OPTIMISATION OF POWER SYSTEM DISPATCH CONTROL STRATEGY BASED ON IMPROVED DEEP FOREST MODEL

Cheng Yang,* Yanwei Wang,* Kuiye Chen,* Xinwei Wang,* and Zhennan Yang*

Abstract

With the increasing application scope of the current power system, people's attention to power scheduling and control strategies in the power system is constantly increasing. At present, there are problems with delayed response and low scheduling accuracy in the power dispatch and control model scheduling of the power system. To address this issue, this study utilises manual sampling algorithms to improve the deep forest algorithm and proposes a novel improved algorithm. By using this algorithm to construct a power dispatch control model, we aim to improve the accuracy and response speed of power dispatch. Comparative experiments on improved algorithms showed that the classification accuracy of this algorithm was 97.7% and the classification time was only 1.8 s. After testing the constructed model, it was found that the response accuracy of the model was as high as 94.4%. The power dispatch control strategy proposed based on this model could control the power stability in the power system and reduce the dispatch cost by 78.9%. The above results indicate that the proposed improved model can improve the accuracy and response time of power dispatching.

Key Words

Power system, scheduling control model, manual sampling algorithm, deep forest algorithm

1. Introduction

With the improvement of the economic level, many regions attach increasing importance to the accuracy and rationality of power system dispatch control (PSDC) strategies [1], [2]. Currently, many scholars have researched the PSDC strategies [3], [4]. For example, Khaloie *et al.* designed a day-ahead and inter-day scheduling model to address the problem of low power generation efficiency and limited dispatchable electricity in current power plants. Comparison with other models showed that it could improve the scheduling efficiency of power plants [5]. To optimise off-grid hybrid microgrid systems with different load scheduling strategies, Ishraque *et al.* designed a PSDC strategy based on load tracking and predictive scheduling. This strategy was used for detection in practical situations and could increase the effect time of the power grid system to 78.6% of the original [6]. Li *et al.* proposed a PSDC scheme built on a continuous time distribution algorithm to address the issue of low power generation efficiency in biomass power plants. The comparison showed that this strategy could increase the power generation efficiency of the power plant by 67.8% [7]. However, the above-mentioned PSDC strategy still faces problems of long-time consumption, high cost, and delayed response [8]. Therefore, proposing a scheduling control model that can improve the speed of model effects, reduce scheduling costs, and enhance the efficiency of power utilisation in the power system is an urgent problem to be solved.

The synthetic minority oversampling technique (SMOTE) is widely used in various models due to its powerful data processing capabilities [9], [10]. For example, Dablain *et al.* constructed a new model based on the SMOTE algorithm to handle the issue of data imbalance in current machine-learning models. Compared with traditional models, this model has improved data balancing performance by 76.5% [11]. The multi-grained cascade forest (gcForest) algorithm was widely used due to its excellent learning representation ability [12], [13]. For example, to solve the problems of high complexity and long computation time in current deep neural networks (DNNs), Chen built an improved deep learning algorithm based on the gcForest. This algorithm reduced the computational complexity of DNN by 87.9% [14]. Ma *et al.* proposed a hash filtering mechanism based on the gcForest to deal with the problems of long computation time and high cost in current distributed forest systems. This mechanism was used for detection in practical situations, and it was found that the computational time of the system was reduced by 67.6% [15].

In summary, the current PSDC model cannot meet the expected requirements, and there are still problems such as inaccurate power scheduling and untimely response time in power dispatching. The improved gcForest based on SMOTE can improve computation speed and accuracy.

* Guangdong Power Dispatching Control Center, 510000, China; e-mail: Yang11134@126.com; Wang4_444@126.com; Chen_112345@126.com; Ww5_200@163.com; Yang_20244@126.com
Corresponding author: Cheng Yang

(DOI: 10.2316/J.2025.203-0619)

Therefore, this study constructs a PSDC model grounded on this algorithm to improve the response time and scheduling accuracy of the model. The innovation lies in the combination of SMOTE and gcForest algorithms, using SMOTE to solve the overfitting phenomenon in gcForest and improve the accuracy of the algorithm.

2. Methods and Materials

2.1 Improved gcForest Model Based on SMOTE Algorithm

The innovative development of society has gradually attracted people's attention to PSDC, but traditional PSDC systems are no longer able to meet the power dispatching needs of large-scale power grids [16]. Therefore, it is necessary to optimise the traditional PSDC model to improve its rationality and timeliness. The gcForest algorithm is a machine learning algorithm that constructs an integrated deep model based on decision trees to achieve deep extraction of data [17]. This study applies the algorithm to the PSDC model to improve the control accuracy of the control model and the rationality of power scheduling. Figure 1 is the flowchart of the gcForest model.

