



OPEN ACCESS

EDITED BY

Hamid Reza Karimi,
Polytechnic University of Milan, Italy

REVIEWED BY

Yanfeng Peng,
Hunan University of Science and Engineering,
China
Xueshuai Zhang,
Chinese Academy of Sciences (CAS), China

*CORRESPONDENCE

Yi Cai,
✉ mailee2sjtu@163.com

RECEIVED 04 January 2025

ACCEPTED 19 February 2025

PUBLISHED 17 March 2025

CITATION

Jiang L, Li Y, Yang D, Hu J, Xu J, Xie J and Cai Y
(2025) Cable laying conveyor state prediction
and fault warning based on attention-driven
CNN-LSTM algorithm.
Front. Mech. Eng. 11:1549053.
doi: 10.3389/fmech.2025.1549053

COPYRIGHT

© 2025 Jiang, Li, Yang, Hu, Xu, Xie and Cai. This
is an open-access article distributed under the
terms of the [Creative Commons Attribution
License \(CC BY\)](https://creativecommons.org/licenses/by/4.0/). The use, distribution or
reproduction in other forums is permitted,
provided the original author(s) and the
copyright owner(s) are credited and that the
original publication in this journal is cited, in
accordance with accepted academic practice.
No use, distribution or reproduction is
permitted which does not comply with these
terms.

Cable laying conveyor state prediction and fault warning based on attention-driven CNN-LSTM algorithm

Leyun Jiang¹, Ye Li¹, Dongxing Yang¹, Jian Hu¹, Jun Xu¹,
Jianghong Xie¹ and Yi Cai^{2*}

¹State Grid Wuxi Power Supply Company, Wuxi, China, ²Department of Electrical Engineering, Shanghai Jiao Tong University, Shanghai, China

The monitoring condition of the cable laying conveyor, such as rotational speed, driving current, and side pressure, reflects the real-time operation status of the cable laying process. Accurate prediction of multiple monitoring states of cable conveyors can assess the status of cable laying in advance and avoid failure. The existing cable laying construction mainly relies on the threshold value to determine the safety status. It rarely predicts the state and does not consider the connection between the various monitoring states, so it is difficult to make accurate predictions. For this reason, this paper proposes an attention-driven Convolutional Neural Network-Long Short Term Memory (A-CNN-LSTM) algorithm for multi-state prediction and fault warning of cable conveyor, which explores the relationship between the states of the cable conveyor and makes a more accurate prediction. CNN is used to mine the connection between the states of the cable conveyor and the attention mechanism is used to intelligently allocate weights. LSTM is used to explore the law of the states of the conveyor over time, and use the attention mechanism to intelligently allocate weights in the time step, which ultimately realizes the prediction. The method is applied to a 110 kV cable laying experiment and compared with the prediction results of the widely used TCN algorithm, and the CNN-RNN algorithm without attention mechanism, which shows that the proposed attention-driven prediction algorithm has higher accuracy, better reflects the connection between multiple monitoring states of the cable conveyor, and performs more accurate prediction.

KEYWORDS

cable laying, conveyor, attention-driven CNN-LSTM, prediction, early warning

1 Introduction

Power cables realize transmission and distribution of large-capacity electric energy by underground compact operation in the power system, which has developed rapidly in recent years due to its reliability, aesthetics, and excellent electrical performance, and has gradually become the mainstream transmission form of urban power supply (Ghorbani et al., 2014; Diban et al., 2022; Montanari et al., 2018; Zhu et al., 2019; Wang et al., 2018). In Wuxi, Jiangsu Province, China, for example, by the end of 2021, 110 kV high-voltage cables reached 5,870 km and maintained an annual growth rate of more than 15%, the development of power cable engineering is related to the development of urban power transmission network, is a key component of the future power grid infrastructure. The

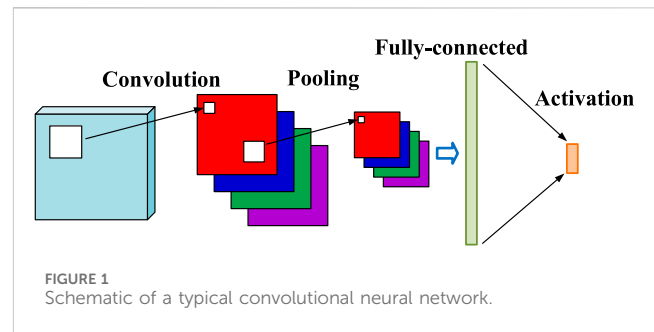
current stage of the high-voltage cable construction process is in urgent need of intelligent digital transformation, in order to meet the requirements of the State Grid Corporation to build a “world-class power grid” in the future.

