frontiers | Frontiers in Energy Research

TYPE Original Research
PUBLISHED 19 January 2023
DOI 10.3389/fenrg.2023.1102949

Check for updates

OPEN ACCESS

EDITED BY
Hongming Yang,
Changsha University of Science and Technology, China

REVIEWED BY
Premkumar M,
GMR Institute of Technology, India
Yushuai Li,
University of Oslo, Norway

*CORRESPONDENCE
Yumin Zhang,
ymzhang2019@sdust.edu.cn

SPECIALTY SECTION
This article was submitted to Process and Energy Systems Engineering, a section of the journal Frontiers in Energy Research

RECEIVED 19 November 2022
ACCEPTED 10 January 2023
PUBLISHED 19 January 2023

CITATION
Yu Y, Yang M, Zhang Y, Ye P, Ji X and Li J (2023), Fast reconfiguration method of low-carbon distribution network based on convolutional neural network.
Front. Energy Res. 11:1102949.
doi: 10.3389/fenrg.2023.1102949

Fast reconfiguration method of low-carbon distribution network based on convolutional neural network

Yixiao Yu[1], Ming Yang[1], Yumin Zhang[1]*, Pingfeng Ye[2], Xingquan Ji[3] and Jingrui Li[3]

[1]Key Laboratory of Power System Intelligent Dispatch and Control of Ministry of Education (Shandong University), Jinan, Shandong Province, China, [2]College of Energy Storage Technology, Shandong University of Science and Technology, Qingdao, China, [3]College of Electrical Engineering and Automation, Shandong University of Science and Technology, Qingdao, China

The existing meta-heuristic distribution network reconfiguration (DNR) algorithm has excellent optimization ability through iteration. However, it is difficult to realize the large-scale fast calculation and online real-time response of DNR solution. In order to improve the security and low-carbon economy of distribution network, this paper proposes a fast reconfiguration method of distribution network based on convolutional neural network (CNN). Taking IEEE 33 system and 185 node system as examples, the effectiveness of the proposed method is verified. The reasons why the proposed method can achieve better results are as follows: By mining the historical data of distribution network, the corresponding relationship between load mode (LM) and its optimal topology is established. For a load mode in actual operation, the reconfiguration scheme can be quickly obtained according to the established corresponding relationship. Thus, iterative calculation is avoided and computational efficiency is improved. A multi-branch CNN model is established based on the distribution network structure, and an inception module is introduced into CNN to improve the ability of CNN to extract data features. This model can reduce the dependence on the specific distribution network structure and is easy to expand.

KEYWORDS

convolutional neural network (CNN), rapid reconfiguration of distribution network, load mode, mixed training, data driven

1 Introduction

In the context of global warming, low-carbon operation is one of the trends in distribution system. At present, the distribution network can achieve the purpose of low-carbon operation (Qing et al., 2021) by means of operation optimization such as reconfiguration (Zhan et al., 2020) and multi-energy coupling comprehensive energy system (Zhang et al., 2022; Li Y et al., 2021). Distribution network reconfiguration (DNR) is an important part of distribution management system (DMS) which can improve the operating state of distribution network (DN) (Zhan et al., 2020). In the smart grid environment, the value of power grid operation data cannot be ignored. Big data technology has been practically applied in operation and management practice of distribution network (Zainab et al., 2021). For DNR problem, mining the relationship between the historical operation data of distribution network and the reconfiguration scheme, so as to achieve rapid adaptive reconfiguration, is conductive to improving the economic benefits of power system operation, in line with the mainstream of low-carbon energy development.

In the early stage of DNR development, distribution network structure was relatively simple and power flow direction was single. The reconfiguration analysis methods commonly used mainly included two categories: analytical algorithm based on mathematical optimization (Wang C et al., 2020; Yi et al., 2021). Traditional heuristic algorithm based on simplified model solution of distribution network (Talukdar et al., 2019).

With the access of distributed Generation (DG), bidirectional power flow occurs in DN. The traditional DNR method based on unidirectional power flow is no longer applicable (Li et al., 2020; Yang et al., 2020). Therefore, various intelligent algorithms and their hybrid algorithms are gradually used to solve DNR problems.

The design idea of intelligent algorithm is derived from the simulation of natural process. If there is no time constraint, the algorithm can guarantee the global optimal solution. At present, the intelligent methods commonly used in DNR are mainly various meta-heuristic algorithms, such as particle swarm algorithm (Wu et al., 2021a), genetic algorithm (Jin et al., 2020), and ant colony algorithm (Chen and Dai, 2020). In order to improve algorithm, researchers adopted measures such as compression solution space (Wang J et al., 2020), addition of mutation operation (Pegado et al., 2019) and collaborative solution of multiple algorithms (Huang et al., 2021; Prasad and Sushama, 2022). However, the iterative optimization method of the above algorithms not only gives it strong optimization performance, but also determines that it is difficult to achieve efficient solution of the distribution network reconfiguration scheme.

Real-time operation is an inherent attribute of NN (Neural Network) algorithm, from the proposed neuron M-P model to the establishment of DL (Deep Learning) theory, Neural Network model. In particular, deep learning model has been widely applied in many fields with real-time analysis requirements, such as image annotation (Abrahamyan et al., 2021), speech recognition (Hussain et al., 2021), semantic understanding (DE Oliveira et al., 2020; Xie et al., 2021), and achieved good application effects (Zheng et al., 2021). In power system, deep learning model has also been introduced into measurement data completion (Wang et al., 2021), fault identification and line selection (Wu et al., 2021b), load prediction (Alavi et al., 2021; Li L. et al., 2021; Li et al., 2021a; Li et al., 2021b), and state estimation (Huang et al., 2021).

As early as in the 1990s, some scholars applied Artificial Neural Network (ANN) to solve DNR problems (Zhe et al., 2020). Neural Network is a data-driven algorithm, and the model quality is extremely dependent on the quantity and quality of training data. For a long time, the acquisition of distribution network operation data has been the constraint of NN algorithm in the development of DNR problem. At present, with the improvement of monitoring equipment in distribution network, the massive data generated by it has become the soil for the development of distribution network big data and data-driven technology. Therefore, it is necessary to study DNR based on data drive (Ozcanli et al., 2020).

In existing studies, literature (Kim et al., 1993) regarded DNR as a switching state prediction problem based on network parameter matrix, so as to be applicable to the solution mode of NN. Literature (Zheng et al., 2020) used lSTM-based probability distribution prediction network to extract the reference joint probability distribution of DG output and load from historical data, and then applied it to the robust optimization and reconfiguration of three-phase unbalanced distribution network. Literature (Oh et al., 2020) proposed an online reconfiguration method of distribution network based on reinforcement learning. The voltage and load state of distribution network buses were introduced into the reinforcement learning reward mechanism, and the optimal topology was obtained by Deep Q-learning DQL algorithm. Convolutional Neural Network (CNN) was used in literature (Yin et al., 2020) to solve DNR problems, and an overall model was constructed based on the idea of loop combination, which verifies the effectiveness of CNN. Ji et al. (2021) proposes a dynamic distribution network reconfiguration model based on LSTM, which takes the network loss and switching action cost of the distribution system as the optimization objectives, and realizes the real-time reconfiguration while reducing the system operating cost and switching loss. In Malekshah et al. (2022), a dynamic distribution network reconfiguration method based on deep Q learning algorithm is proposed under the condition of a large number of distributed generation equipment being connected to the grid. The optimization objective is to minimize the network loss and voltage deviation, so as to realize the economical and reliable operation of the distribution system. A three-stage distribution network reconfiguration method based on deep deterministic policy gradient (DDPG) is proposed in Bui and Su (2022). By changing the topology of the system and the access location of the distributed power supply, the operating cost and load loss of the system can be reduced. Wang et al. (2021) proposes a distribution network reconfiguration method based on NoisyNet deep Q-learning network (DQN), which gets rid of the constraint of network structure to a certain extent, and achieves the purpose of reducing power consumption and improving voltage level.

Therefore, based on previous work, this paper proposes a fast reconfiguration model of distribution network based on convolutional neural network:

(1) The CNN-based distribution network fast reconfiguration model proposed for the first time in this paper, which does not require the complex and time-consuming iterative calculation, thus significantly improving the solution efficiency.
(2) The Inception module (Szegedy et al., 2016) is introduced into the CNN model to extract multi-scale features of data through multiple convolution channels in parallel, which effectively improves the ability of the CNN model to extract data features.