In Fig. 1, the data extraction process of the gcForest model based on the deep forest is divided into four steps: multi-granularity scanning (MGS), random forest training (RFT), feature extraction, and feature fusion and classification. In MGS, samples are slide sampled by setting sliding windows of different sizes to obtain feature sub-vectors of multiple dimensions. The obtained eigenvectors are used in the RTF model, and each random forest obtains a probability vector with the same number of categories through the RTF model. These probability vectors can reflect the likelihood of each category. Feature extraction is to combine these probability vectors to form a representation vector matrix, and then use sliding windows of different sizes to perform inverse sampling on the formed vector matrix rows to achieve true feature extraction. Finally, the extracted feature vectors are fused and classified. All the extraction results of random forests are concatenated together for output, and finally, the output results are analysed. MGS can be segmented into multiple different stages. First, the attribute features of the raw data are scanned for the first time by sliding the window. Second, the attribute features extracted during the initial scan form a new dataset and are used as a new sample. These newly formed samples are trained to obtain different forest types. Each forest type corresponds to the distribution vectors of different categories of the sample data. Finally, all obtained vectors are combined, and the combined samples are used as inputs for subsequent operations. These steps are beneficial for extracting more accurate feature information. When performing RTF, a cascaded forest structure is used to further improve the accuracy of data extraction. Cascade forest is composed of multiple decision tree forests, with each forest containing one hyperparameter of decision trees. The formal expression of cascading forests is

$$f_t(x) = H_t[X, f_{t-1}(x)] \tag{1}$$

In (1), $f_t(x)$ is the output class distribution vector of the t-th decision tree. t is the number of layers of the decision tree. H_t represents the class distribution vector of the total decision level output. x is the input value. $[X, f_{t-1}(x)]$ is the distribution vector output from the previous layer. The expression of the binary cascade forest model is shown in (2)

$$g(x) = argmax[f_T(x)]_c \tag{2}$$

$[f_T(x)]_c$ is the c-th term of the class distribution vector $f_T(x)$. The calculation of the binary cascade forest model is shown in (3)

$$g(x) = argmax[f_t(x)]_c \tag{3}$$

This model is highly sensitive to noise and other data, which can easily lead to overfitting [18]. Given this, it is necessary to optimise the gcForest algorithm in the model to meet practical needs. The SMOTE algorithm can solve the phenomenon of data imbalance in classification problems, and its working principle is shown in Fig. 2.

In Fig. 2, the principle is to balance the dataset by synthesising new minority class samples, thereby improving model performance and reducing model overfitting. After receiving data, the algorithm first selects a sample point from the minority class samples and then calculates the distance from that sample point to other sample points in the minority class samples to find the closest sample point. One or more samples are randomly selected from the found sample points, connecting the selected sample points with the previous ones, treating them as a new sample, and adding them to the original dataset. By doing so, the sample size of minority classes can be increased, maintaining data balance and preventing overfitting. In this process, the formula for sample generation is shown in (4)

$$r = X + g \cdot d \tag{4}$$

In (4), X is the selected sample point, g is the distance between two random points, and d represents the distance between the two selected samples. The formula for sample synthesis is shown in (5)

$$y_n = y_i + r(0, 1) \cdot (y_n - y_i) \tag{5}$$

In (5), y_i and y_n are two randomly selected samples. $r(0, 1)$ is a random number between 0 and 1. To improve the overfitting phenomenon in the gcForest algorithm, this study utilises the SMOTE algorithm to enhance it. First, the data are processed using SMOTE to balance the minority class sample data, and then the processed data are input into the gcForest model for data analysis. The improved gcForest algorithm is shown in Fig. 3.

In Fig. 3, after inputting the sample data, the data are first input into the SMOTE module for data preprocessing. Then, the module synthesises a new dataset by adding minority class sample data. The synthesised new dataset is input into the gcForest module, and the features in the new dataset are accurately extracted through four steps: MGS, RTF, feature extraction, and feature fusion. Finally,

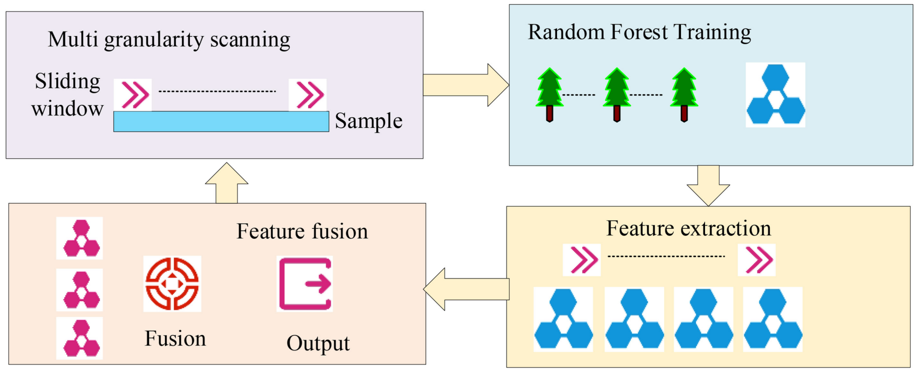


Figure 1. Flowchart of the deep forest model.

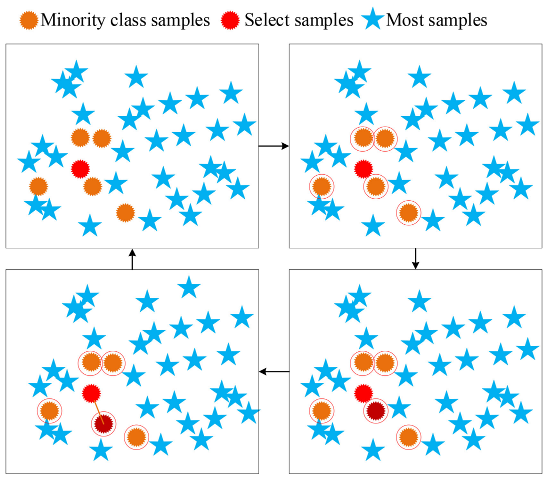


Figure 2. Principle of the SMOTE algorithm.

