Journal of Combinatorial Mathematics
and Combinatorial Computing
www.combinatorialpress.com/jcmcc

Combinatorial Press

Deep Learning-based Classification and Prediction Modeling for Big Data

Youcheng Peng[1,✉], Sihan Chen[1]

[1] *Northeastern University at Qinhuangdao, Sydney Smart Technology College, Northeastern University, Qinhuangdao, Hebei, 066004, China*

ABSTRACT

In today's rapid development of information technology, the big data industry has ushered in explosive growth, and big data analysis has become an important research topic in the cross-cutting field of computing. This study constructs a big data prediction base model based on deep learning, and uses the improved butterfly optimization algorithm with OGRU model to realize feature selection and classification processing of big data. Then the Adam algorithm is used to optimize the parameters of the model, and finally the classification and prediction model of big data based on deep learning is constructed. Simulation and empirical analysis results show that the model proposed in this paper has excellent classification and prediction performance, and can meet the efficiency requirements of big data classification and prediction. The prediction errors of distribution network load data and smart charging pile operation data are lower than 9% and 16%, respectively, which have high practical application value. This study is of great significance to the research related to big data classification and prediction in different fields, and provides an effective method for data prediction in complex scenarios such as industrial as well as power grid scheduling.

Keywords: deep learning; OGRU model; butterfly optimization algorithm; Adam algorithm; big data classification prediction

1. Introduction

With the rapid development of the Internet and the continuous progress of information technology, the application of big data has become more and more widespread and has a significant impact on the development of various industries [1]. Since the effective classification and optimization of data is one of the key issues in the application of big data, although many data classification and optimization methods exist, their efficiency is low and their accuracy is poor. Therefore, the research on big data classification and optimization methods has become a hot issue for people at present [2-3].

✉ Corresponding author.
E-mail address: 19565385668@163.com (Y. Peng).
Received 08 February 2024; Revised 18 April 2024; Accepted 30 November 2024; Published Online 15 April 2025.
DOI: 10.61091/jcmcc127a-448

At the same time should pay attention to the research of data classification prediction, as in an episode of Captain America about a very avant-garde algorithm, of course perhaps this algorithm is yet to be developed and researched. Hydra organization in order to limit the threat of others to them, the use of advanced algorithmic techniques for each person's useful information extraction, this information includes each person's school grades, to participate in the content of the training, etc., and then according to these taken information on each person may be engaged in the future to classify the prediction of the work [4-5]. If the work that these people may engage in poses a threat to the Hydra organization, then take action to rape and destroy. Behind this advanced technology, in fact, it is the method and way of classification and prediction research is taken, the feature extraction and then make information processing, data classification and prediction research malefactor can be seen as a valuable direction [6].

The mainstream research on big data classification has centered on big data classification scalability and classification accuracy, through the comparison and analysis of distributed algorithms, FRBCS themes, feature selection strategies, etc. Elkano, M et al. attempted to introduce a newly-designed FRBCS in order to enhance the fuzzy rule-based classification system based on big data technology and considered the MapReduce paradigm's applicability, which effectively improves the classification accuracy of fuzzy rule classification system based on big data technology [7]. Ducange, P et al. reviewed contemporary distributed learning algorithms for generating fuzzy classification models for big data, including the design and practical effectiveness of these algorithms, and compared some of these distributed learning algorithms around accuracy and interpretability, and investigated their scalability in depth [8]. Devi, S. G et al. reviewed the research literature related to feature selection (FS) techniques as well as online feature selection (OFS) algorithms, pointed out that the feature selection strategy can effectively reduce the noise and redundancy of the big data classification models, and revealed that online feature selection (OFS) algorithms can be used to solve the scalability problem of traditional classification algorithms [9]. Dagdia, Z. C et al. deeply analyzed the advantages and disadvantages of Dendritic Cell Algorithm (DCA), a bio-inspired classifier, and conceptualized a distributed DCA version of the data classification model with MapReduce framework as the core logic, and experimentally corroborated the feasibility of the proposed scheme for big data classification [10].

Big data analytics techniques are commonly used for forecasting in various fields, and research on big data analytics forecasting techniques in related fields tries to study the optimization of big data forecasting techniques in practice from the dimensions of improvement of big data analytics algorithms, architectural adjustments, etc., or by comparing the strengths and weaknesses of big data forecasting algorithms, in order to identify the best fit of the major big data forecasting algorithms. Fathi, M et al. conceptualized a technical and hybrid approach as a underlying architecture classification approach to compare big data analytics strategies used for weather prediction by summarizing the conditions of use, modeling, and strengths and weaknesses of these algorithms in detail [11]. Bok, B et al. attempted to analyze the practice of big data technology in econometrics in order to understand how it enables the prediction and monitoring of GDP growth, which was corroborated and discussed in detail by synthesizing relevant macroeconomic data analysis [12]. Seyedan, M et al. examined the practice of big data techniques in predictive supply chain management and categorized these big data analytics algorithms as time series prediction, clustering, k-nearest neighbor, etc. and pointed out the research gaps of predictive algorithms of big data analytics in the case of closed-loop supply chains (CLSCs) and made targeted research recommendations [13]. Wang, K et al. designed an electricity price prediction model integrating a hybrid feature selector based on grey correlation analysis (GCA), a kernel function combined with principal component analysis (KPCA) for dimensionality reduction, and a support vector machine classifier based on differential evolution (DE), which efficiently optimizes the redundancy of feature selection in the price prediction process and coordinates the lack of an integrated infrastructure for the price prediction process [14].

In this paper, the Bagging strategy in deep learning is used to construct a big data prediction base model, and the pruning strategy is used to reduce the model size of the Bagging strategy and improve the prediction efficiency of the deep learning model. The butterfly optimization algorithm is improved to obtain the SBOA algorithm, which is used to identify and select various types of features in big data, and then the recurrent neural network in the OGRU model is used to assign different classification labels to various types of data in the dataset, so as to realize the classification processing of big data. Subsequently, the Adam algorithm is used to optimize and adjust the parameters of the classification model, and a deep learning-based big data classification and prediction model is constructed on the basis of the convolutional neural network architecture. In this study, the classification accuracy of the big data classification and prediction model is first simulated and tested using engineering time series big data as the test data set. The model is then applied to two example scenarios, namely, distribution network load and smart charging pile, to analyze the classification and prediction effect of the model on big data in empirical applications.