The rest of this paper is structured as follows. Section 2 introduces the principle and model building of CNN. Section 3 describes the data set construction of CNN-DNR model. The case studies are conducted in Section 4 to verify the proposed model, and the conclusions are drawn in Section 5.

2 Principle and model building of CNN

The powerful feature extraction performance of CNN comes from its deep structure and convolution operation, which is further enhanced by the expansion of Inception module on the network scale. Therefore, in this paper, the Inception V3 module is used as the basic unit for feature extraction, combined with the loop structure of the distribution network to construct the DR-CNN

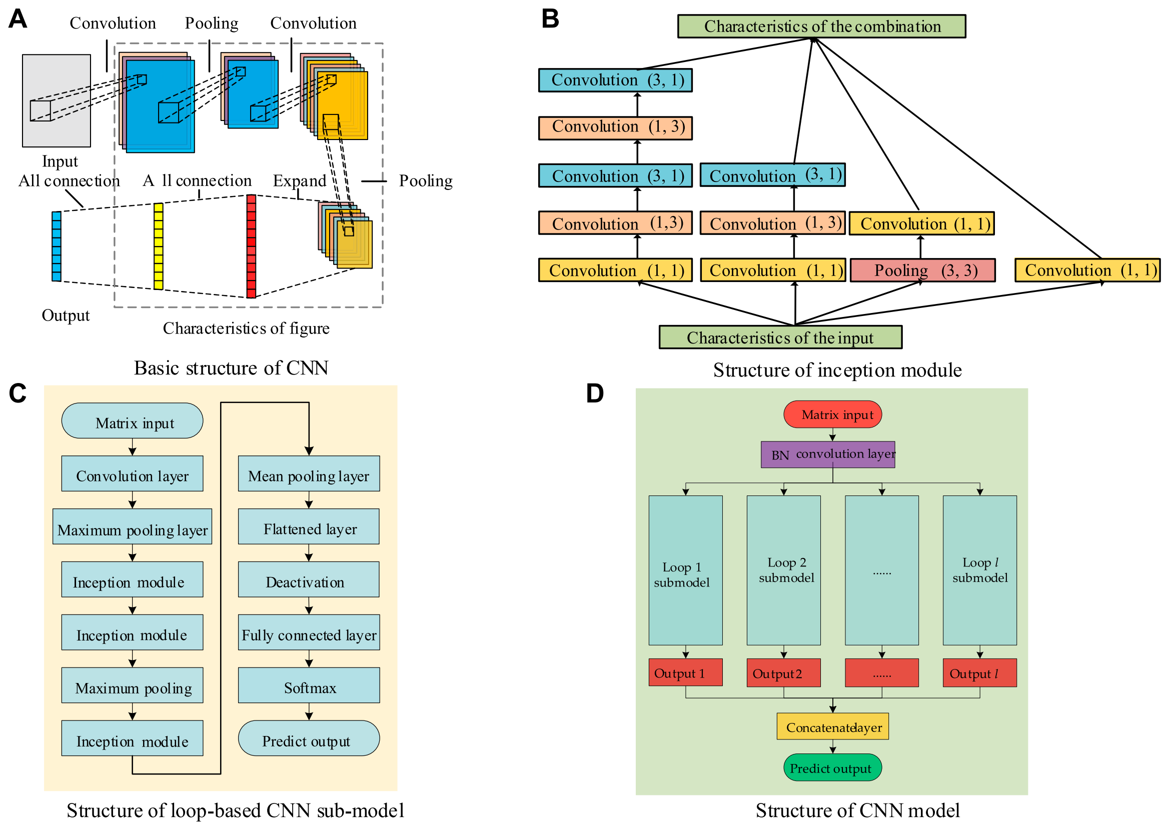

FIGURE 1
Basic structure of CNN model.

model, and the integrated network is trained based on the method of weight transfer.

2.1 CNN infrastructure

The infrastructure of CNN is shown in Figure 1A. The basic CNN structure consists of convolution, activation and pooling. In general, when processing classification tasks, the output of CNN is taken as the input of the full connection layer, and the mapping be-tween input matrix and label set is completed by the full connection layer.

2.2 CNN model building

DNR based on CNN can be regarded as a prediction problem of distribution net-work switch status. Data features of distribution network load patterns are extracted through Inception module to match the corresponding switch status. The calculation of switching state is based on Softmax classifier, and the number of classification is the same as the number of distribution network buses. In order to improve the performance of the classifier, the whole CNN model is constructed with the structure of split loop and multi-classifier.

2.2.1 Google inception module structure

In order to facilitate the extraction of high-dimensional features of data, the Inception module is used to build a CNN model.

Inception module is the basic component of GoogLeNet. This module decomposes the larger convolution layer in CNN into smaller convolution layer and pooling layer, which can improve the performance in the following two aspects: First, network learning parameters are reduced through the decomposition of the convolution layer, and the model learning efficiency is improved. Second, the decomposition of the convolution layer enables the model to have the ability of multi-channel information processing, which improves the breadth of the network. Since the model outputs the summary of multi-channel processing results, the information representation ability is enhanced. The specific operation of convolution kernel decomposition is as follows:

(1) Convolution kernel decomposition. 5×5 convolution is decomposed into two serial 3×3 convolution with the same receptive field, while the latter has more non-linear operations. The number ratio of the two parameters is $(3^2 + 3^2)/5^2 = 0.72$, that is, the number of parameters is reduced by 28%.
(2) Convolution decomposition of space. Decomposition of 3×3 convolution into the asymmetric convolution of 3×1 and 1×3 in series can also improve the representation ability. The ratio of the two parameters was $(3 \times 1 + 3 \times 1)/3^2 \approx 0.66$, and the number of parameters decreased by 33%.

The form of convolution decomposition can be seen in the Inception module structure shown in Figure 1B. As can be seen from Figure 1B, Inception module expands both the breadth and depth of the network. That is, pooling operation and multi-scale convolution processing are adopted to process data. Such multi-channel data processing method increases the network breadth. The decomposition of convolution kernel increases the depth of the network, and also increases the number of activation functions, thus strengthening the characterization ability of the network.

2.2.2 Single loop CNN model

Power switch classification problem requirement model to predict all the state of the switch in the power distribution network, to reduce the pressure of the classification of the Softmax classifier, this paper takes the IEEE 33 buses system as an example, based on the loop structure of distribution network structures, multiple single loop CNN submodel, combining classifier output of each sub models predict each state of the switch. The structure of CNN sub-model is shown in Figure 1C.

As can be seen from Figure 1C, the model first preprocesses the input distribution network parameter matrix rows with convolution and pooling, and then imports the feature extraction part composed of Inception module. The data features are transmitted to the full connection layer after secondary sampling and flattening for mapping be-tween data and labels. Finally, Softmax function is used to calculate and output the state prediction results of each switch in the corresponding loop.

Loop switch state prediction is a multi-classification problem, and its loss optimization process is as follows:

(1) The Softmax classifier outputs the classification index. The Softmax function is defined as the ratio of the exponent of a single element to the exponent sum of all elements:

$$Z_\eta = \frac{e^{z_\eta}}{\sum_c e^{z_c}} \tag{1}$$

(2) The loss of the model is calculated based on classification cross entropy:

$$L(\theta) = -\frac{1}{k}\sum_{m=1}^{k}\sum_{n=1}^{c} y_{m,n}\cdot\log\left(Z_{\theta,(m,n)}(z)\right) \tag{2}$$

(3) The model loss value is imported into the optimizer to update the network weights, and the Adaptive Moment Estimation (Adam) method is used to accelerate the gradient descent.

Adam algorithm is the combination of momentum acceleration decline and root mean square back propagation. By calculating the first and second moment estimates of the gradient, independent adaptive learning rates are designed for different parameters. The algorithm can correct the problems of learning rate disappearance and slow con-vergence in the optimization process, and its iterations are as follows:

$$\begin{cases} g_t = \nabla_\theta L_t(\theta_{t-1}) \\ \hat{m}_t = \dfrac{\beta_1 m_{t-1} + (1-\beta_1) g_t}{1-\beta_1^t} \\ \hat{v}_t = \dfrac{\beta_2 v_{t-1} + (1-\beta_1) g_t^2}{1-\beta_2^t} \\ \theta_t = \theta_{t-1} - \dfrac{\alpha \hat{m}_t}{\sqrt{\hat{v}_t}+\varepsilon} \\ t = t+1 \end{cases} \tag{3}$$

2.2.3 Loop model integration

In the reconfiguration calculation of distribution network, the actions of switches between loops will affect each other. Therefore, the interaction between sub-models should also be considered in the training of CNN model for unified optimization. The structure of the integration model is shown in Figure 1D.