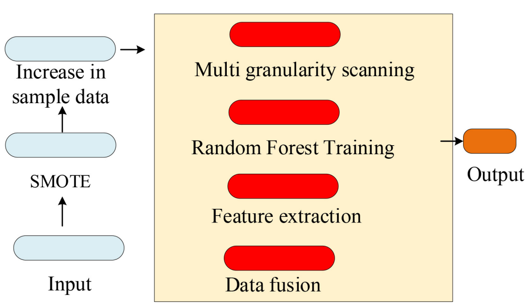


Figure 3. Improved deep forest algorithm.

the extracted feature information is output for subsequent analysis.

2.2 PSDC Model Combining SMOTE and gcForest

After SMOTE optimisation, the gcForest algorithm is constructed into a control model to improve the control accuracy of power system scheduling. Figure 4 shows the basic framework of the SMOTR-gcForest control model.

In Fig. 4, the model includes a data input layer (L1), a data preprocessing layer (L2), a data computation layer (L3), and a data result analysis layer (L4). L1 receives sample data and standardises the data format for subsequent calculations. Then, the data are input into L2 for preprocessing operations. This layer will have a SMOTE module that expands individual data from the initial data to form a new dataset, ensuring data diversity and preventing overfitting. The newly synthesised dataset is transmitted to L3, and the feature information of the newly synthesised dataset is extracted through the MGS, RTF, feature extraction, and feature fusion steps of the gcForest module in this layer. Finally, the extracted feature information is input into L4 for processing and classification. This model can accurately classify most data information. This model is applied to the DC power system and proposes corresponding power scheduling strategies based on its output data to improve the accuracy and rationality of the DC power system scheduling strategies. The PSDC model based on SMOTE-gcForest is shown in Fig. 5.

In Fig. 5, the model first collects various data from the power dispatch system, inputs the collected data into the control model, and uniformly formats the collected data. Then, using the SMOTE algorithm, the data are preprocessed to prevent overfitting and form a new dataset for the power dispatch system. Using the gcForest model, the newly formed dataset is subjected to feature extraction, and then the extracted data is analysed. Based on the analysis results, it is determined that there are problems with the scheduling strategy of the power system. If there are any problems, optimisation is based on the analysis results to ensure the stable operation of the power system. The stable operation of the power system also requires a judgement of the power output. When collecting data from the power system, it is necessary to consider the operating cost, power index, and constraints of power output. The operating cost of wind power is shown in (6)

$$P = \sum_{j=1}^{M}\sum_{i=1}^{n}[\alpha_i J_{i,j} + \alpha_2(\widehat{J}_{i,j} - J_{i,j}) + \alpha_3(\widehat{J}_{i,j} - J_{i,j})] \quad (6)$$

In (6), i is the i-th wind turbine in the wind turbine unit, j is the j-th scheduling cycle of the wind turbine, J is the output power of the wind turbine, n is the number of wind turbines in a wind turbine, M is the number of operating cycles during wind turbine scheduling, and α_i, α_2 and α_3 are direct, overestimated, and underestimated cost coefficients for wind energy. When wind turbines generate

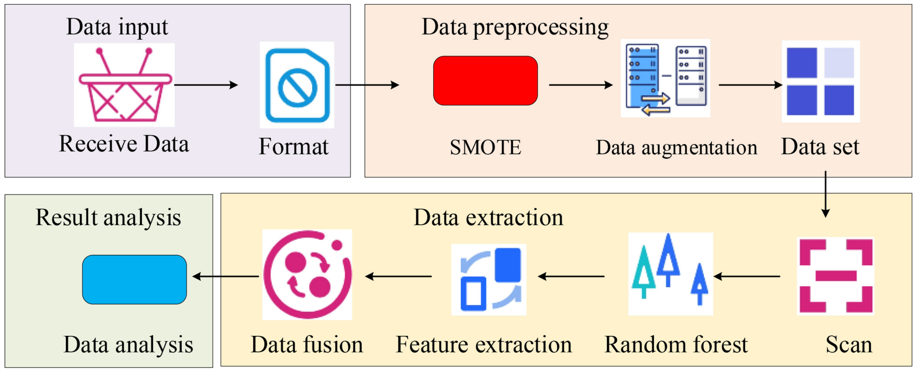


Figure 4. The SMOTR-gcForest model.

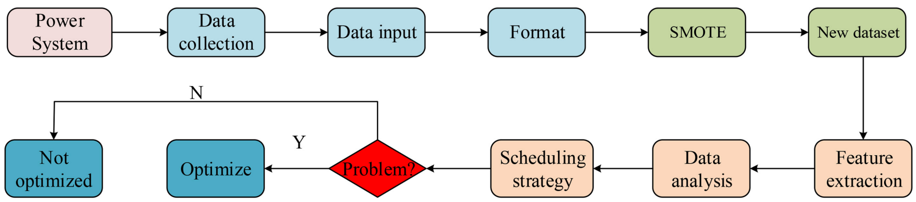


Figure 5. Power dispatching control model of SMOTE-gcForest.

limited power, they need to control the output power through pitch angle. This increases the regulation pressure and affects the stability of the power plant, so the power limit of each unit should be minimised to ensure stable operation of the system. The power limit state formula of the system is shown in (7)

$$P_1 = \frac{1}{a \times b} \sum_{j=1}^{M} \sum_{i=1}^{n} \frac{{J_{i,j}}^{\max} - J_{i,j}}{{J_{i,j}}^{\max} - {J_{i,j}}^{\min}} \tag{7}$$