2. Method

2.1. Big Data Predictive Modeling

2.1.1. Deep learning model construction. Deep learning [15] is a machine learning strategy that combines the predictions of multiple base models. The Bagging strategy, also known as Bootstrap aggregation strategy, uses a subset of randomly selected samples to train the base model for equal-weighted ensemble voting. The Boosting strategy [16] builds a deep learning framework step-by-step by training a new base model, focusing on previously trained samples that were misclassified by the base model. Boosting deep learning models may provide better predictive performance than the Bagging strategy, but may also lead to overfitting of anomalous samples. The stacking strategy trains the meta-learner to optimally combine the predictions of the base model and relies on a sufficient number of samples to train the meta-learner.

This study uses the Bagging strategy to data predict the base model. Thirty training subsets were randomly selected to construct 30 diverse base models. Four strategies were used to combine the predictions of the 30 base models, i.e., voting, averaging, Sigmoid averaging, and weighted averaging strategies. The voting strategy determines the final prediction by the majority of the predictions from the base models. In the averaging strategy, the pooled results are computed from the average of all base models. In the Sigmoid averaging strategy, the results of all 30 base models are normalized and the average is output as the pooled result. The weighted averaging strategy uses the validation error as a weight for each base model and computes the weighted result as the output.

2.1.2. Pruning strategy. Pruning strategy [17] is a strategy to reduce the size of the Bagging model, which is utilized to improve the efficiency of the model. Two pruning strategies are used in this study. First, the validation error rate of the basic model is evaluated and the top ranked model with smaller error rate is selected, called error rate pruning. In this study, the accuracy of different models in the validation dataset was calculated and the models were ranked according to the accuracy. Models with lower accuracy were removed from the ensemble framework. Second, diversity among models was increased by removing models using a subset of models with higher richness (called diversity pruning). Three metrics were used to measure the difference between the two underlying models, namely the inconsistency metric, the double error metric, and the kappa statistic.

The inconsistency measure (DisM) between the i st and j nd base models is calculated as shown in (1):

$$DisM\left(i,j\right)=\frac{a}{m} \tag{1}$$

Variable a is the number of samples assigned to different categories for the i nd and j rd models, and variable m is the number of all samples. The higher $DisM(i,j)$ is, the greater the difference between the two models. The proposed deep learning framework attempts to select a diverse subset of models. The inconsistency $DisM(i,j)$ of each model pair is evaluated and the model pair with the highest DisM value is selected for further analysis.

The Dual Fault Metric (DF) between the i th and j th base models is calculated as shown in Equation (2):

$$DF(i,j)=\frac{e}{m} \tag{2}$$

Variable m is the number of all samples and e is the number of samples assigned to the error class by the i rd or j th base model. The model pairs are ranked according to $DF(i,j)$ and models are selected among the top ranked model pairs until a subset of models with satisfactory prediction accuracy is found.

The Kappa statistic is used for consistency assessment and can also be used to measure classification accuracy.Kappa is calculated as shown in equation (3):

$$Kappa(i,j)=\frac{P(A)-P(E)}{1-P(E)} \tag{3}$$

Variable $P(A)$ is the percentage of consistency between the i nd and j rd models. $P(E)$ is the probability of consistency. $Kappa(i,j)=1$ indicates perfect agreement between two models and $Kappa(i,j)=0$ indicates inconsistency. The pairs of models are ranked according to $Kappa(i,j)$, using their ascending numerical order, and the models in the top-ranked pairs are selected until a subset of models with satisfactory prediction accuracy is found.

2.2. Feature Selection and Classification Algorithm

2.2.1. Feature selection based on SBOA-FS. Butterfly Optimization Algorithm (BOA) [18] shows more excellent results in avoiding, converging, exploiting and exploring local optimization.BOA has some advantages over other optimization algorithms to optimize most of the well-known single and multi-exponential combinatorial distribution functions.The basic ability of BOA is performance and maximum convergence speed because of the use of an arbitrary, decentralized step. In BOA flavor's are expressed as equation (4):

$$pf_i=cI^a \tag{4}$$

The butterfly is a position vector which is upgraded in the optimization process using the method in Eq. (5):

$$x_i^{(t+1)}=x_i^t+F_i^{(t+1)} \tag{5}$$

The BOA algorithm is divided into two basic steps: global search and local search. These steps are represented in equations (6) and (7) as:

$$F_i^{(t+1)}=\left(r^2\times g^*-x_i^t\right)\times pf_i \tag{6}$$

$$F_i^{(t+1)}=\left(r^2\times x_j^t-x_k^t\right)\times pf_i \tag{7}$$

where x_k^t and x_j^t are the j rd and k th butterflies in the solution space. When x_k^t and x_j^t enter the similar totality and r denotes uniform arbitrary probabilities 0 and 1 following the equation, the local random wandering switching probability p has been developed which has been used to switch between a general global search to a local search.

To solve the feature selection (FS) problem, a new BOA algorithm, SBOA, is proposed that utilizes a Sigmoid function to move the butterfly through the binary search space. This Sigmoid function is in equation (8):

$$S\left(F_i^k(t)\right)=\frac{1}{1+e^{-F_i^k(t)}} \tag{8}$$

where $F_i^k(t)$ is the continuous-valued flavor of the i th butterfly from the k rd dimension at iteration $F_i^k(t)$.

After that, the stochastic threshold has been implemented as described in Eq. (4) Eq. (9) for the binary solution of the Sigmoid function.The Sigmoid function effectively maps infinite inputs to finite outcomes:

$$X_i^k(t+1)=\begin{cases}0 & if rand < S\left(F_i^k(t)\right)\\ 1 & otherwise\end{cases} \tag{9}$$

where $X_i^k(t)$ and $F_i^k(t)$ denote the position and flavor of the i rd butterfly at iteration t starting from dimension k.

FS is considered as a multi-objective optimization problem. In SBOA, the optimal solution contains the minimum number of features with maximum classifier accuracy. Therefore, the fitness function (FF) of the technique is expressed as:

$$Fitness=\alpha\gamma_R(D)+\beta\frac{|R|}{|N|} \tag{10}$$

where $\gamma_R(D)$ denotes the classification error rate of the classifier, $|R|$ denotes the base of the selected feature subset, and $|N|$ denotes the full amount of features from the original dataset.