It can be seen from Figure 1D that the number of branches of the integration model is the same as the loop number L of the distribution network. During the data input phase of the model, a convolutional layer with Batch Normalization BN is set up for initial data processing, which is then fed into each circuit sub-model.

The Concatenate layer is used to combine the output of the model, and the final output of the model is the splicing of the output of sub-model classifiers in the first dimension:

$$\mathbf{O}^{\mathrm{con}} = [\mathbf{O}_1, \mathbf{O}_2, ..., \mathbf{O}_l] \tag{4}$$

The model output needs to be combined with the corresponding data labels to calculate the loss of each branch, and take the maximum value to participate in the weight iteration of the model. The loss is calculated as:

$$L^{\mathrm{c}} = \max[L_1, L_2, ..., L_l] \tag{5}$$

For the loss L^{c} calculated by the model, the iteration format of Adam algorithm's parameters is:

$$\begin{cases} g_t^{\mathrm{c}} = \nabla_\theta L_t^{\mathrm{c}}(\theta_{t-1}^{\mathrm{c}}) \\ \hat{m}_{l,t} = \dfrac{\beta_1 m_{l,(t-1)} + (1-\beta_1) g_t^{\mathrm{c}}}{1-\beta_1^t} \\ \hat{v}_{l,t} = \dfrac{\beta_2 v_{l,(t-1)} + (1-\beta_1)(g_t^{\mathrm{c}})^2}{1-\beta_2^t} \\ \theta_{l,t} = \theta_{l,t} - \dfrac{\alpha \hat{m}_{l,t}}{\sqrt{\hat{v}_{l,t}}+\varepsilon} \\ t = t+1 \end{cases} \tag{6}$$

2.3 Model mixed training based on weight transfer

Aiming at the difficulty of training due to the complex structure of the integrated model, this paper adopts the hybrid training method based on weight transfer to train the model. The process can be divided into three stages: single loop model training—weight transfer—integrated model mixed training.

The following constraints are set as the criteria for process termination in the training:

(1) Training cycle constraints

$$E > E^{\mathrm{set}} \tag{7}$$

(2) Optimization effect constraint

$$L_E < L_{E+e}, e = 1, 2, ..., E^{\mathrm{lim}} \tag{8}$$

The training process is shown in Figure 2. The specific steps are as follows:

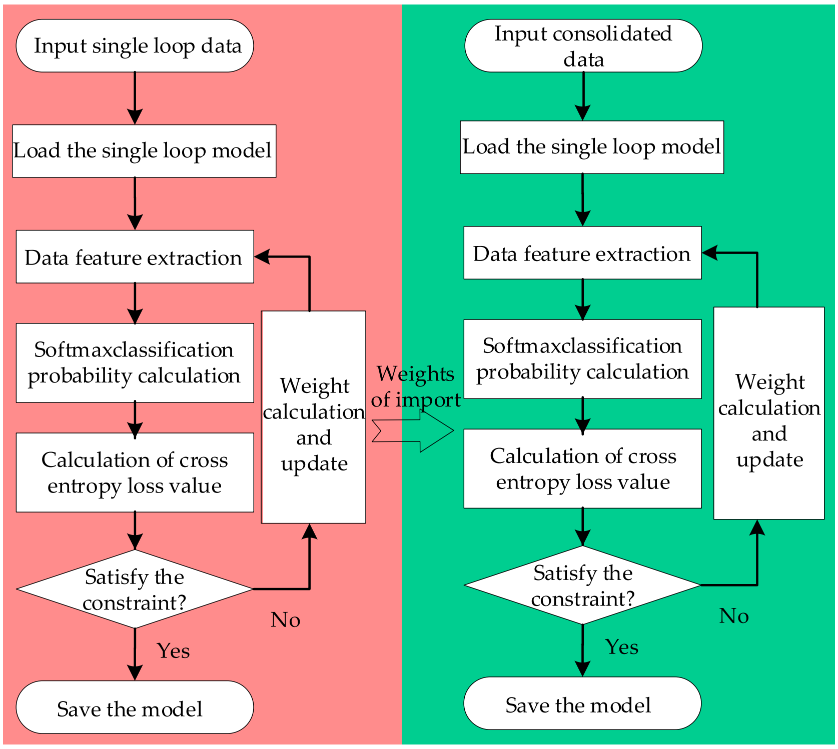

FIGURE 2
Training process.

Step 1: Load single-loop data of distribution network for data processing. The distribution network parameter matrix is normalized, the loop switch state is converted into one HOT code, and the training set and test set are divided.

Step 2: Load the single-loop model and initialize the model parameters.

Step 3: Import data, extract data features, and complete the forward propagation process.

Step 4: Calculate the state probability matrix of loop switch according to Formula 1. The larger the matrix element value is, the greater the probability of switch dis-connection is.

Step 5: According to the output matrix obtained in Step 4 and the switch coding in Step 1, calculate the model loss according to Formula 2.

Step 6: Constraint judgment. According to the initial training rounds and callback round limits of the model, determine whether the training process is terminated in combination with Eqs 7, 8. If yes, end the training and save the model, and go to Step 8. Otherwise, go to Step 7.

Step 7: Calculate the new weights according to Formula 3, update the network weights, complete the back propagation process, and go to Step 3.

Step 8: Extract weight of single loop model.

Step 9: Load the overall data of distribution network, normalize the data, and convert labels into one-HOT coding.

Step 10: Load the integration model, and import the corresponding sub-model weights for each branch.

Step 11: Import data, extract data features, and complete the forward propagation process.

Step 12: Calculate the output of Softmax classifier according to Eq. 1, and the output is the state probability matrix of each loop switch, as shown in Eq. 4.

Step 13: According to the output matrix obtained in Step 12, combined with the switch coding in Step 8, the model is calculated according to Eqs 2, 5.

Step 14: Constraint judgment: Determine whether the training process is terminated according to Eqs 7, 8. If so, end the training and save the model. Otherwise, go to Step 15.

Step 15: Calculate the new weights according to Formula 6, update the network weights, complete the back propagation process, and go to Step 11.

3 Data set construction of CNN-DNR model

The data set of convolutional neural network in this paper is a data set composed of system load mode and corresponding optimization switch combination, and its construction process is shown in Figure 3.

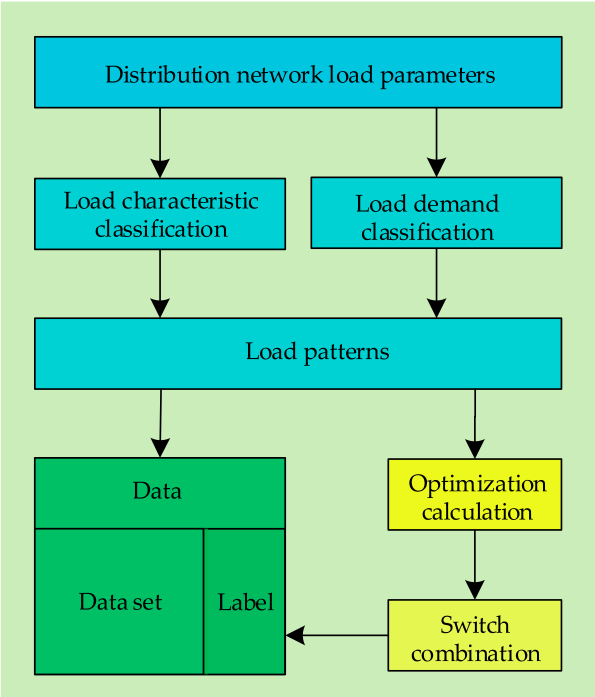

FIGURE 3
Data set construction process.

TABLE 1 Comparison of actual load and calculated load.

Actual load	Computational load (%)
≤35%	30
36%–45%	40
46%–55%	50
56%–65%	60
66%–75%	70
76%–85%	80
86%–95%	90
≥96%	100

3.1 Load patterns of CNN datasets

Load mode is the set of all load demand states of the system. For a given system, the demand-based load mode is divided into: load mode is the set of all load demand states of the system. For a given system, the demand-based load mode is divided into:

$$\mathbf{LM} = M^N \tag{9}$$

According to the percentage of peak load demand, demand level is classified into eight levels. The corresponding relationship between actual load and calculated load is shown in Table 1.