In (7), ${J_{i,j}}^{\max}$ and ${J_{i,j}}^{\min}$ are the maximum and minimum output power of the wind turbine. In any period, the relationship between the dispatching command in the wind turbine and the output power of the wind turbine is shown in (8)

$$\sum_{i=1}^{n} J_i = J_p + J_l \tag{8}$$

In (8), J_p represents the dispatch instruction power during the wind farm dispatch cycle, p is the output power of the wind turbine, and J_l is the power loss during the operation of wind turbines. Any wind turbine unit has an extreme value, which is the maximum and minimum output capacity. During the operation of a wind turbine, its output capacity should not exceed its own extreme value to avoid damaging the structure of the wind turbine and reducing its service life. The boundary expression of unit power is shown in (9)

$${J_{i,j}}^{\min} \leq J_{i,j} \leq {J_{i,j}}^{\max} \tag{9}$$

In (9), $J_{i,j}$ is the actual output power of the wind turbine during actual operation. The output capacity of wind turbines during operation is also affected by other factors. A wind turbine cannot increase or decrease its output power without standards, and it is necessary to limit the output power of the wind turbine. The calculation of limiting the output power of wind turbines is shown in (10)

$$-r_a \Delta t \leq J_{i,j+1} - J_{i,j} \leq r_m \Delta t \tag{10}$$

In (10), r_a and r_m are the maximum values of the downhill speed and uphill speed of the wind turbine during uphill operation and Δt is the cycle of scheduling instructions for wind turbines. The inequality in (10) also requires the addition of constraints to accurately calculate the inequality. The general penalty factor is used to constrain inequalities. The formula after adding penalty factor constraints is shown in (11)

$$K = \varpi \sum_{j=1}^{T-1} \sum_{i=1}^{n} [min(0, ({J_i}^u - |J_{i,j+1} - J_{i,j}|))] \tag{11}$$

In (11), ${J_i}^u$ is the maximum climbing amount of the wind turbine, ϖ represents the penalty factor, and when this value is set to 0 or infinite, and ϖ is used to constrain the climbing rate of the wind turbine. If the climbing amount of the wind turbine during operation does not reach the constraint condition, constraint condition K will take a maximum value to adjust and constrain the climbing rate of the wind turbine. By outputting the results of the model, the operating cost, power index, and constraint conditions of the power system are judged, and based on these results, the scheduling strategy of the power system is optimised.

Table 1
Basic Experimental Environment Table

Environmental Form	Project	Specification
Hardware environment	CPU model	Intel Core i9
	CPU dominant frequency	2.30GHz The biggest rui frequency4.60GHz
	GPU model	NVIDIA GeForce RTX
	Memory size	64GB DDR4 3200MHz
	Memory; Storage	1TB NVMe SSD
Software environment	OS	Windows 10
	Programming language	Python 3.8.10
	Python environment	Anaconda 3
	Version control	Git 2.30.0

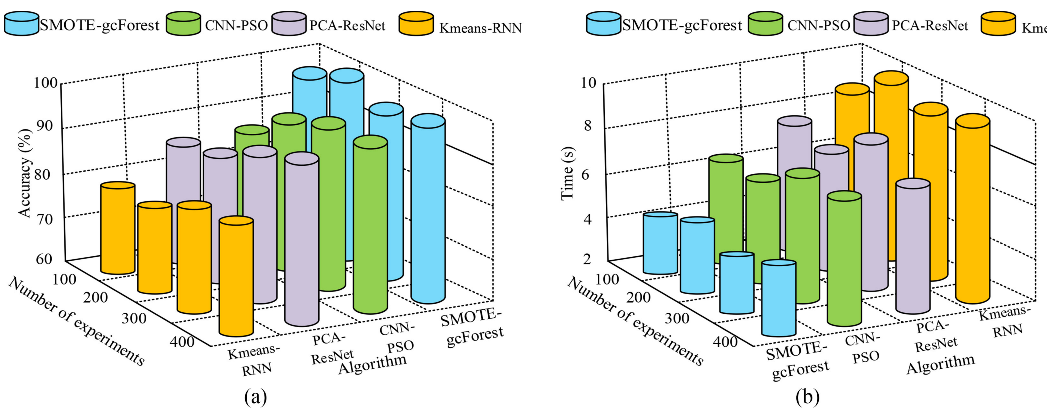


Figure 6. The algorithm classification accuracy and classification speed: (a) Algorithm accuracy and (b) Algorithm classification time consumption.

3. Results

3.1 Performance Analysis of SMOTE-gcForest Algorithm

To verify the superiority of SMOTE-gcForest algorithm in classification, this study conducts comparative experiments with convolutional neural networks - particle swarm optimisation (CNN-PSO), principal component analysis - residual network (PCA-ResNet), and K-Means clustering algorithm - recurrent neural network (Kmeans-RNN). Table 1 shows the experimental setup.

Based on the above configuration, the experiment selects 500 data points from the digital image dataset MNIST to compare four algorithms. Figure 6 shows the comparison results of data classification accuracy and speed among four algorithms.