2.2.2. OGRU model based data classification. After the selection of features is done, the next step is the data classification process in which the data instances are assigned to different class labels using OGRU model [19]. Recurrent Neural Network (RNN) is suitable for nonlinear time series modeling. RNN has input layer x, hidden layer h and resultant layer y. If controlling the time series data, RNN is the exact part. The result as well as the hidden layer is computed on the basis of:

$$y_t=g\left(s_t * w_{hy}\right) \tag{11}$$

$$s_t=f\left(x_t * w_{sx}+s_{t-1} * w_{ss}\right) \tag{12}$$

In this paper, we propose the use of GRU models to address the challenges of multivariate time series data processing, especially in overcoming the gradient vanishing problem encountered in classical recurrent neural networks (RNNs).

The reset gate is calculated based on the previous moment's hidden state h_{t-1} and the current moment's input x_t. This gating mechanism determines which past hidden state information should be discarded or reset in order to efficiently incorporate the new input information. The update gate is responsible for determining what percentage of the current moment's hidden state h_t should retain information from the previous moment h_{t-1}, and what percentage should be updated with candidate hidden states. Candidate hidden states (also known as new hidden state proposals) result from the combination of historical information and current inputs after considering the effects of the reset gate. Ultimately, the actual hidden state h_t at the current moment is determined by combining the result of the escalation gate with the candidate hidden state, a process expressed by Eq. (16), which realizes the selective retention of past information and the effective fusion of new information.

Throughout the training process, the gating variables z_t and r_t (corresponding to the reset gate and the upgrade gate, respectively), as well as the related weight parameter matrices W (including,

but not limited to, the gating-related weight matrices) in the GRU model are automatically adjusted and optimized by the back-propagation algorithm, in order to improve the model's ability to learn from the time series data and its prediction accuracy. The calculation formula is as follows:

$$z_t = \sigma\left(W_z * \left[h_{t-1}, x_t\right]\right) \tag{13}$$

$$r_t = \sigma\left(W_r * \left[h_{t-1}, x_t\right]\right) \tag{14}$$

$$h_t = \tanh\left(W_* * \left[r_t * h_{t-1}, x_t\right]\right) \tag{15}$$

$$h_t = \left(1 - z_t\right) * h_{t-1} + z_t * h_t \tag{16}$$

To optimally tune the hyperparameters involved in the GRU model, the Adam optimizer is used. In the Adam optimizer algorithm, the exponentially decaying average of the past gradient m_k and the past squared gradient v_k is considered as in the following equation:

$$m_k = \beta_1 m_{k-1} + \left(1 - \beta_1\right) g_k \tag{17}$$

$$v_k = \beta_2 v_{k-1} + \left(1 - \beta_2\right) g_k^2 \tag{18}$$

Whereas g_k denotes the gradient, β_1 and β_2 denote the decay rate, which is closer to 1. Accordingly m_k and v_k serve as the estimates of the first-order moments (mean) and second-order moments (non-centered variance) of the gradient. This bias is canceled out by the bias-corrected first-order moment and second-order moment estimates, as shown in the following equation:

$$\hat{m} = \frac{m_k}{1 - \beta_1^k} \tag{19}$$

$$\hat{v}_k = \frac{v_k}{1 - \beta_2^k} \tag{20}$$

Therefore, the Adam update rule is given by the following equation Eq. (21):

$$w^{(k+1)} = w^k - \alpha * \frac{\hat{m}}{\sqrt{\hat{v}, k} + \delta} \tag{21}$$

where δ denotes the smoothing term used to avoid division by zero.The evaluation procedure for Adam's method is summarized in Algorithm 1.

2.2.3. Parameter optimization. Adam is a popular optimization algorithm widely used in deep learning and other problems where a large number of parameters need to be optimized. Adam combines the ideas of two classical optimization algorithms, Momentum and Adaptive Learning Rate Tuning, and is particularly influenced by the RMSProp algorithm.

1) Adaptive learning rate. The Adam algorithm [20] tracks independent gradient first-order moment estimates (m_t, similar to the momentum term) and second-order moment estimates (v_t, similar to the sliding average squared gradient in RMSProp) for each parameter. The first-order moments are exponentially weighted moving averages of past gradients, while the second-order moments are exponentially weighted moving averages of the squared past gradients, which are used to adjust the learning rate for each parameter.

2) Correcting bias. In order to eliminate the initialization bias, Adam also corrects the bias of these two estimates to obtain the corrected first-order moments and second-order moments.

3) Parameter Updating. Model parameter θ is updated using the corrected moment estimates described above.

2.3. Model implementation details

The basic model of this paper's framework is an architecture-based convolutional neural network. This study uses the published ACNE04 dataset to evaluate the acne grading algorithm. This paper hypothesizes that neural network work optimized by the authors of the published ACNE04 dataset may be a good candidate for the base model of the deep learning framework. From there, the default values of the neural network hyperparameters are selected. That is, in the framework of this paper, stochastic gradient descent (SGD) and 32 small batches are used to optimize the base model. This experiment was trained for 120 epochs. Momentum and momentum decay values were set to 0.9 and 5e-4. The learning rate starts at 0.001 and decays 0.5 every 30 epochs. The proposed framework, AcneGrader, is a deep learner that uses a subset of different samples from the training dataset to train the base model. The algorithms in this paper run on an NVIDIA P100 GPU with 16GB VRAM.

2.4. Model flow design

The specific flow of the proposed big data classification and prediction model based on deep learning in this study is shown in Figure 1. First of all, the proposed neural network serves as the network architecture that needs to train the base model, and in order to obtain a subset of the base model with high richness and diversity, this paper trains the base network architecture by selecting a subset of different training sets. When selecting the training set, this paper uses random sampling to sample the big data training set and trains N base model based on the big data subset. Second, in order to remove redundant base models, using pruning algorithm is a good way to remove $(N-n)$ redundant base models and get n base models. Third, the selected base models are used to construct features. The training dataset consists of m sets of big data, which are predicted using n base models, and the predictions are used as features, each represented by a n-dimensional vector, and each feature corresponds to a base model. Next, the redundant features constructed in the previous step are removed using an iterative feature selection algorithm, since each feature corresponds to a base model, so that the feature selection algorithm can be used to re-remove the redundant base models for better performance. Finally, for the multi-base model learning problem, instead of using simple deep learning strategies such as voting, the final results are summarized by a classifier in order to learn to combine the results of the multi-base models, and this method enables non-linear learning for more complex deep learning strategies.