In order to further reduce the data complexity, the load bus is divided into three groups: industrial, commercial and residential. The loads in the same group have similar load characteristics and change curves. By considering the total requirements of the load group, the number of load modes can be reduced to $8^3 = 256$.

3.2 Distributional robust optimization model based on box decomposition algorithm

In this paper, Quantum Particle Swarm Optimization (QPSO) algorithm is adopted to solve the corresponding switch combination for the load mode divided. The algorithm takes the minimum active power loss of the system as the objective function.

$$\min f = \sum_{b=1}^{N_b} K_b \cdot R_b \cdot I_b^2 \tag{10}$$

The optimized switch combination should meet the following constraints:

(1) Topology constraints

$$\varphi \in \mathbf{\Gamma} \tag{11}$$

(2) Flow constraint

$$\begin{cases} P_i + P_i^{DG} = P_i^L + V_i \sum_{j=1}^{N} V_j \left(G_{ij} \cos \delta_{ij} + B_{ij} \sin \delta_{ij} \right) \\ Q_i + Q_i^{DG} = Q_i^L + V_i \sum_{j=1}^{N} V_j \left(G_{ij} \cos \delta_{ij} + B_{ij} \sin \delta_{ij} \right) \end{cases} \tag{12}$$

(3) Bus voltage constraint

$$V_i \in \left[V_i^{\ min}, V_i^{\ max} \right] \tag{13}$$

(4) Branch capacity constraints

$$S_b \leq S_b^{\ max} \tag{14}$$

3.3 Data and label processing

The training of CNN model requires training data and corresponding labels, which are transformed from load mode and optimization topology respectively. According to the training requirements of CNN, the input data is the load mode after matrix and normalization. Label distribution loop topology in one-HOT encoding format.

The matrix Λ describing the parameters of the distribution network is taken as the input matrix of the CNN model, where the number of columns of the matrix is consistent with the number of nodes of the training system, and each row of the matrix represents different data types of the distribution network such as the access state of DG, the active power of DG, the reactive power of PQ-type DG, the voltage of PV type DG, the current of PI type DG, the type of DG, the active power of load and the reactive power of each node.

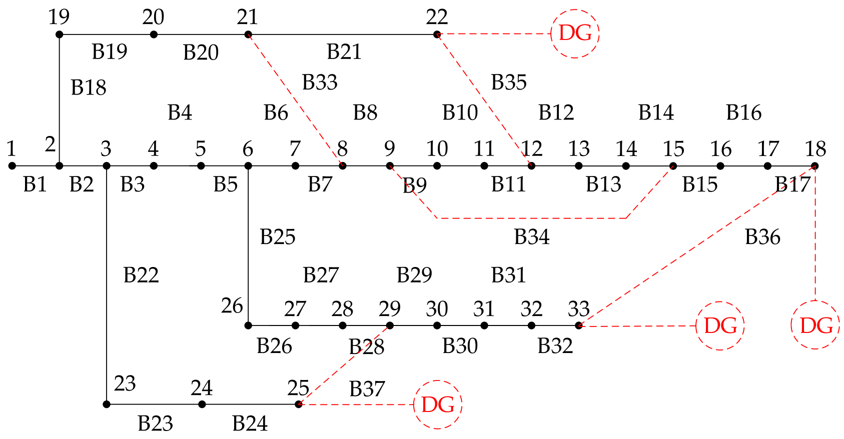

FIGURE 4
IEEE 33-bus system.

TABLE 2 **IEEE 33-bus system loop-branch division.**

Loop	Contains branch
1	{B2,B3,B5,B6,B7,B18,B19,B20,B33}
2	{B9,B10,B11,B12,B13,B14,B34}
3	{B2,B3,B4,B5,B6,B7,B8,B9,B10,B18,B19,B20,B21,B35}
4	{B6,B7,B8,B9,B10,B11,B12,B13,B14,B15,B16,B17,B25,B26,B27,B28,B29,B30,B31,B32,B36}
5	{B3,B4,B5,B23,B24,B25,B26,B27,B28,B37}

$$\mathbf{\Lambda} = \begin{bmatrix} F_1^{DG} & F_2^{DG} & \dots & F_i^{DG} & \dots & F_N^{DG} \\ P_1^{DG} & P_2^{DG} & \dots & P_i^{DG} & \dots & P_N^{DG} \\ Q_1^{PQ} & 0 & \dots & 0 & \dots & Q_N^{PQ} \\ 0 & V_2^{PV} & \dots & 0 & \dots & V_N^{PV} \\ 0 & 0 & \dots & I_3^{PI} & \dots & I_N^{PI} \\ T_1^{DG} & T_2^{DG} & \dots & T_i^{DG} & \dots & T_N^{DG} \\ P_1^{L} & P_2^{L} & \dots & P_i^{L} & \dots & P_N^{L} \\ Q_1^{L} & Q_2^{L} & \dots & Q_i^{L} & \dots & Q_N^{L} \end{bmatrix}_{8\times N} \tag{15}$$

Before model training, data shown in Eq. 15 should be normalized and ex-pressed as:

$$\mathbf{A}_a = \frac{\mathbf{\Lambda}_a}{\lambda_a} \tag{16}$$

The network topology needs to be transformed into the one-hot coding combination of each loop topology, and then regarded as the label of multi-classification problem. The radial topology constraint of distribution network requires that only one switch in each loop is off at the same time. According to this characteristic, the status of the loop interrupt on switch is set to 1, and the status of the closed switch is set to 0. In this way, the loop l_b with branches has l_b states, and only one switch in each state is in the open position, which conforms to the one-HOT encoding format. Taking the IEEE16 bus system as an example, the one-Hot coding principle of loop topology is explained in detail.

For switch configuration C_h , its corresponding one-HOT code is as follows:

$$\begin{cases} C_h = [B14, B15, B16] \\ H_{1,h} = [0, 0, 0, 0, 0, 1] \\ H_{2,h} = [0, 0, 0, 0, 1] \\ H_{3,h} = [0, 0, 0, 0, 0, 0, 1] \end{cases} \tag{17}$$

For a power distribution system with l loops, each input matrix A has a corresponding one-hot code H of l loops. Based on this principle, the DNR problem can be transformed into the switch multi-classification problem of each loop.

4 Case studies

In order to verify the effectiveness of the distribution network rapid reconfiguration model based on deep learning proposed in this paper, IEEE 33-bus power distribution system is taken as a test example, and the system structure is shown in Figure 4 and the system loop-branch division is shown in Table 2. Based on Python environment programming, call Keras deep learning library to build the model. The computer operating system is Windows10. The CPU is Intel I7 1165G7, the main frequency is 2.8 GHz, and the memory is 8 GB. Table 2.

The specific parameters of the DG are shown in Table 3. The connected DG is di-vided into 30%, 60%, and 100% according to the output level. Therefore, there are 19 scenarios of distributed power supply.

TABLE 3 DG configuration.

Serial number	DG number	Type	Parameter (p.u.)	Added bus
1	0	-	-	-
2	1	PQ	P = 0.03, Q = 0.01	18
3	4	PQ	P = 0.03, Q = 0.01	18, 22, 25, 33
4	1	PV	P = 0.03, |U| = 0.95	18
5	4	PV	P = 0.03, |U| = 0.95	18, 22, 25, 33
6	1	PI	P = 0.03, |I| = 0.0325	18
7	4	PI	P = 0.03, |I| = 0.0325	18,22,25,33

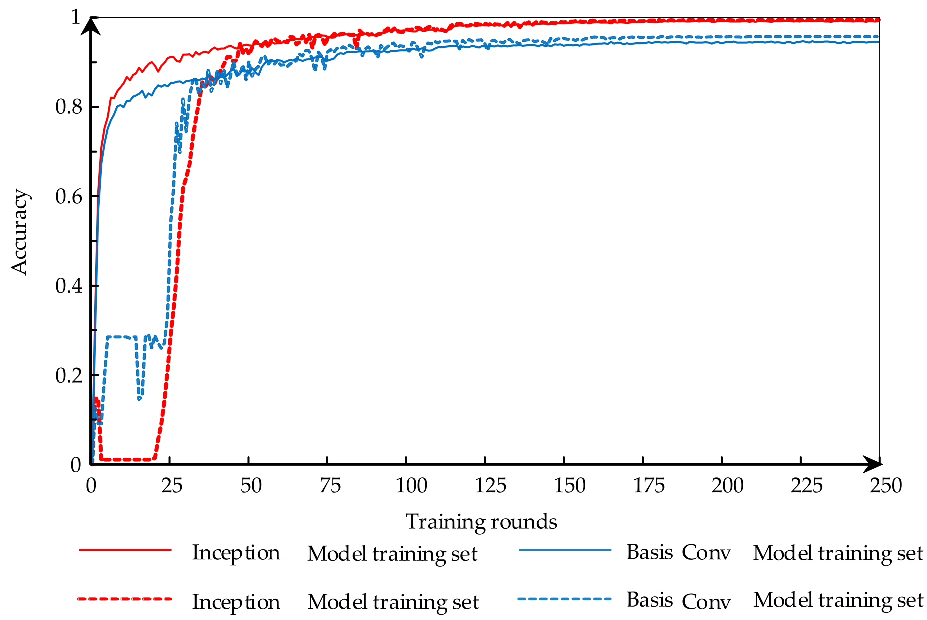

FIGURE 5
Accuracy of model classification.

