there are some shortcomings. Only the GRU and VAE models are compared. No other DL models are compared. As a result, more models will be used in experiments to compare and find the shortcomings of the GRU-VAE model.

**5. Acknowledgement.** This study was supported by State Grid Shanxi Electric Power Company Science and Technology Project (Project No. 52053022000B).

REFERENCES

[1] Patro, E. R., Kishore, T. S., Haghighi, A. T. (2022). Levelized Cost of Electricity Generation by Small Hydropower Projects under Clean Development Mechanism in India[J]. Energies, 15(4), 1473.
[2] Kandemir, C., Celik, M. (2020). A human reliability assessment of marine auxiliary machinery maintenance operations under ship PMS and maintenance 4.0 concepts. Cognition[J], Technology & Work, 22(3), 473-487.
[3] Liu, Y., Tong, K. D., Mao, F., et al. (2020). Research on digital production technology for traditional manufacturing enterprises based on industrial Internet of Things in 5G era[J]. The International Journal of Advanced Manufacturing Technology, 107(3), 1101-1114.
[4] Pivoto, D. G., de Almeida, L. F., da Rosa Righi, R., et al. (2021). Cyber-physical systems architectures for industrial internet of things applications in Industry 4.0: A literature review[J]. Journal of Manufacturing Systems, 58, 176-192.
[5] Zhu, W., Wang, Z., Zhang, Z. (2020). Renovation of automation system based on Industrial Internet of Things: A case study of a sewage treatment plant[J]. Sensors, 20(8), 2175.
[6] Samek, W., Montavon, G., Lapuschkin, S., et al. (2021). Explaining deep neural networks and beyond: A review of methods and applications[J]. Proceedings of the IEEE, 109(3), 247-278.
[7] Zhang, Y. G., Tang, J., He, Z. Y., et al. (2021). A novel displacement prediction method using gated recurrent unit model with time series analysis in the Erdaohe landslide[J]. Natural Hazards, 105(1), 783-813.
[8] Yin, B., Zuo, R., Xiong, Y. (2022). Mineral prospectivity mapping via gated recurrent unit model[J]. Natural Resources Research, 31(4), 2065-2079.
[9] Jin, X. B., Gong, W. T., Kong, J. L., et al. (2022). PFVAE: a planar flow-based variational auto-encoder prediction model for time series data[J]. Mathematics, 10(4), 610.
[10] Yang, Z. L., Zhang, S. Y., Hu, Y. T., et al. (2020). VAE-Stega: linguistic steganography based on variational auto-encoder[J]. IEEE Transactions on Information Forensics and Security, 16, 880-895.
[11] Sadeeq, M. M., Abdulkareem, N. M., Zeebaree, S. R. M., et al. (2021). IoT and Cloud computing issues, challenges and opportunities: A review[J]. Qubahan Academic Journal, 1(2), 1-7.
[12] Alam, T. (2020). Cloud Computing and its role in the Information Technology[J]. IAIC Transactions on Sustainable Digital Innovation (ITSDI), 1(2), pp. 108-115.
[13] Meneguette, R., Grande, R. D., Ueyama, J., et al. (2021). Vehicular Edge Computing: Architecture, Resource Management, Security, and Challenges[J]. ACM Computing Surveys (CSUR), 55(1). 1-46.
[14] Deng, S., Zhao, H., Fang, W., et al. (2020). Edge intelligence: The confluence of edge computing and artificial intelligence[J]. IEEE Internet of Things Journal, 7(8), 7457-7469.
[15] Bonassi, F., Farina, M., Scattolini, R. (2021). On the stability properties of gated recurrent units neural networks[J]. Systems & Control Letters, 157, 105049.
[16] Benussi, A., Grassi, M., Palluzzi, F., et al. (2021). Classification accuracy of TMS for the diagnosis of mild cognitive impairment[J]. Brain Stimulation, 14(2), 241-249.

*Edited by:* Bradha Madhavan
*Special issue on:* High-performance Computing Algorithms for Material Sciences
*Received:* Jan 30, 2024
*Accepted:* Mar 26, 2024