Cable laying is one of the most important construction links in the construction of electric power projects, and the quality of power cable laying and laying utility directly affect the quality of electric power project construction (Choi et al., 2018). Ensuring the reliability of cable laying technology and quality of the whole power project stability, safety and reliability is of great significance. Power cable laying methods mainly include direct burial, cable trench laying, pipe laying, bridge laying, and so on. Direct burial cable laying depth is generally more than 0.7 m. Because of its low investment costs and other outstanding advantages, it has become one of the widely used laying methods. Cable trench laying needs to set up metal brackets. When the cable is too much, the metal brackets on both sides of the cable are needed. Cable pipe laying is used when the cable needs to pass through the asbestos cement pipe, plastic pipe, and concrete pipe, and then according to the corresponding order of laying to the well to make the cable is not harmed. In case of improper management of the laying process in the above cable laying process, the cable body and the laying equipment may be damaged, affecting the reliability of subsequent cable operation.

At present, the research on cable laying control and early warning is relatively weak. The control strategy is only the open-loop mode of positive-negative-start-stop. The control basis relies on the judgment of the workers, the operation precision is low, and the synergy between the conveyors is poor. In terms of the completeness of control, the system lacks effective control of key variables such as lateral pressure and traction, and it relies on the threshold judgment or the manual emergency stop when the laying fault occurs. Therefore, the construction reliability and emergency response capability need to be further strengthened and the emergency response capability needs to be further strengthened.

A cable-laying conveyor is an important device to promote cable transportation, generally using a motor-driven crawler to move cables. Cable conveyor motor speed, current, side pressure, and other states can be monitored to reflect the condition of cable laying and transportation process, to evaluate the laying fault, and to perform early warning. Cable laying states have a certain correlation with each other, such as motor speed can reflect the cable conveying speed, motor voltage and current can reflect the output of the motor, the side pressure of the conveyor in contact with the cable can reflect the bearing force of the cable outer sheath. Therefore, it is feasible to use sensors to monitor the above states and further synthesize and analyze the effective information contained in multiple monitoring states to predict the trend and then provide early warning of cable laying faults that have not occurred.

Currently, forecasting methods for time series data can be mainly categorized into statistical methods and artificial intelligence methods (Mirowski and LeCun, 2012; Wang et al., 2011; Zhao et al., 2019; Yang et al., 2023). Statistical methods predict future trends through the statistical characteristics of time series (Sun et al., 2022; Arora et al., 2023), which mainly include autoregression (AR), multiple linear regression (MLR), auto-regressive moving averages (ARMA), etc. Statistical methods require that the data to be processed have a large scale, and the data itself needs to show a certain pattern to make a recursive prediction of the monitoring volume. With the development of artificial intelligence, scholars have gradually applied