4.1 Model training effect

4.1.1 Training

The dataset contains 19 DG scenarios as shown in Table 3, each of which is divided into 512 load modes and a total of 9,728 system state sections. In the training process, the proportion of training set to total data is 80%, test set to 15%, verification set to 5%. To verify the performance advantage of the CNN model based on Inception module in this paper, the training effect is compared with that of the basic CNN model. The accuracy of the two models on the training set and test set is shown in Figure 5.

It can be seen from Figure 5 that the Inception CNN model has better feature extraction capability than the basic convolutional model. The training effect of the former on the training set and test set and the overall accuracy increase rate of the former are better than the latter. After 250 rounds of training, the accuracy of the model in the training set and test set is 99.31% and 99.28% respectively. From the curve trend, in the first 50 rounds, the precision of the model training set is ahead of that of the test set. During the 50 to 250 rounds of training, the accuracy curves of the two are similar, and tend to be stable after 200 rounds. In the early stages of training (rounds 0–25), due to the differences among the branches of the Inception model, which enable the optimization of the test accuracy is delayed. Thereby, resulting in the classification accuracy of the Inception test set being lower than that of the basic Conv model test set. Due to the focus of the hybrid training method on the high-loss branch of the model, the final training effect of the model is guaranteed.

4.1.2 Generalization ability

According to Figure 6, the main operation switches of system optimization recon-figuration are located at B7, B9, B14, B23, and B26. In addition, the classification error of the model is mainly reflected in the confusion of switch states between branches B9-B10-B14 and B22-B23.

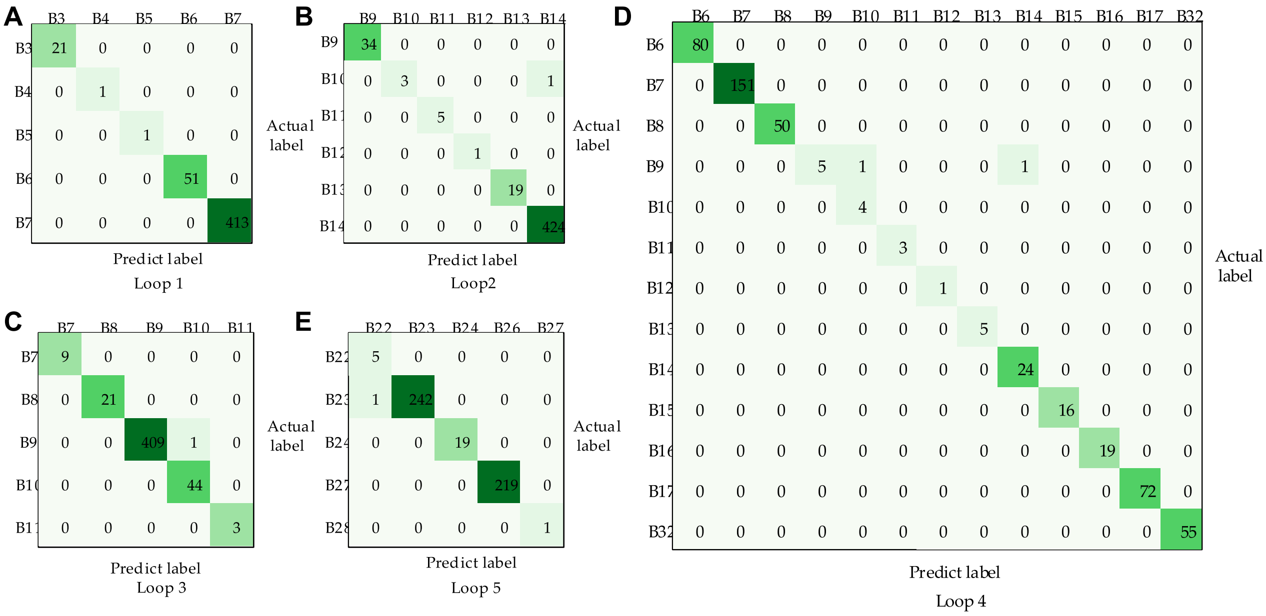

FIGURE 6
Loop confusion matrix.

TABLE 4 Classification index of main action branch.

Branch	Number of samples	Accuracy (%)	Recall rate (%)	F1 score
B6	131	100	100	100
B7	573	100	100	100
B8	71	100	100	100
B9	451	100	99.33	99.66
B10	52	96.23	98.08	97.15
B13	24	100	100	100
B14	448	99.56	100	99.78
B17	72	100	100	100
B22	5	83.33	100	90.91
B23	243	100	99.59	99.79
B26	219	100	100	100
B32	55	100	100	100

In addition, the performance index of DL model is the quantification of model performance based on confusion matrix, including accuracy, recall rate and F1 score. Accuracy rate represents the real tag proportion of model prediction results. Recall rate represents the prediction accuracy of the model for real labels. The F1 score is the harmonic average of the first two, and the F1 score of the model is positively correlated with its performance. The classification index of action branch based on confusion matrix is shown in Table 4.

It can be seen from Table 4 that the switch classification prediction effect of the model for large sample branches and most small sample branches is good, and each performance score is above 99. However, the classification performance of small sample branches B10 and B22 decreased, and their F1 scores were 97.15 and 90.91, respectively. It shows that insufficient label sample size will lead to the decrease of model generalization ability. The reason is on the one hand, the data feature coverage of small sample is not complete. On the other hand, it is the coverage of large sample features to small sample features in data sets.

 10.3389/fenrg.2023.1102949

TABLE 5 Program running time statistics.

Configuration	Amount of data	Running time (s)	All time calculation (ms)	Model and data loading	
				Time (s)	Ratio (%)
a	2000	60.77	1.99	58.78	93.43
b	20,000	86.89	1.51	58.78	65.35
C	2000	3.99	1.99	–	–

TABLE 6 Comparison of algorithm solution time.

Algorithm	QPSO	ICSA	RF	LSTM	The algorithm in this paper	
					Online	Offline
Time/s	7.315	4.509	4.767	5.137	0.002	0.641
Loss/kW	139.584	139.554	139.569	139.588	139.554	139.554

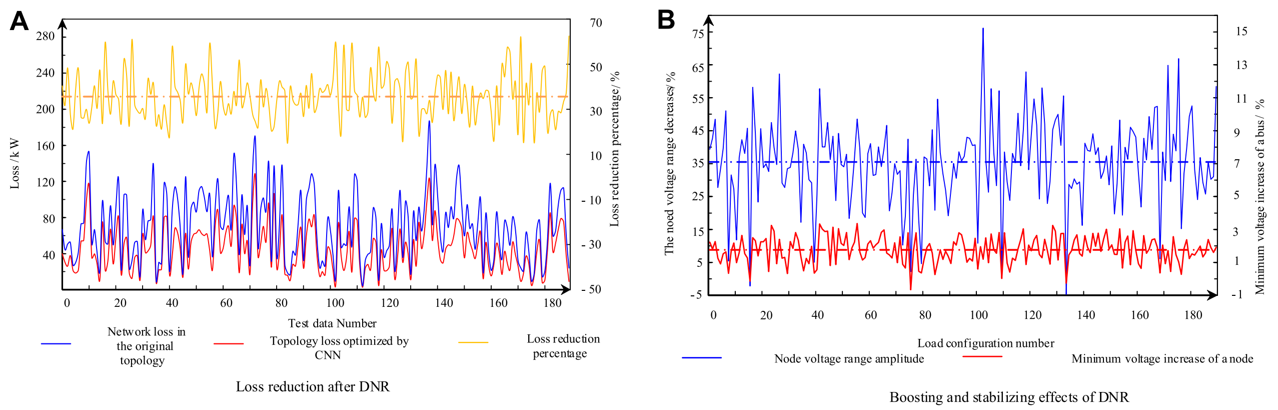

FIGURE 7
Results and effects after DNR.