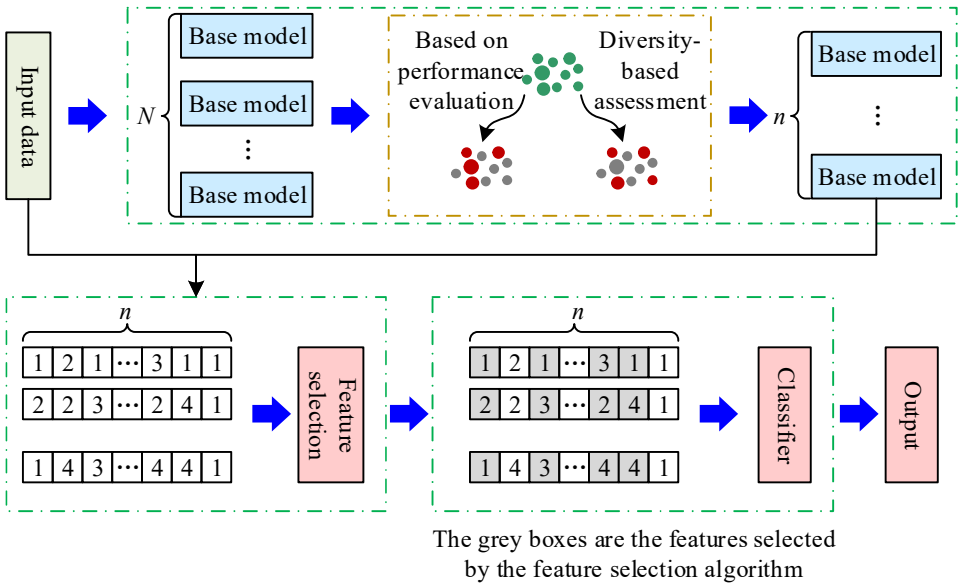


Fig. 1. Classification and prediction model process

3. Results and discussion

3.1. Model classification accuracy validation

3.1.1. Indicators for assessing the level of accuracy. There are two metrics to measure the degree of accuracy, one is the categorization accuracy Ac and the other is the multiclassification F_β metric designed in this paper in a similar way to the dichotomous F_β scores. The evaluation dataset is S, the set of categories is C, the model estimated sample label mapping is $M:S\to C$, and the real sample label mapping is $r:S\to C$.

The Ac is calculated as shown in equation (22) as the ratio of samples correctly categorized by the model to the total number of samples:

$$Ac=\sum_{X\in S}Eq\left(M\left(X\right)=r\left(X\right)\right)/\left|S\right| \tag{22}$$

Among them:

$$Eq\left(x\right)=\begin{cases}1 & x=true\\ 0 & x=false\end{cases} \tag{23}$$

The multiclassification F_β indicator is calculated as shown in equation (24):

$$\frac{1}{F_\beta}=\frac{1}{\left|c\right|}\sum_{i\in C}\frac{1}{\left(1+\beta^2\right)}\left(\frac{1}{precision_i}+\frac{\beta^2}{recall_i}\right) \tag{24}$$

i represents the calculated checking accuracy $precision$ and recall $recall$ on each category, and β represents how many times more important $recall$ is $precision$. $precision$ and $recall$ are calculated as shown in Eq:

$$precision_i=\left|PS_i\cap TS_i\right|/\left|PS_i\right| \tag{25}$$

$$recall_i=\left|PS_i\cap TS_i\right|/\left|TS_i\right| \tag{26}$$

Among them:

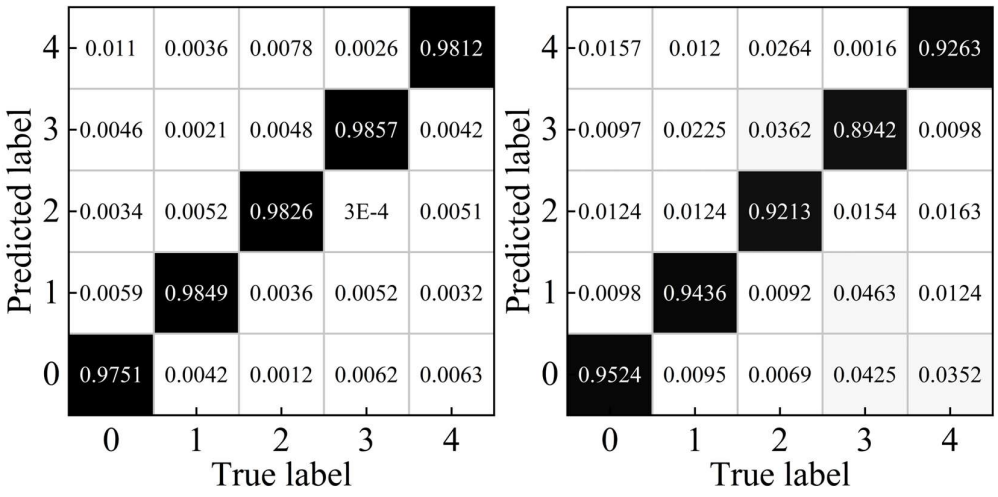
$$\begin{aligned}TS_i&=\left\{X\left|r\left(X\right)=i,X\in S\right.\right\},i\in C\\ PS_i&=\left\{X\left|M\left(X\right)=i,X\in S\right.\right\},i\in C\end{aligned} \tag{27}$$

Accuracy Ac is easy to calculate and can roughly estimate the classification accuracy of the model, while F_β is focused and averages $precision$ and $recall$. It focuses on the model's level of detection and recall, and can be changed according to the application scenarios by β trying to change the focus of the evaluation metrics. Since the experimental data in this paper belongs to the application scenario of detecting anomalous states, the recall rate is more important, and thus $\beta=2$ is used in the experiments in this paper.The average of the results of multiple model evaluations in the experiments is taken as the final model evaluation result, so as to reduce the random error.