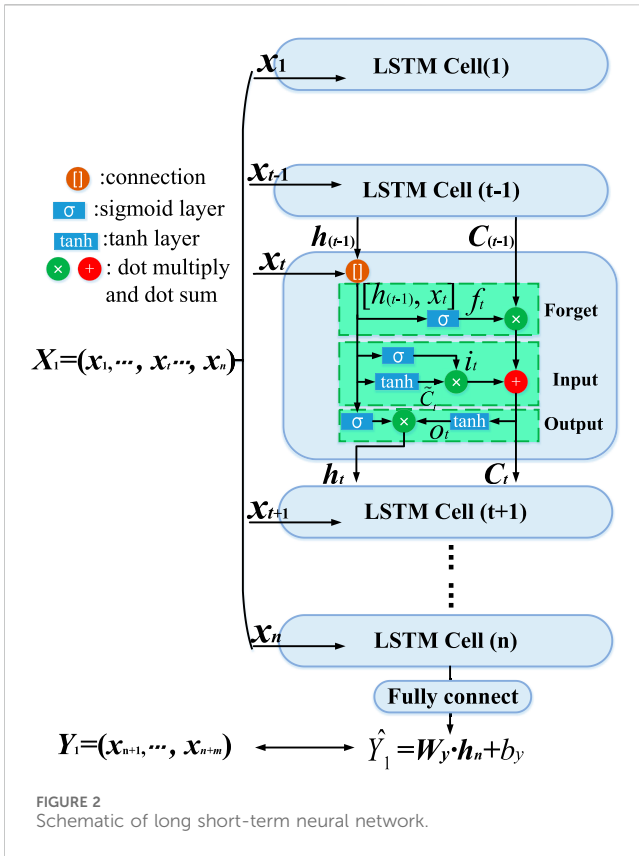
artificial intelligence algorithms to equipment condition assessment and prediction. Artificial intelligence prediction methods are mainly based on training and learning through the linear or nonlinear relationship of a large number of monitoring historical data and updating models continuously reducing the gap between the training output and the expected output (Yang et al., 2023). Conventional methods include conventional neural networks (NN) (Ge et al., 2021), support vector machine (SVM) (Li and Ma, 2022), and restricted Boltzmann machine (RBM) (Hou and Han, 2010). However, conventional AI methods are insensitive to the temporal correlation of sequence data and have very limited applications. The emergence of deep learning models such as convolutional neural networks (CNN), recurrent neural networks (RNN), and their variants has solved the widespread problems of gradient vanishing and gradient explosion and has been widely used in the fields of feature extraction and time series prediction. However, the above time series models only predict a single parameter at a time, focusing on the correlation of monitoring states on the time scale but ignoring the physical correlation between monitoring states in the actual operation of the laying equipment. Therefore, they fail to predict the mutation of multiple monitoring states through the change of correlated monitoring states. Due to the fast-thriving development of machine learning, algorithms such as center jumping boosting machine (CJBM) (Li et al., 2023), temporal convolutional networks (TCN) (Hewage et al., 2020), and multilayer perceptron (Fan et al., 2024) emerge endlessly.

To make full use of the correlation between the monitoring states of the cable-laying conveyor and the dependence of the timing information, and to solve the problem of state prediction and fault warning in cable laying, this paper proposes an attention-driven CNN-LSTM algorithm for predicting the states of the cable-laying conveyor. CNN is used to excavate the connection between the states of the conveyor, and the attention mechanism is used to allocate the weights of each variable intelligently. LSTM is used to excavate the states over time, and the attention mechanism also intelligently allocates the time-step weights of each variable and ultimately realizes the prediction of the states of the cable-laying conveyor to provide a reference for fault early warning.

2 Algorithm background

2.1 CNN

CNNs are widely used in image and video recognition tasks, and similar to traditional neural networks, CNNs are composed of numerous neurons, each of which receives inputs and performs



scalar product as well as nonlinear activation function processing. CNNs generally consist of a stack of convolutional, pooling, and fully connected layers, as shown in Figure 1. As the name suggests, the convolutional layer plays a crucial role in the way the CNN operates. The convolutional layer parameters focus on learnable convolutional kernels, which are usually small in spatial dimensions and are convolved on each filter in the spatial dimensions of the input as the data reaches the convolutional layer. The convolutional layer is used to extract local features of the image or data, and different convolutional kernels with different dimensions and parameters are used as feature extractors of different sizes and types. The pooling layer aims to gradually reduce the dimensionality of the result after convolution, thus further reducing the number of parameters and the computational complexity of the model and avoiding overfitting. In most CNNs, the pooling layer appears in the form of an extracted maximum. The fully connected layer contains a layer of neurons that, similar to traditional neural networks, are directly connected to neurons in neighboring layers. The fully connected layer performs a linear or nonlinear transformation for the extracted features, generally through an activation function as the final output of the classifier.