4.2 Model performance analysis

4.2.1 Solution efficiency analysis

In order to verify the timeliness of THE CNN model, 2,000 groups of data were randomly selected from the data set as the prediction target, and imported into the CNN model completed by training for prediction and timing. The results of program running time are shown in Table 5. In Table 5, a and b are configured to simulate offline operation, and the model loading time is included in the total time consumed, which includes the model loading time and the program operation time. Configuration c is set to simulate online operation, and the total time consumed is only the operation time of the program.

By comparing the operation of each configuration in Table 5, it can be seen that the model loading time accounts for more than 65% of the total offline operation. Excluding model loading, the calculation time of all three is less than 2 ms, showing no significant difference. By comparing configuration A and B, it can be seen that when the amount of calculation data is increased, the proportion of model loading time decreases from 93.43% to 65.35%, and the influence on the total time is reduced.

The standard IEEE33-bus distribution network is taken as the test system. The proposed method is compared with QPSO, improved cuckoo algorithm improved cuckoo search algorithm (ICSA), random forest (RF) and long short-term memory (LSTM) to calculate the average of 50 metaheuristic algorithms. The solution results are shown in Table 6.

As shown in Table 6, compared with QPSO, ICSA, RF, and LSTM methods, the online and offline operation time of the proposed method is significantly reduced and controlled to within 1s in terms of solution efficiency. Therefore, the proposed method can realize the rapid reconfiguration of distribution network, because the CNN model does not need to perform iterative optimization in the solution space. The reconfiguration strategy can be directly obtained according to the mapping relationship between the load pattern established in the training process and the reconfiguration strategy. In terms of network loss, the loss reduction ability of all the five

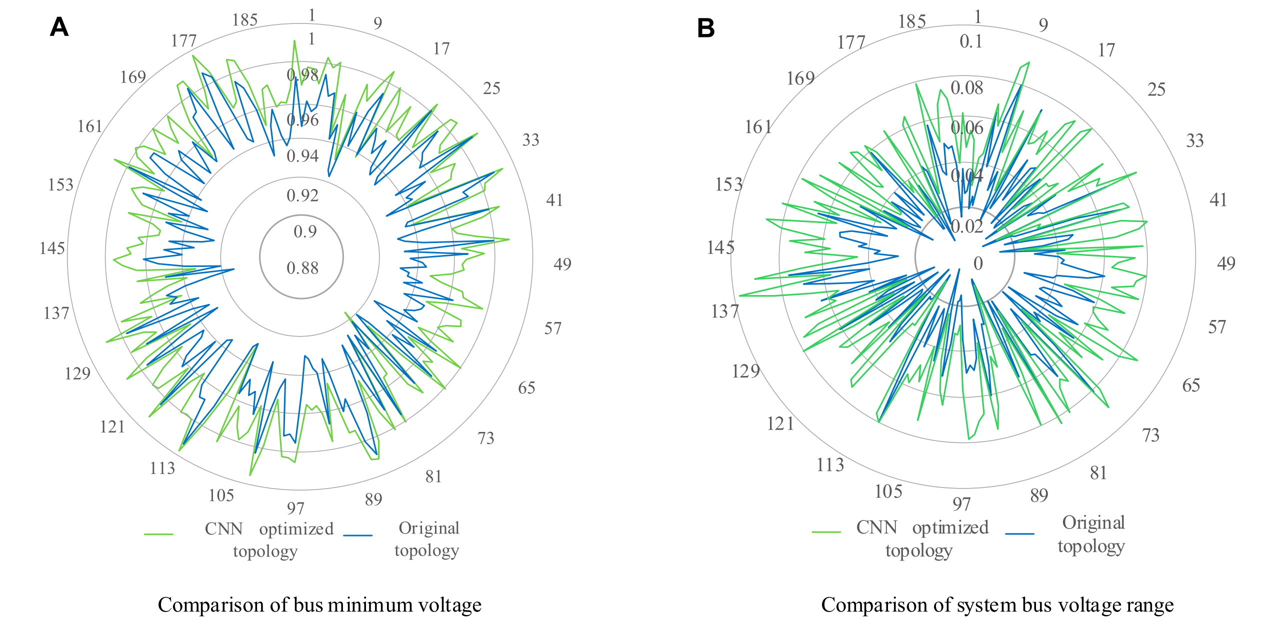

FIGURE 8
Comparison of bus voltage.

algorithms is similar. Among them, the network loss of the ICSA algorithm is the same as that of the proposed method, which is 139.554 KW, while the network loss of the other three methods is slightly higher but it can be maintained below 139.590 KW. Therefore, compared with other algorithms, the fast reconfiguration method of distribution network based on CNN proposed in this paper improves the solution efficiency and reduces the system network loss at the same time.

In conclusion, there is no significant difference between online and offline data calculation speed of CNN model, while the influence of model loading time on the overall calculation speed can be diluted by increasing single input data amount in offline calculation. In addition, the precision of data-driven model needs a large amount of data support, while the meta-heuristic algorithm has little data requirements, which is more advantageous in the optimization and reconfiguration of new systems. Therefore, CNN model is more suitable for the scenarios of online real-time reconfiguration and mass reconfiguration of data operations.

4.2.2 Loss reduction effect analysis

In order to verify the effect of CNN model in loss reduction proposed in this paper, 190 groups of data in the validation set were extracted for prediction, and network loss and voltage level before and after reconfiguration were compared and predicted. Figure 7A shows the comparison of network loss before and after reconfiguration.

According to Figure 7A, compared with the topology before system reconfiguration, the topology optimized by CNN has lower network active power loss. According to the loss reduction percentage curve, in the test data, the average network loss of the system is reduced by 35.63% after network reconfiguration. The reason is that CNN's optimized re-selection of network open-loop points balances the power flow of each branch and reduces the active network loss of the system.

TABLE 7 DG Configuration.

Bus	Type	Active power (MW)	Power factor
140	photovoltaic	10	0.8
45	photovoltaic	1	0.9
7	photovoltaic	1	0.8
114	photovoltaic	1	0.9
18	wind power	0.5	0.85
52	wind power	0.5	0.9
181	wind power	0.4	0.85
145	wind power	0.4	0.9

4.2.3 Analysis of improvement effect of voltage level

Further, in order to analyze the influence of CNN model proposed in this paper on voltage before and after reconfiguration. Figures 8A,B respectively show the comparison diagram of bus minimum voltage and voltage range before and after system re-configuration. Figure 7B shows the effect of the reconfiguration on the system voltage level.

It can be seen from Figures 8A,B that the optimization of network bus voltage by CNN prediction topology is reflected in the increase of bus minimum voltage and the decrease of bus voltage range. According to Figure 7B, in the test data, the voltage range decreased by 35.47% on average, and the minimum voltage of system buses increased by 1.76% on average. The reason is that CNN prediction topology optimizes the power flow of the system, reduces the voltage loss of each branch, reduces the voltage difference between buses, improves the power supply quality of the system, and contributes to the low-carbon economic operation of the distribution network.

TABLE 8 185 bus system program running result.

Configuration	Amount of data	Running time (s)	Average time (ms)	Model data loading	
				Time(s)	Account (%)
a	2,000	81.03	2.30	76.41	94.29
b	20,000	121.81	2.13	76.41	64.20
c	2,000	4.62	2.31	–	–

TABLE 9 185 bus system program running time statistics.

	Loss (kW)	Lowest bus voltage (p.u.)	Loss reduction rate (%)
Original network	2843.20	0.9253	–
CNN model	2193.13	0.9508	22.86

4.2.4 185 bus system

In order to further verify the effectiveness of the fast reconfiguration method of low-carbon distribution network based on CNN in large-scale distribution systems, a 185-bus distribution system is taken as a test system. The system has 185 buses, 184 branches and 75 switches, including 20 tie switches and 55 segment switches. The parameters of DGs are shown in Table 7.