In Fig. 6(a), the classification accuracy of SMOTE-gcForest algorithm is 97.7%, which is the highest among the four algorithms. The classification accuracies of CNN-PSO, PCA-ResNet, and Kmeans-RNN are 87.8%, 81.2%, and 76.4%, respectively. In Fig. 6(b), the four algorithms took 1.8 s, 5.6 s, 7.3 s, and 8.4 s to classify the raw data. This indicates that the research algorithm has the highest classification accuracy and the fastest classification speed. F1 value is the harmonic average of algorithm accuracy and recall rate, which is used to evaluate the classification performance. The higher the value, the better the classification performance of the algorithm. Figure 7 compares the F1 values and loss function values of various algorithms.

In Fig. 7, the SMOTE-gcForest algorithm has the highest F1 value and the lowest loss function value, indicating the lowest algorithm performance. The F1 value of this algorithm is 0.95, and the loss function value is 0.02. The F1 value of CNN-PSO is 0.81, and the loss function value is 0.04. The loss function value of PCA-ResNet is lower than the first two algorithms, only 0.72, but the loss function value is higher than the first two algorithms, at 0.12. The F1 value of Kmeans-RNN is the lowest among the four algorithms, only 0.68, but its loss function value is the highest, reaching 0.16. Figure 8 compares the error values and spatial complexity of the algorithm.

In Fig. 8(a), the average error rate of SMOTE-gcForest is 1.2%, and its error rate variation is relatively stable with low fluctuation amplitude. The error rate of CNN-PSO fluctuates greatly, with an average error rate of 3.6%. The error rates of PCA-ResNet and Kmeans-RNN are 4.5% and

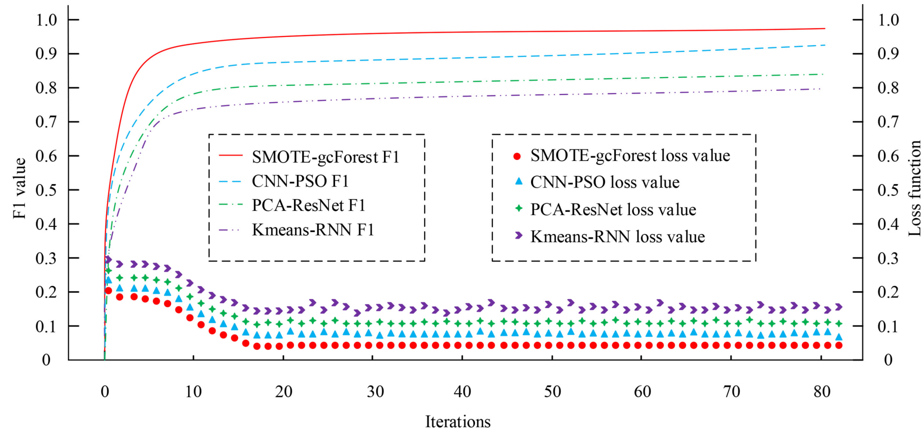


Figure 7. F1 values and loss function values of the four algorithms.

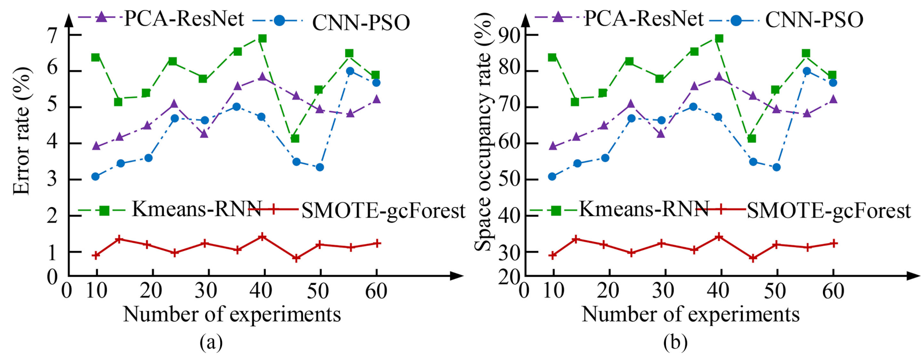


Figure 8. Error values and spatial complexity contrasts: (a) Error value and (b) Space complexity.

5.8%, respectively. In Fig. 8(b), SMOTE-gcForest, CNN-PSO, PCA-ResNet, and Kmeans-RNN have computer space occupancy rates of 28.7%, 52.3%, 62.6%, and 72.4%, respectively, during computation. Therefore, the research algorithm has the highest classification accuracy, the shortest classification time, the highest F1 value, and the lowest loss function value. The comprehensive performance of this algorithm is far superior to the comparative algorithms, so this study uses this algorithm to construct a PSDC model to improve the scheduling accuracy and response speed of the model.

3.2 Empirical Analysis of PSDC Model Based on SMOTE-gcForest Algorithm

After verifying the superiority of the SMOTE-gcForest algorithm, the control model based on this algorithm is analysed. A comparison is made between the PSDC model based on SMOTE-gcForest algorithm, the widely used PSDC model based on improved ant colony Optimisation (ACO) algorithm, and the traditional PSDC model. Figure 9 compares the response time and response accuracy of three models.