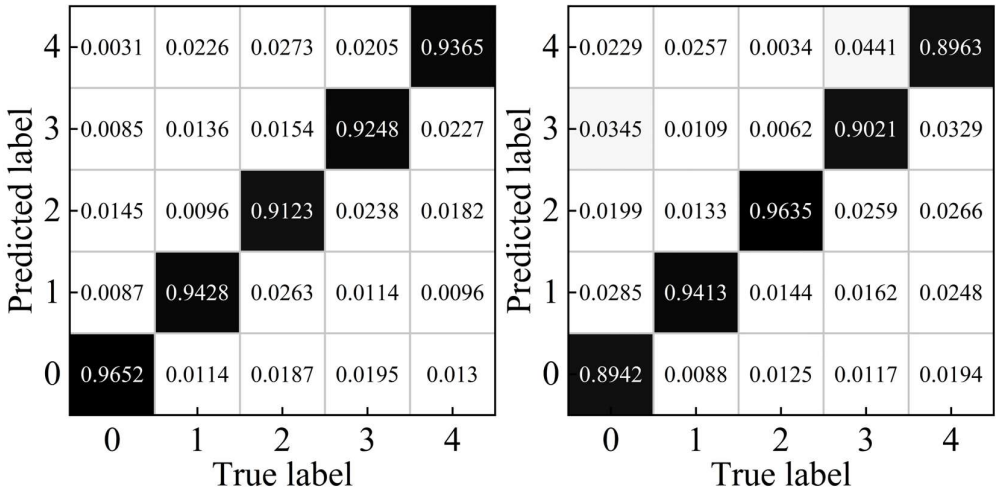
3.1.2. Simulation experiment results and analysis. In order to investigate the accuracy of the big data classification and prediction model constructed based on deep learning in this paper, the experimental data is taken from the SIS real-time/historical database of a power plant, which is the data of each sensor during the operation of the induced draft fan of unit 3 in the city. The complexity of this part of the data is high, and more than one million time series data with lengths ranging from 12 frames to 40 frames were extracted by the data preprocessing module, each frame having 63 dimensions of sensor data. There are six categories of sample data, representing different states of the device, and the

number of samples in each category is relatively balanced. In the experiment, according to the time sequence of sample collection, 750,000 samples are taken as the initial model training set, and the remaining 400,000 samples are taken as the new distribution data. The base equipment for the simulation experiment is NVIDIA GTX980Ti graphics card, CPU 3.3GHz Intel Core i7-5820K, 32GB 2400MHz DDR4 memory, and the simulation environment is Tensorflow 1.8.0 and Python 3.6.5. The neural network model in the experiment is realized using the Tensorflow framework. The neural network model in the experiment is realized using the Tensorflow framework, and the model realized under this framework can be accelerated by the GPU of the graphics card.

In the field of big data analytics, the classification and prediction performance of LSTM model, BPNN model, and EE model are superior, so these three models are selected as control models in this paper. The normalized confusion matrices of the classification results of different models are shown in Fig. 2, and (a)-(d) represent the analysis results of this paper's model, LSTM model, BPNN model, and EE model, respectively. The efficient deep learning algorithm adapted to big data classification and prediction proposed in this paper has a large improvement in the accuracy performance F_β (0.9751-0.9857) of classifying industrial time-series data compared to the BPNN algorithm (0.9123-0.9652) used in industry. In addition, comparing with the LSTM model which has the best effect in the related research, the classification accuracy of the model in this paper is significantly higher than that of the LSTM model, and the accuracy in classifying 0, 1, 2, 3 and 4 samples in the data set is improved by 2.27%, 4.13%, 6.13%, 9.15% and 5.49%, respectively. And it can meet the efficiency requirements of industrial time-series big data classification. In conclusion, the efficient deep learning algorithm for big data classification and prediction proposed in this paper meets the efficiency requirements of big data classification while ensuring high accuracy.



(a) This article (b) LSTM



(c) BPNN (d) EE

Fig. 2. The model classification results are normalized to the confusion matrix

3.2. Analysis of empirical applications of big data classification and prediction models

Through the previous analysis, it is found that the accuracy of the big data classification and prediction model constructed based on deep learning in this paper has been verified in simulation experiments, and in order to further explore the practical effect of the model in big data analysis in various fields, this study selects two scenarios of power distribution network and smart charging pile for empirical analysis. In the era of rapid development of big data, the big data in the fields of distribution network scheduling and intelligent charging pile fault detection are more complex and difficult to accurately classify and predict, so it is typical and reasonable to select these two cases for empirical analysis.

3.2.1. Empirical analysis of distribution network load forecasting:

1) Source of grid data. Under the general structure of Henan power grid, as of December 31, 2023, there are two 220 kV substations in Xiayi County, with five main transformers and a capacity of 600 MVA. The construction of the Xiayi power grid has serious deficiencies, especially poor equipment health, the automation of the distribution network has not yet taken shape, and some substations still do not meet the "N-1" requirement, etc. These factors lead to certain power supply risks in the operation of the Xiayi power grid. Therefore, this paper utilizes a big data classification and prediction model to classify and predict the data in Xiayi County's distribution network, and provides an effective model for the rational dispatch of its power grid. All the raw data used in this study come from the power dispatch data network system. The dedicated dispatch data network is a specialized wide-area data network constructed by the State Grid Corporation for power dispatch production services.

2) Overall classification and prediction results of distribution network big data. The model is used to classify and forecast the grid big data in Xiayi County for the period of 2021-2023, and after data training and adjusting the parameters, the load data with relatively high accuracy is finally obtained. Comparison and analysis with the actual value, the results of comparison and analysis of the classification and prediction load data obtained are shown in Figure 3. Calculation found that the error between the model classification and prediction of electricity load value and the real data is between 0.00%-8.72%, with an average error rate of only 1.48%, indicating that the model has a better effect on the overall classification and prediction of large data of electricity load in distribution networks. After obtaining the overall prediction results, the load predictions on weekdays and under severe

weather are extracted and analyzed in comparison with the actual values to further verify the model's classification and prediction accuracy of the distribution data.