The results of program running time are shown in Table 8. As can be seen, due to the complexity of the 185-bus test system, the CNN model requires additional parameters to describe the relationship between load patterns and reconfiguration strategies, resulting in model data loading times and mean times longer than the IEEE33 node system, ultimately increasing the solution time. However, even though the test system is more complex, the mean solution time of the proposed CNN model remains below 2.5 ms and rapid reconfiguration is achieved.

Running results of 185-bus system are shown in Table 9. It can been seen in Table 9, compared with the original network, CNN reduces the power loss by 22.86% while improving the lowest node voltage from 0.9253p.u. to 0.9508 p.u. As a result, the CNN method proposed in this paper also has good performance advantages in the 185-node system.

5 Conclusion

This paper proposes a power distribution network based on convolution neural network fast reconfiguration model, based on system load model and the corresponding optimization combination switch building data sets, using CNN to extract data collection of information, and to all the status of the network switch decisions, the last on IEEE33 bus system validation DNR performance of the proposed CNN model in this paper. Through the analysis of examples, the following conclusions are obtained:

(1) The CNN model based on Inception module has stronger feature extraction capability and better final training effect, and the resulting model has good generalization capability for validation sets.
(2) The CNN model constructed in this paper has an average decision speed of milli-second level, which is suitable for online fast reconfiguration and offline mass computation.
(3) The network topology predicted by the model can effectively improve the power supply quality and low-carbon operation economy of the distribution system.

Although the distribution network fast reconfiguration algorithm based on CNN proposed in this paper has excellent performance in static reconfiguration for a single period, the performance of this method in dynamic reconfiguration has not been verified. In the future research, the data-driven distribution network dynamic reconfiguration combined with the reconfiguration cycle division can be further studied, so as to realize the formulation of multi-period reconfiguration strategy.

Data availability statement

The original contributions presented in the study are included in the article/Supplementary Material, further inquiries can be directed to the corresponding author.

Author contributions

All authors listed have made a substantial, direct, and intellectual contribution to the work and approved it for publication.

Funding

This work is supported by National Natural Science Foundation of China, 52107111 and Shandong Provincial Natural Science Foundation, ZR2022ME219, ZR2021QE117.

Conflict of interest

The authors declare that the research was conducted in the absence of any commercial or financial relationships that could be construed as a potential conflict of interest.

Publisher's note

References

Abrahamyan, L., Chen, Y., Bekoulis, G., and Deligiannis, N. (2021). Learned gradient compression for distributed deep learning. *IEEE Trans. neural Netw. Learn. Syst.* 33, 7330–7344. doi:10.1109/tnnls.2021.3084806

Alavi, A. S., Mehran, K., Vahidinasab, V., and Catalao, J. P. S. (2021). Forecast-based consensus control for dc microgrids using distributed long short-term memory deep learning models. *IEEE Trans. Smart Grid* 12 (5), 3718–3730. doi:10.1109/tsg.2021.3070959

Bui, V., and Su, W. (2022). Real-time operation of distribution network: A deep reinforcement learning-based reconfiguration approach. *Sustain. Energy Technol. Assessments* 2022, 101841. doi:10.1016/j.seta.2021.101841

Chen, X., and Dai, Y. (2020). "Research on an improved ant colony algorithm fusion with genetic algorithm for route planning," in IEEE 4th information technology, networking, Electronic and Automation Control Conference (ITNEC), Chongqing, China, 12-14 June 2020, 1273–1278.

DE Oliveira, J. P., Costa, M. G. F., and Filho, C. F. F. C. (2020). Methodology of data fusion using deep learning for semantic segmentation of land types in the amazon. *IEEE Access* 8, 187864–187875. doi:10.1109/access.2020.3031533

Huang, W., Zheng, W., and Hill, D. (2021). Distribution network reconfiguration for short-term voltage stability enhancement: An efficient deep learning approach. *IEEE Trans. Smart Grid* 12 (6), 5385–5395. doi:10.1109/tsg.2021.3097330

Hussain, M., Shakir, H., and Rasheed, H. (2021). Deep learning approaches for impulse noise mitigation and classification in noma-based systems. *IEEE Access* 9, 143836–143846. doi:10.1109/access.2021.3121533

Jin, Y., Zhang, L., and Niu, Q. (2020). Reconfiguration of distribution network with wind power generators considering ran-domness and fuzziness. *Proc. CSU-EPSA 2020* 32 (07), 67–72.

Ji, X., Yin, Z., Zhang, Y., Xu, B., and liu, Q. (2021). Real-time autonomous dynamic reconfiguration based on deep learning algorithm for distribution network. *Electr. Power Syst. Res.* 195 (3), 107132. doi:10.1016/j.epsr.2021.107132

Kim, H., Ko, Y., and Jung, K. (1993). Artificial neural-network based feeder reconfiguration for loss reduction in distribution systems. *IEEE Trans. Power Deliv.* 8 (3), 1356–1366. doi:10.1109/61.252662

Li, J., Wei, S., and Dai, W. (2021). Combination of manifold learning and deep learning algorithms for mid-term electrical load forecasting. *IEEE Trans. Neural Netw. Learn. Syst.* 2021, 1–10. doi:10.1109/TNNLS.2021.3106968

Li, Y., Gao, D., Gao, W., Zhang, H., and Zhou, J. (2021). A distributed double-Newton descent algorithm for cooperative energy management of multiple energy bodies in energy internet. *IEEE Trans. Industrial Inf.* 17 (9), 5993–6003. doi:10.1109/tii.2020.3029974

Li, Z., Li, Y., Liu, Y., Wang, P., Lu, R., and Gooi, H. B. (2021a). Deep learning based densely connected network for load forecasting. *IEEE Trans. Power Syst.* 36 (4), 2829–2840. doi:10.1109/tpwrs.2020.3048359

Li, Z., Wu, L., and Xu, Y. (2021b). Risk-averse coordinated operation of a multi-energy microgrid considering voltage/var control and thermal flow: An adaptive stochastic approach. *IEEE Trans. Smart Grid* 12, 3914–3927. doi:10.1109/tsg.2021.3080312

Li, Z., Xu, Y., Feng, X., and Wu, Q. (2020). Optimal stochastic deployment of heterogeneous energy storage in a residential multienergy microgrid with demand-side management. *IEEE Trans. Ind. Inf.* 17, 991–1004. doi:10.1109/tii.2020.2971227

Malekshah, S., Rasouli, A., Malekshah, Y., Ramezani, A., and Malekshah, A. (2022). Reliability-driven distribution power network dynamic reconfiguration in presence of distributed generation by the deep reinforcement learning method. *Alexandria Eng. J.* 61 (8), 6541–6556. doi:10.1016/j.aej.2021.12.012

Oh, S., Yoon, Y. T., and Kim, S. W. (2020). Online reconfiguration scheme of self-sufficient distribution network based on a reinforcement learning approach. *Appl. Energy* 280, 115900. doi:10.1016/j.apenergy.2020.115900

Ozcanli, A. K., Yaprakdal, F., and Baysal, M. (2020). Deep learning meth-ods and applications for electrical power systems: A com-prehensive review. *Int. J. Energy Re-search* 44 (9), 7136–7157. doi:10.1002/er.5331

Pegado, R., Ñaupari, Z., Molina, Y., and Castillo, C. (2019). Radial distribution network reconfiguration for power losses reduction based on improved selective BPSO. *Electr. Power Syst. Res.* 169, 206–213. doi:10.1016/j.epsr.2018.12.030

Prasad, P. S., and Sushama, M. (2022). "Distribution network reconfiguration and capacitor allocation in distribution system using discrete improved grey wolf optimization," in *Innovations in electrical and electronic engineering. ICEEE 2022.* Editors S. Mekhilef, R. N. Shaw, and P. Siano (Singapore: Springer). Lecture Notes in Electrical Engineering.

Qing, Y., Liu, T., He, C., Nan, L., Dong, G., Gao, W., et al. (2021). Low-carbon coordinated scheduling of integrated electricity-gas distribution system with hybrid AC/DC network. *IET Renew. Power Gener.* 16, 2566–2578. doi:10.1049/rpg2.12439

Szegedy, C., Vanhoucke, V., and Ioffe, S. (2016). "Rethinking the inception architecture for computer vision," in Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 27-30 June 2016, 2818–2826.

Talukdar, B., Deka, B., Goswami, A., and Saha, D. (2019). "Reconfiguration of radial distribution network implementing tlbo algorithm for loss minimization and reliability improvement," in Intelligent Techniques and Applications in Science and technology. *ICIMSAT 2019*. Editors S. Dawn, V. Balas, A. Esposito, and S. Gope (Cham: Springer). Learning and Analytics in Intelligent Systems.