In Fig. 9(a), the response accuracy of SMOTE-gcForest reaches a maximum of 94.4%. The response accuracy of ACO is lower than that of SMOTE-gcForest, only 61.9%. The response accuracy of traditional models is the lowest, only 45.3%. In Fig. 9(b), the response time of

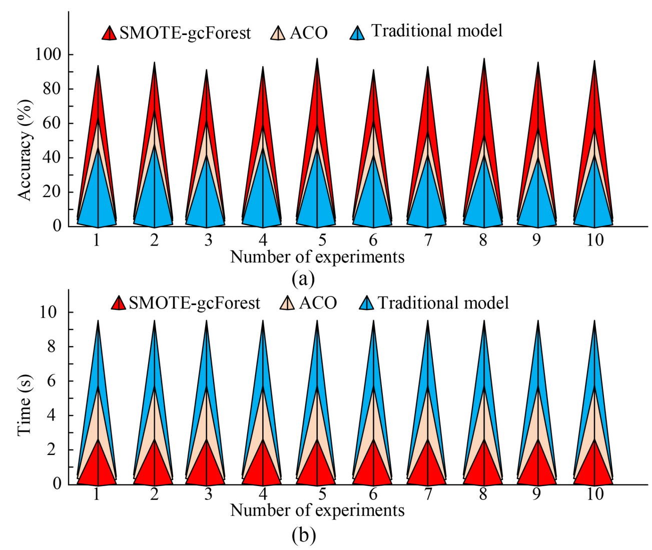
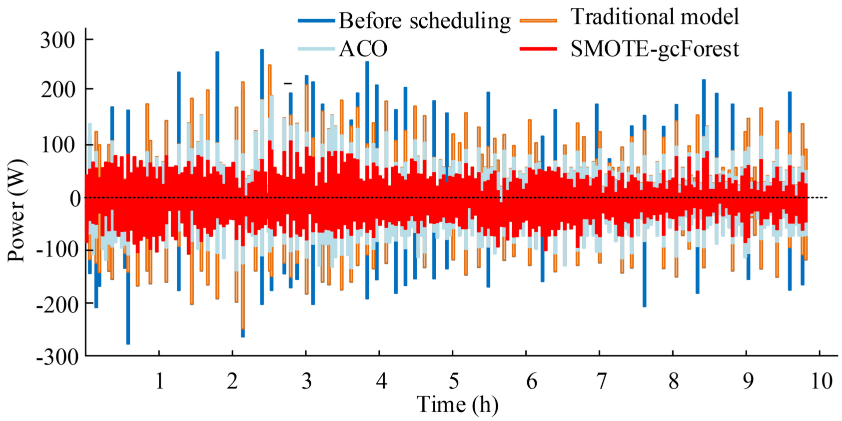


Figure 9. A comparison of model response time and response accuracy: (a) Accuracy rate and (b) Response time.

the traditional power dispatch model is 9.4 s, the average time of ACO is 5.7 s, while SMOTE-gcForest has the fastest response speed, with a response time of only 1.8 s. Figure 10 shows the changes in electrical power of the power system after controlling the power dispatch strategies proposed based on three models.



The changes in electrical power before and after control

Figure 10. The electric power of a power system.

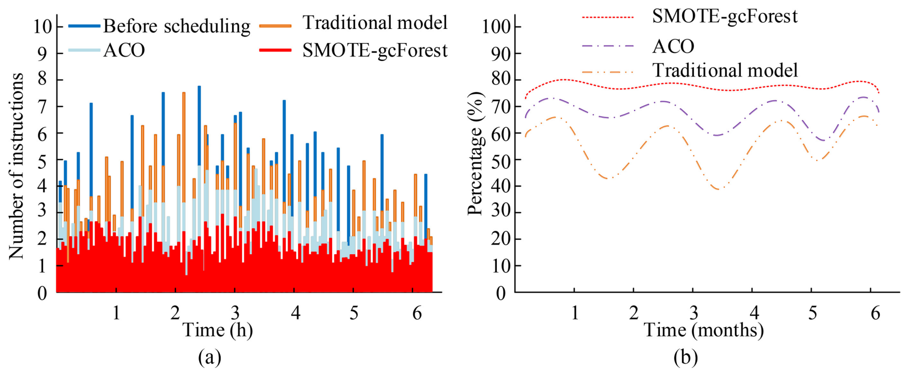


Figure 11. Number of orders and cost of power dispatching strategy based on three models: (a) Changes in the number of instructins in the power system and (b) Cost reduction of power system dispatching.

In Fig. 10, before power dispatch, the power variation of the power system is extremely unstable, with a large fluctuation range. After proposing power dispatch strategies based on three models, the changes in electric power are relatively stable, and the magnitude of the changes is approaching zero. Among the three models, SMOTE-gcForest has the largest variation and the strongest stability, with its electrical power fluctuating between -59 W and 70 W. The electric power variation range of the ACO model is between -100 W and 120 W. The variation of electric power in traditional models is very similar to that before scheduling, with electric power ranging from -200 W to 200 W. Figure 11 shows the comparison of power dispatch costs and command quantities based on three proposed models for power dispatch strategies.

In Fig. 11(a), before the proposed dispatching strategy, when the power system conducts power dispatching, the number of instructions issued per hour is about 7. The strategy proposed based on traditional models has reduced the number of instructions issued to 6 times per hour, with very low variation, which is the same as the number before scheduling. The ACO model significantly reduces the number of instructions issued by the power system, but the reduction is not as significant as SMOTE-gcForest. The research model reduces the number of instructions issued to 2 times per hour, greatly reducing operational complexity. In Fig. 11(b), the PSDC strategy proposed based on three models can significantly reduce the cost of power dispatch. Among them, the SMOTE-gcForest model reduces power dispatch costs the most, with an average of 78.9%, and the frequency of reduction is relatively stable. The ACO model performs slightly worse, with a cost reduction of only 69.5%. The traditional model has a minimum reduction of only 54.3%, and the cost reduction of this model is extremely unstable, making it difficult to estimate the cost. The above results are summarised to more intuitively observe the performance of SMOTE-gcForest model. The results are shown in Table 2.