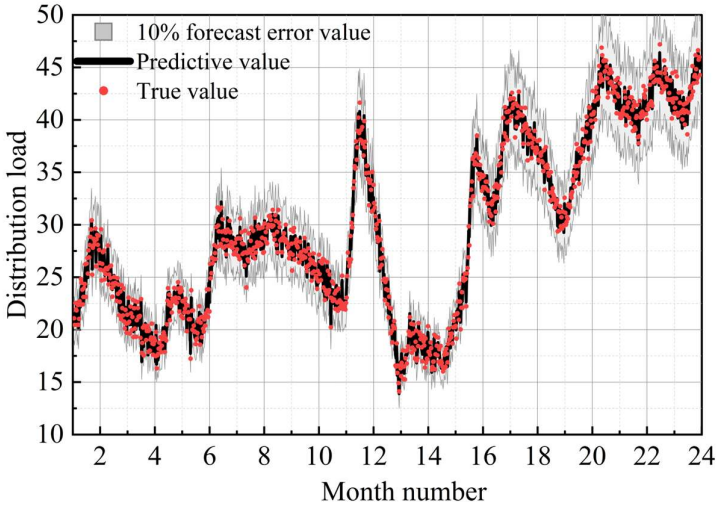


Fig. 3. The distribution network is classified and predicted

3) Weekday load forecast. It is obvious from the previous historical load that the load curve in the spring and fall seasons is the most regular, basically no sudden change in the value, so this weekday load randomly selected two weeks each of the spring and fall seasons of 2023 to analyze the data, and validate it by comparing the predicted value with the actual value, and the validation results of the spring weekday grid load data are shown in Table 1. The grid load in the early spring season is mainly residential living electricity, industrial load electricity and spring planting, in the 16 days randomly selected in the early spring season, the load prediction error is between 2% and 4% for 7 days, and the load prediction error is below 1% for 3 days, which is a very good prediction accuracy.

Table 1. Load forecast analysis of spring working day

Date	Predictive value	True value	Error
2023.4.10	25.34	25.37	0.10%
2023.4.11	25.33	25.31	0.06%
2023.4.12	24.01	23.11	3.88%
2023.4.13	24.66	24.96	1.20%
2023.4.14	23.16	22.44	3.22%
2023.4.15	27.93	27.38	2.01%
2023.4.16	25.49	24.82	2.68%
2023.4.17	27.70	26.73	3.62%
2023.4.18	26.95	26.67	1.04%
2023.4.19	25.06	25.52	1.81%
2023.4.20	22.03	22.21	0.80%
2023.4.21	22.39	22.12	1.23%
2023.4.22	26.78	26.47	1.18%
2023.4.23	26.79	26.29	1.90%
2023.4.24	24.80	24.21	2.44%
2023.4.25	27.95	27.08	3.23%

Autumn is high and cool with suitable temperature, and the load generated is similar to that of the early spring period, which is also dominated by residential living electricity consumption, industrial load electricity consumption and agricultural fall harvest. The forecast data and real data of distribution network load electricity consumption in October 2023 were verified and analyzed, and the results obtained are shown in Table 2. Of the 19 days taken in October, the load forecast error exceeded 2% on 7 days, between 1% and 2% on 3 days, and the error of the load forecast results was below 1% on the remaining 9 days. Summarizing the load forecasts for normal weekdays, it can be seen that the results of the load forecasting model can reach the desired values and have some application value.

Table 2. Forecast Load Analysis of Autumn Workday

Date	Predictive value	True value	Error
2023.10.9	29.00	28.66	1.17%
2023.10.10	28.37	27.58	2.87%
2023.10.11	26.79	26.48	1.16%
2023.10.12	25.39	24.69	2.85%
2023.10.13	30.46	29.73	2.44%
2023.10.14	25.93	26	0.28%
2023.10.15	24.95	24.9	0.21%
2023.10.16	27.55	27.64	0.34%
2023.10.17	24.83	24.26	2.36%
2023.10.18	30.01	29.96	0.16%
2023.10.19	28.50	28.12	1.35%
2023.10.20	30.26	29.58	2.29%
2023.10.21	28.68	28.86	0.64%
2023.10.22	25.84	25.13	2.82%
2023.10.23	30.31	29.66	2.20%
2023.10.24	29.48	29.46	0.07%
2023.10.25	24.24	24.34	0.40%
2023.10.26	27.60	27.64	0.13%

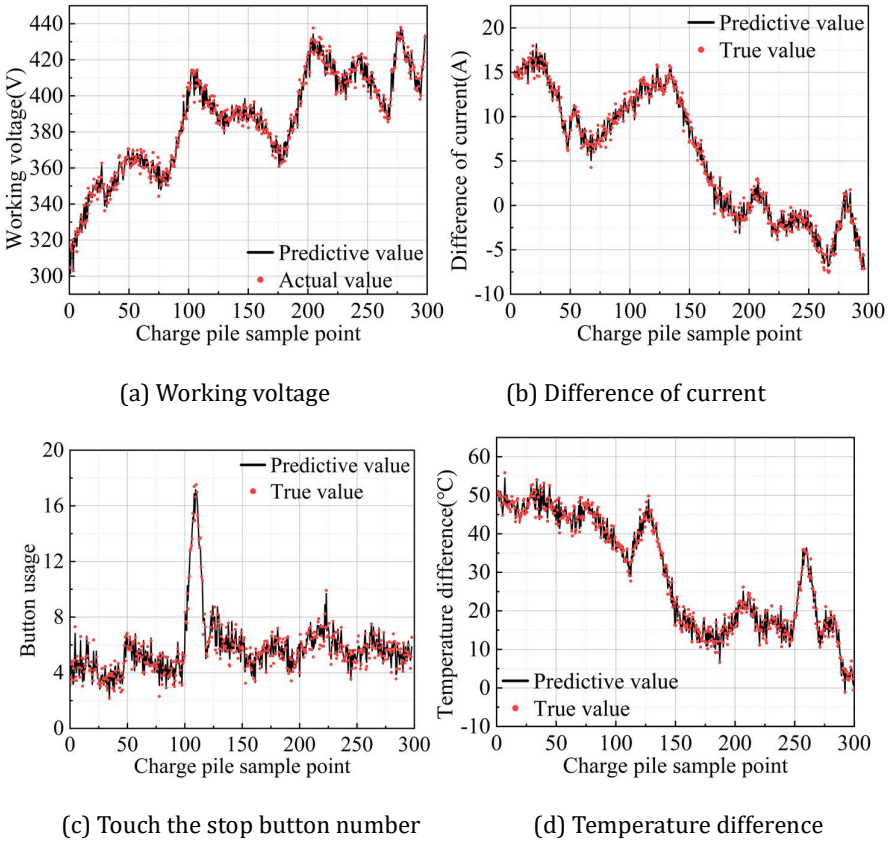
4) Analysis of heavy load day prediction. In 2023, the winter big load date of Xiayi power grid is from January 5 to 9, and the summer big load date is from July 12 to 16, respectively, the load of these 10 days is predicted and analyzed by the load prediction model, and the results of the load prediction of the distribution network on the big load days are shown in Table 3. From the prediction results, it can be concluded that the load prediction results produce a small error (4.21%-8.74%) during the summer and winter big load periods in 2023. This is because the model is able to make judgments about extreme weather, resulting in accurate predictions under such weather unlike other models. Through comparative analysis, it can be seen that the prediction accuracy of the model in normal working days fully meets the needs of the daily dispatch work of the power grid, in addition to the model can be classified in the load forecasting to consider the extreme weather factors, resulting in less error in the load prediction during the summer and winter periods, which meets the needs of the daily work of the distribution network scheduling.