Wang, B., Zhu, H., Xu, H., Bao, Y., and Di, H. (2021). Distribution network reconfiguration based on noisynet deep Q-learning network. *IEEE Access* 9, 90358–90365. doi:10.1109/access.2021.3089625

Wang, C., Lei, S., Ju, P., Chen, Ch., Peng, C. Y., and Hou, Y. (2020). MDP-based distribution network reconfiguration with renewable distributed generation: Approximate dynamic programming approach. *IEEE Trans. Smart Grid* 11 (4), 3620–3631. doi:10.1109/tsg.2019.2963696

Wang, J., Wang, W., Yuan, Z., Wang, H., and Wu, J. (2020). A chaos disturbed beetle antennae search algorithm for a multiobjective distribution network reconfiguration considering the variation of load and DG. *IEEE Access* 8, 97392–97407. doi:10.1109/access.2020.2997378

Wu, Y., Liu, Y., Li, N., and Wang, S. (2021a). "Hybrid multi-objective particle swarm optimization algorithm based on particle sorting," in IEEE International Conference on Emergency Science and Information Technology (ICESIT), Chongqing, China, 22-24 November 2021, 257–260.

Wu, Y., Zhang, P., and Lu, G. (2021b). Detection and location of aged cable segment in underground power distribution system using deep learning approach. *IEEE Trans. Industrial Inf.* 17 (11), 7379–7389. doi:10.1109/tii.2021.3056993

Xie, H., Qin, Z., Li, G., and Juang, B. (2021). Deep learning enabled semantic communication systems. *IEEE Trans. Signal Process.* 69, 2663–2675. doi:10.1109/tsp.2021.3071210

Yang, T., George, J., Qin, J., Yi, X., and Wu, J. (2020). Distributed least squares solver for network linear equations. *Automatica* 113, 108798. doi:10.1016/j.automatica.2019.108798

Yi, X., Tian, M., Chen, C., and Zhang, G. (2021). Reactive power optimization and reconfiguration of distribution network based on improved gsa algorithm. *IEEE/IAS Industrial Commer. Power Syst. Asia (I&CPS Asia)* 2021, 674–679.

Yin, Z., Ji, X., Zhang, Y., Liu, Q., and Bai, X. (2020). Data-driven approach for re-al-time distribution network reconfiguration. *IET Gen-eration, Transm. Distribution* 14 (13), 2450–2463. doi:10.1049/iet-gtd.2019.1733

Zainab, A., Ghrayeb, A., Syed, D., Abu-Rub, H., Refaat, S. S., and Bouhali, O. (2021). Big data management in smart grids: Technologies and challenges. *IEEE Access* 9, 73046–73059. doi:10.1109/access.2021.3080433

Zhan, J., Liu, W., Chung, C., and Yang, J. (2020). Switch opening and exchange method for stochastic distribution network re-configuration. *IEEE Trans. Smart Grid* 11 (4), 2995–3007. doi:10.1109/tsg.2020.2974922

Zhang, N., Sun, Q., Yang, L., and Li, Y. (2022). Event-triggered distributed hybrid control scheme for the integrated energy system. *IEEE Trans. Industrial Inf.* 2022 (18-2), 835–846. doi:10.1109/tii.2021.3075718

Zhe, J., Jing, Z., Yu, C., Xi, F., Wu, F., Tao, Z., et al. (2020). Reliability analysis of distribution network operation based on short-term future big data technology. *J. Phys. Conf. Ser.* 2020 (1), 012027. doi:10.1088/1742-6596/1584/1/012027

Zheng, W., Huang, W., Hill, D. J., and Hou, Y. (2020). An adaptive distributionally robust model for three-phase distribution network reconfiguration. *IEEE Trans. Smart Grid* 12 (2), 1224–1237. doi:10.1109/tsg.2020.3030299

Zheng, W., Huang, W., Hill, D. J., and Hou, Y. (2021). An adaptive distributionally robust model for three-phase distribution network reconfiguration. *IEEE Trans. Smart Grid* 12 (2), 1224–1237. doi:10.1109/tsg.2020.3030299

Nomenclature

Sets

$\mathbf{O}^{\mathbf{con}}$ Final output of the network

$\mathbf{O}_l$ Model output of loop l

$\boldsymbol{L}^c$ Calculated loss of the model

$\boldsymbol{L}_l$ Loss value returned by submodel l

$\boldsymbol{\Lambda}$ Parameter matrix

$\boldsymbol{C}_h$ The h-th switch configuration scheme of the system

Parameters

$\boldsymbol{c}$ Total number of categories

$\boldsymbol{k}$ Number of batch samples

$\boldsymbol{y}_{m,n}$ The true probability of sample m on classification n

$\boldsymbol{\alpha}$ Learning rate

$\boldsymbol{\beta}_1/\boldsymbol{\beta}_2$ The exponential decay rates of the first and second moments respectively

$\boldsymbol{g}_t^c$ Gradient of loss value

$\boldsymbol{L}_t^c$ To weight $\boldsymbol{\theta}$

LM The number of load modes

$\boldsymbol{N}_b$ Total number of system branches

$\boldsymbol{\Gamma}$ Set of all connected network structures

$\boldsymbol{V}_i{}^{\mathbf{min}}/\boldsymbol{V}_i{}^{\mathbf{max}}$ The lower and upper voltage limits of bus i

$\boldsymbol{S}_b{}^{\mathbf{max}}$ The upper limit of branch b capacity

$\boldsymbol{N}$ Total number of buses in the system

$\boldsymbol{H}_{1,h}/\boldsymbol{H}_{2,h}/\boldsymbol{H}_{3,h}$ The one-hot codes of loops 1, 2, and 3

Variables

$\boldsymbol{Z}_\eta$ Output of neuron η

$\boldsymbol{z}_\eta$ Input of neuron η

$\boldsymbol{L}(\boldsymbol{\theta})$ Model loss under weight θ

$\boldsymbol{t}$ Time step

$\boldsymbol{g}_t$ Gradient of $L_t(\theta_{t-1})$ with respect to θ_{t-1}

$\hat{\boldsymbol{m}}_t$ The correction of the first moment estimation of gradient m_t

$\hat{\boldsymbol{v}}_t$ The correction of the second moment estimation of gradient v_t

$\boldsymbol{E}^{\mathbf{set}}$ The default value of the training cycle

$\boldsymbol{E}$ The predicted loss under the current training cycle

$\boldsymbol{L}_E$ Binary variable gas turbines start up

$\boldsymbol{E}^{\mathbf{lim}}$ The cycle limit of the callback function

$\boldsymbol{K}_b$ The 0–1 variable representing the open state of branch b

$\boldsymbol{R}_b$ The resistance of branch b

$\boldsymbol{I}_b$ The current flowing through the branch

$\boldsymbol{\varphi}$ Network structure obtained through reconfiguration

$\boldsymbol{P}_i/\boldsymbol{Q}_i$ The injected active power and reactive power of bus i

$\boldsymbol{P}_i^{\mathbf{DG}}/\boldsymbol{Q}_i^{\mathbf{DG}}$ The active power and reactive power injected by DG into bus i respectively

$\boldsymbol{V}_i/\boldsymbol{V}_j$ The voltages of buses i and j respectively

$\boldsymbol{G}_{ij}/\boldsymbol{B}_{ij}/\boldsymbol{\delta}_{ij}$ The conductance, susceptance and phase Angle difference between ij buses respectively

$\boldsymbol{S}_b$ Apparent power of branch b

$\boldsymbol{F}_i^{\mathbf{DG}}$ DG access status of bus i

$\boldsymbol{P}_i^{\mathbf{DG}}$ Active output of bus i access DG

$\boldsymbol{V}_i^{\mathbf{PV}}$ Voltage unit value of PV type DG

$\boldsymbol{I}_i^{\mathbf{PI}}$ Current unit value of PI type DG

$\boldsymbol{T}_i^{\mathbf{DG}}$ DG type of A-Bus i

$\boldsymbol{P}_i^{\mathbf{L}}/\boldsymbol{Q}_i^{\mathbf{L}}$ Load active and reactive power of bus i respectively

$\mathbf{A}_a$ row a of model input matrix A

$\boldsymbol{\Lambda}_a$ row a of parameter matrix $\boldsymbol{\Lambda}$

$\boldsymbol{\lambda}_a$ Maximum number of elements in $\boldsymbol{\Lambda}_a$