Table 2 shows that the performance of SMOTE-gcForest power dispatching model is superior to other methods.

4. Discussion

To test the scheduling accuracy, response time, and cost of the proposed SMOTE-gcForest model, a comparative experiment was conducted on the SMOTE-gcForest algorithm. The comparative experiments of SMOTE-gcForest, CNN-PSO, PCA-ResNet, and K-means-RNN showed that SMOTE-gcForest had the highest classification accuracy (97.7%), while the classification accuracy of the other

Table 2
Performance Analysis of SMOTE-gcForest Model

Model	SMOTE-gcForest	ACO	Traditional Model
Response time	1.8 s	5.7 s	9.4 s
Response accuracy	94.4%	61.9%	45.3%
Electric power variation range	-59~70 W	-100~120 W	-200~200 W
Number of instructions	2 times/h	4 times/h	6 times/h
Dispatch cost reduction	78.9%	69.5%	54.3%

three algorithms was 87.8%, 81.2%, and 76.4%. Among the four algorithms, SMOTE-gcForest had the shortest classification time, while Kmeans-RNN had the longest classification time. This result was similar to Guan *et al.*'s findings. The reason for this result might be that the SMOTE algorithm had balanced the original data reasonably, making it more rational and reducing the overfitting of the data, thus improving the classification accuracy of the algorithm [19]. Among the comparison results of F1 values and loss function values of the four algorithms, SMOTE-gcForest had the highest F1 value of 0.95, the lowest loss function value of 0.02, and the lowest F1 value. The Kmeans-RNN algorithm had the highest loss function value. This result was roughly similar to that of Pradipta *et al.*, who achieved a loss function value of 0.1 for the gcForest algorithm in their experiment. The reason might be that Pradipta did not perform data preprocessing during the experiment, resulting in a low F1 value of the algorithm during the experiment [20]. The actual application effects of PSDC strategies based on SMOTE-gcForest, ACO, and traditional models were compared. After using the SMOTE-gcForest strategy for scheduling, the changes in electrical power in the power system became more stable. The stability of the ACO-based strategy was lower than that of SMOTE-gcForest, while the system based on traditional models was even more unstable. This result was similar to the conclusion of Zhang *et al.* [21]. When comparing the number of commands and scheduling costs in the power system, SMOTE-gcForest reduced the number of commands to 2 per hour, the traditional model had 6 commands per hour, and the ACO model reduced it to 3 commands per hour. Moreover, the SMOTE-gcForest model could reduce scheduling costs by 78.9%. This result is correlated with the study by Liu *et al.* [22].

The above results indicate that the proposed SMOTE-gcForest model can optimise the PSDC strategy, reduce scheduling costs, and improve the efficiency of power scheduling.

5. Conclusion

In response to the current shortcomings of delayed scheduling and slow scheduling speed in PSDC strategies, this study proposed the SMOTE-cForest algorithm that integrated SMOTE and gcForest and constructed a PSDC model based on this algorithm. To verify the superiority of SMOTE-gcForest, it was first compared with CNN-PSO, PCA-ResNet, and Kmeans-RNN. In the experiment, SMOTE-gcForest had the lowest loss function value, the lowest error rate, and the best overall performance. In the comparison based on research models, ACO models, and traditional models, SMOTE-gcForest had the best response accuracy and speed, and the lowest scheduling cost. The above experiments indicate that the SMOTE-gcForest model can optimise the PSDC strategy and improve scheduling efficiency. However, the experiments in this study were conducted under ideal conditions, and it remains to be verified whether the model can optimiwe power dispatch strategies in case of emergencies.

References

[1] Moreno-Castro J, Ocaña Guevara V S, León Viltre L T, Gallego Landera Y, Cuaresma Zevallos O. Microgrid management strategies for economic dispatch of electricity using model predictive control techniques: A review. Energies, 2023, 16(16): 5935-5948.

[2] Guo W.H. Chain Fault Identification and Power Grid Planning Optimisation in Power Systems Considering Multiple Scenarios, International Journal of Power and Energy Systems, 2024, 203: 1-14.

[3] Ahmed T., Ahmed A. L., and Souheil S. A New Adaptive Under-Frequency Load Shedding Scheme for Multi-Area Power System, International Journal of Power and Energy Systems, 2023, 203: 1-10.

[4] Alonso A M, Matute G, Yusta J M, Coosemans T. Multi-state optimal power dispatch model for power-to-power systems in off-grid hybrid energy systems: A case study in Spain. International journal of hydrogen energy, 2024, 52(5): 1045-1061.

[5] Khaloie H, Vallée F, Lai C S, Toubeau J F, Hatziargyriou N D. Day-ahead and intraday dispatch of an integrated biomass-concentrated solar system: A multi-objective risk-controlling approach. IEEE Transactions on Power Systems, 2021, 37(1): 701-714.

[6] Ishraque M F, Shezan S A, Rashid M M, Bhadra A B, Hossain M A, Chakrabortty R K, Das S K. Techno-economic and power system optimisation of a renewable rich islanded microgrid considering different dispatch strategies. IEEE Access, 2021, 9(5): 77325-77340.