Table 3. Load Forecast Analysis on Heavy Load Day

Date	Predictive value	True value	Error
2023.1.5	52.14	49.96	4.37%
2023.1.6	51.30	48.03	6.81%
2023.1.7	52.73	49.35	6.85%
2023.1.8	46.51	42.77	8.74%
2023.1.9	50.98	48.92	4.21%
2023.7.12	49.51	46.8	5.79%
2023.7.13	45.58	43.19	5.54%
2023.7.14	48.11	44.87	7.23%
2023.7.15	51.16	49.26	3.86%
2023.7.16	43.48	40.82	6.52%
2023.7.17	51.74	48.03	7.73%

3.2.2. Charging pile state classification and fault prediction analysis:

1) Validation results of intelligent charging pile big data classification and prediction. Firstly, the classification and prediction effect of the model on the data of each classification feature of smart charging pile is analyzed, and the training set adopts a total of 12,364 trouble-free charging pile data in the background management system of a smart charging pile. The big data classification and prediction model proposed in this paper is first trained, and then the model is used to classify and analyze the big data in normal smart charging piles. From the background management system, the complex work data of any charging pile in a period of time is selected, and every 1 minute is used as a data sample point to analyze the error of the predicted parameter values of charging piles in different classified data. The results of the analysis of the predicted and real values of the data of the smart charging pile are shown in Fig. 4, and (a)-(d) represent the analysis results of the working voltage, the difference between the actual current and the demand current, the number of times of touching the emergency stop button, and the difference between the temperature of the pile body and the room temperature in the work of the smart charging pile, respectively. Obviously, the charging pile in normal operating conditions, the prediction results of each categorized feature parameter are more accurate, and the prediction error is kept between 0.44% and 10.52%. It is verified that the big data classification and prediction model based on deep learning has the ability of describing the trend change of each characteristic parameter of the charging pile under normal operating conditions.

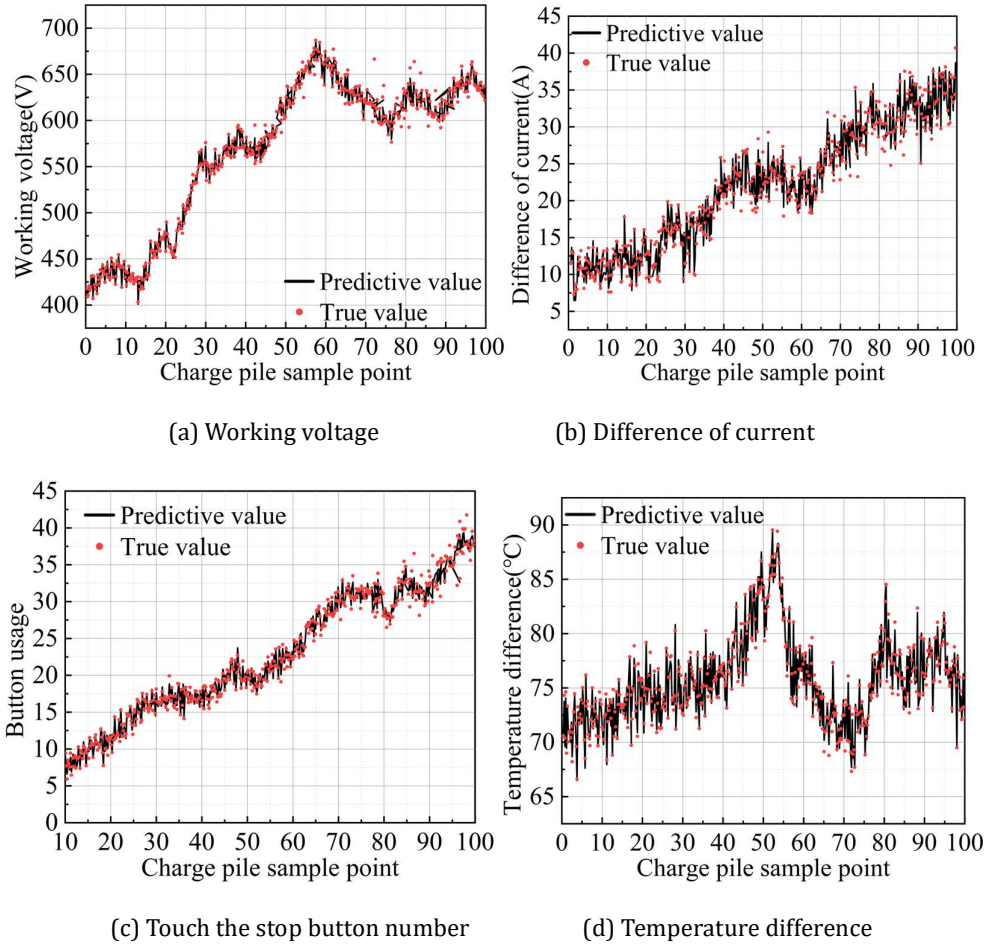


(a) Working voltage (b) Difference of current

(c) Touch the stop button number (d) Temperature difference

Fig. 4. The intelligent charging pile classification prediction results

2) Analysis of the effect of intelligent charging pile fault warning. According to the fault record of the background management system of the smart charging pile, the data of a charging pile fault warning issued at 3:18 a.m. on May 16, 2023 are extracted, and the analysis results of the curve of the prediction results of each classification feature data 65 minutes before the smart charging pile sends out the warning are shown in Figure 5, and (a)-(d) represent the analysis results of the working voltage, current difference, number of times the emergency stop button is touched and the temperature difference value in the work of the smart charging pile, respectively. At this time, the error between the predicted value and the actual value of each categorized feature parameter shows an increasing trend over time, and when the smart charging pile operates under abnormal conditions, its average relative error reaches about 10%, and the maximum relative error reaches 15.42%. This is due to the fact that the classification and prediction model is trained using the charging pile data when there is no fault, and when the charging pile operates under normal conditions, the model prediction value can fit the monitoring value of each characteristic index well. When the charging pile is abnormal, the prediction accuracy of the classification and prediction model will be slightly decreased. However, the error value of this paper can still be controlled within 16%, compared with the existing model has a strong application advantage, can effectively monitor the failure of the smart charging pile and make early warning.



(a) Working voltage (b) Difference of current

(c) Touch the stop button number (d) Temperature difference

Fig. 5. The prediction results of the charging pile failure

4. Conclusion

This paper combines the butterfly optimization algorithm and OGRU model to establish a big data classification module, and constructs a big data prediction model based on deep learning, and builds a big data classification and prediction model after optimizing each parameter of the model. The application effect of the model is analyzed through simulation tests and empirical analysis experiments, and the results show that:

1) The efficient deep learning algorithm adapted to big data classification and prediction proposed in this paper has a large improvement in the accuracy performance F_β (0.9751-0.9857) of classifying industrial time-series data compared to the BPNN algorithm (0.9123-0.9652) used in the industry, and meets the efficiency requirements of big data classification.

2) When classifying and predicting the electricity load of the distribution network, the model error ranges from 0.00% to 8.72%, and the average error rate is only 1.48%, indicating that the model has a better effect on the overall classification and prediction of the big data of the electricity load of the distribution network. In addition, summarizing the load prediction for normal weekdays and heavy load days, it can be seen that the results of the load prediction model can reach the desired values, which has certain application value. It is also found that the model is more accurate in classifying and predicting the data related to the operation of normal and faulty smart charging piles, and the prediction error is kept between 0.44% and 15.42%.

3) The big data classification and prediction model proposed in this study has stable performance and high accuracy, which greatly improves the accuracy of data prediction in various scenarios, and is

of great value for the development of systems such as automated scheduling of distribution grids and industrial dispatch automation.

References

[1] Akhtar, N., Parwej, F., & Perwej, Y. (2017). A perusal of big data classification and hadoop technology. International Transaction of Electrical and Computer Engineers System (ITECES), USA, 4(1), 26-38.

[2] Torres, J. F., Hadjout, D., Sebaa, A., Martínez-Álvarez, F., & Troncoso, A. (2021). Deep learning for time series forecasting: a survey. Big Data, 9(1), 3-21.

[3] Dubey, A. K., Kumar, A., & Agrawal, R. (2021). An efficient ACO-PSO-based framework for data classification and preprocessing in big data. Evolutionary Intelligence, 14, 909-922.

[4] Torres, J. F., Galicia, A., Troncoso, A., & Martínez-Álvarez, F. (2018). A scalable approach based on deep learning for big data time series forecasting. Integrated Computer-Aided Engineering, 25(4), 335-348.

[5] Abukhodair, F., Alsaggaf, W., Jamal, A. T., Abdel-Khalek, S., & Mansour, R. F. (2021). An intelligent metaheuristic binary pigeon optimization-based feature selection and big data classification in a MapReduce environment. Mathematics, 9(20), 2627.

[6] Galicia, A., Talavera-Llames, R., Troncoso, A., Koprinska, I., & Martínez-Álvarez, F. (2019). Multi-step forecasting for big data time series based on ensemble learning. Knowledge-Based Systems, 163, 830-841.

[7] Elkano, M., Galar, M., Sanz, J., & Bustince, H. (2018). CHI-BD: A fuzzy rule-based classification system for Big Data classification problems. Fuzzy Sets and Systems, 348, 75-101.

[8] Ducange, P., Fazzolari, M., & Marcelloni, F. (2020). An overview of recent distributed algorithms for learning fuzzy models in Big Data classification. Journal of Big Data, 7(1), 19.

[9] Devi, S. G., & Sabrigiriraj, M. (2018, March). Feature selection, online feature selection techniques for big data classification:-a review. In 2018 International Conference on Current Trends towards Converging Technologies (ICCTCT) (pp. 1-9). IEEE.

[10] Dagdia, Z. C. (2019). A scalable and distributed dendritic cell algorithm for big data classification. Swarm and Evolutionary Computation, 50, 100432.

[11] Fathi, M., Haghi Kashani, M., Jameii, S. M., & Mahdipour, E. (2022). Big data analytics in weather forecasting: A systematic review. Archives of Computational Methods in Engineering, 29(2), 1247-1275.

[12] Bok, B., Caratelli, D., Giannone, D., Sbordone, A. M., & Tambalotti, A. (2018). Macroeconomic nowcasting and forecasting with big data. Annual Review of Economics, 10(1), 615-643.

[13] Seyedan, M., & Mafakheri, F. (2020). Predictive big data analytics for supply chain demand forecasting: methods, applications, and research opportunities. Journal of Big Data, 7(1), 53.

[14] Wang, K., Xu, C., Zhang, Y., Guo, S., & Zomaya, A. Y. (2017). Robust big data analytics for electricity price forecasting in the smart grid. IEEE Transactions on Big Data, 5(1), 34-45.

[15] Meysam Salehi & Shahrbanoo Ghahari. (2024). Classification of domestic violence Persian textual content in social media based on topic modeling and ensemble learning. Heliyon(22),e39953-e39953.

[16] Lanping Chen,Sichao Sun,Lei Xu,Fan Yang,Chenyuan Chen & Yue Zhu. (2024). A novel bi-level optimal strategy-assisted design boosting discovery of specific surface area of perovskite oxide for effective photocatalysis. Inorganic Chemistry Communications(P3),113393-113393.

[17] Shi Heng,Ma Wen,Xu ZhenHao & Lin Peng. (2023). A novel integrated strategy of easy pruning, parameter searching, and re-parameterization for lightweight intelligent lithology identification. Expert Systems With Applications.

[18] Tingze Long,Han Yi,Yatong Kang,Ying Qiao,Ying Guan & Chao Chen. (2024). Study on bionics-based swarm intelligence optimization algorithms for wavelength selection in near-infrared spectroscopy. Infrared Physics and Technology105594-105594.

[19] Renjith V. S. & Jose P. Subha Hency. (2023). Early-Stage Detection and Classification of Breast Neoplasm Stages Using OGRU-LSTM-BiRNN and Multivariate Data Analysis. Journal of The Institution of Engineers (India): Series B(3),659-678.

[20] Zhonghao Chang,Shuangcheng Sun,Lin Li & Linyang Wei. (2024). Reconstruction of infrared absorption and scattering coefficient distributions of semitransparent medium using Adam algorithm combined with adjoint model. Infrared Physics and Technology105481-105481.