[7] Li P, Hu J, Qiu L, Zhao Y, Ghosh B K. A distributed economic dispatch strategy for power–water networks. IEEE Transactions on Control of Network Systems, 2021, 9(1): 356-366.

[8] Tang H, Wang S, Chang K, Guan J. Intra-day Dynamic Optimal Dispatch for Power System Based on Deep Q-Learning. IEEJ transactions on electrical and electronic engineering, 2021, 16(7): 954-964.

[9] Maulidevi N U, Surendro K. SMOTE-LOF for noise identification in imbalanced data classification. Journal of King Saud University-Computer and Information Sciences, 2022, 34(6): 3413-3423.

[10] Wang S, Dai Y, Shen J, Xuan J. Research on expansion and classification of imbalanced data based on SMOTE algorithm. Scientific reports, 2021, 11(1): 24039-24041.

[11] Dablain D, Krawczyk B, Chawla N V. DeepSMOTE: Fusing deep learning and SMOTE for imbalanced data. IEEE Transactions on Neural Networks and Learning Systems, 2022, 34(9): 6390-6404.

[12] Huang J, Chen P, Lu L, Deng Y, Zou Q. WCDForest: a weighted cascade deep forest model toward the classification tasks. Applied Intelligence, 2023, 53(23): 29169-29182.

[13] Chen J, Cui J, Lin C, Ge H. An Intelligent Fault Diagnostic Method Based on 2D-gcForest and L ${} _ {\text {2, p}} $-PCA Under Different Data Distributions. IEEE Transactions on Industrial Informatics, 2022, 18(10): 6652-6662.

[14] Chen Z, Wang T, Cai H, Mondal S K, Sahoo J P. Blb-gcforest: A high-performance distributed deep forest with adaptive sub-forest splitting. IEEE Transactions on Parallel and Distributed Systems, 2021, 33(11): 3141-3152.

[15] Ma P, Wu Y, Li Y, Guo L, Jiang H, Zhu X, Wu X. HW-Forest: Deep forest with hashing screening and window screening. ACM Transactions on Knowledge Discovery from Data (TKDD), 2022, 16(6): 1-24.

[16] Akbari-Dibavar A, Mohammadi-Ivatloo B, Zare K, Khalili T, Bidram A. Economic-emission dispatch problem in power systems with carbon capture power plants. IEEE Transactions on Industry Applications, 2021, 57(4): 3341-3351.

[17] Bi S, Mu L, Liu X. Deep multi-sequence multi-grained cascade forest for tobacco drying condition identification. Drying Technology, 2022, 40(9): 1832-1844.

[18] Liu Y, Liu Z, Luo X, Zhao H. Diagnosis of Parkinson's disease based on SHAP value feature selection. Biocybernetics and Biomedical Engineering, 2022, 42(3): 856-869.

[19] Guan H, Zhang Y, Xian M, Cheng H D, Tang X. SMOTE-WENN: Solving class imbalance and small sample problems by oversampling and distance scaling. Applied Intelligence, 2021, 51(7): 1394-1409.

[20] Pradipta G A, Wardoyo R, Musdholifah A, Sanjaya I N H. Radius-SMOTE: a new oversampling technique of minority samples based on radius distance for learning from imbalanced data. IEEE Access, 2021, 9(6): 74763-74777.

[21] Zhang Z, Wang C, Chen S, Zhao Y, Dong X, Han X. Multitime scale co-optimised dispatch for integrated electricity and natural gas system considering bidirectional interactions and renewable uncertainties. IEEE Transactions on Industry Applications, 2022, 58(4): 5317-5327.

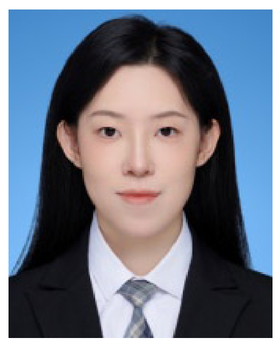
[22] Liu X, Guo Y, Xie H, Tang H, Li B, Wang X, Tong X. Fast machine learning-based extraction of the peak number from ICESat full-waveform data. Remote Sensing Letters, 2021, 12(10): 1004-1014.

Biographies

Cheng Yang received the master's degree from the Institute of Microelectronics, Tsinghua University in 2013. He is currently a Senior Engineer with Guangdong Power Dispatching Control Center. His research interests include areas such as power system operation analysis and stability control.

Yanwei Wang received the master's degree in electrical engineering from the School of Electrical Engineering, Wuhan University in 2012. He is currently a Senior Engineer with Guangdong Power Dispatching Control Center. His research interests include areas such as power system operation analysis and scheduling optimisation.

Kuiye Chen received the bachelor's degree in electrical engineering and automation from North China Electric Power University and the master's degree in electrical engineering from Fuzhou University in 2020. He is currently a Staff of Power Grid Analysis with Guangdong Power Dispatching Control Center. His research interests include areas such as power system operation analysis and scheduling optimisation.



Xinwei Wang received the master's degree in electrical engineering from North China Electric Power University in 2018. She is currently an Engineer with Guangdong Power Dispatching Control Center. Her research interests include areas such as power system operation analysis and stability control.

Zhennan Yang received the master's degree in electrical and electronic engineering from the University of Leicester in 2016. He is currently an Engineer with Guangdong Power Dispatching Control Center. His research interests include areas such as power system operation analysis and stability control.