2.2 LSTM

LSTM is one of the most widely used variants of recurrent neural networks (RNN). The advantage of RNN is that the output of the front part of the network can be used as an input to affect the neurons in the back part of the network again so that the information of earlier time

series can be preserved and acted upon in the subsequent time series (Hochreiter and Schmidhuber, 1997), which makes it more suitable for the prediction of time sequence data. However, the disadvantage of RNN is that it is difficult to preserve long time history information, LSTM improves the problem of dissipation of the effect of long-time data that exists in RNN by adding a gating mechanism and also avoids the problem of gradient vanishing and gradient explosion in RNN during training.

LSTM learns long-term dependencies from historical data through a recurrent structure and an introduced gating mechanism. To illustrate how LSTM works, suppose n historical data are used to predict the next m data, then each n historical data can be regarded as an input vector and the next m data are the actual output vectors, thus constituting a training pair, e.g., the input vectors $X_1 = (x_1, \dots, x_n)$ and $Y_1 = (x_{n+1}, \dots, x_{n+m})$. The basic LSTM cell structure is shown in Figure 2, where the training data are progressively fed into the LSTM cell to compute the hidden state vectors h_t and the cell state vectors C_t . The flow of data within and between cells is controlled by three gates: the forget gate, the input gate, and the output gate (green rectangles with dashed lines in Figure 2).

The forget gate decides which information to discard. The input vector x_t is connected to the hidden state vector h_{t-1} at the last time step, and the forget gate output is computed by Equation 1.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (1)$$

where W_f and b_f are the weight matrix and bias of the forget gate, which contains the trainable variables. σ is a Sigmoid function that outputs a number between zero and one, describing each component that can be passed through the LSTM cell. The next step is to determine what new information should be stored in the LSTM cell state vector for the input gate. This process involves combining the information from the current input with the information contained in the previous hidden state vector $h(t)$. the sigmoid layer decides which values will be updated, and the tanh layer creates a new candidate value that can be added to the LSTM cell state vector (Li et al., 2023) as shown in Equation 2.

$$\begin{cases} i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \\ \tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \\ C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \end{cases} \quad (2)$$

where W_i , W_C , b_i , and b_C are weight matrices and biases of the same dimension with trainable variables. The “ $*$ ” symbol indicates a dot product, which does not change the dimension of the unit state vector. Finally, the output gate will determine the output of the LSTM cell based on the current LSTM cell state and input information (Li et al., 2023) as shown in Equation 3.

$$\begin{cases} o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \\ h_t = o_t * \tanh(C_t) \end{cases} \quad (3)$$

where W_o and b_o are the weight matrix and bias of the output gate containing the trainable variables. When all the input vectors are fed to the LSTM unit, the last hidden state vector will go through a fully connected layer to obtain the final predicted output data as shown in Equation 4.

$$\hat{Y} = W_y \cdot h_n + b_y \tag{4}$$

where W_y and b_y are the weight matrix and bias respectively. The error between the predicted output and the actual output is represented by the loss function and is used to update the weight and bias matrices so that the output of the LSTM model is as close as possible to the actual output.

2.3 Attenuation mechanism

Attention mechanisms have been widely used in various areas of deep learning in recent years, whether in image processing, speech recognition, or natural language processing for a variety of different types of tasks. The attention mechanism in deep learning is essentially similar to the human selective visual attention mechanism, and the core goal is also to select the information that is more critical to the current task goal from a large amount of information.

The general process of the attention mechanism is to assign weights, the attention mechanism is often used in the structure of the self-encoder, the encoder according to the input vector x_t through the fully connected layer and softmax layer to calculate the attention weight α_t , which indicates that the different input features of the different degree of importance of the output. The attention weights combine input vectors, to generate a new input vector s_t imported into the subsequent neural network model, as shown in Figure 3. If different time-step input vectors of the RNN model are processed with the attention mechanism and then imported into it, the degree of influence of different time steps on the output prediction can be reflected. The attention mechanism can be expressed by Equation 5.

$$\begin{cases} \alpha_t = \text{softmax}(\text{Dense}(x_t)) \\ s_t = \text{merge}(\alpha_t, x_t) \end{cases} \tag{5}$$

where α is the attention weight, s is the result after the application of attention, and the merge operation uses the weights to multiply with the input variables.

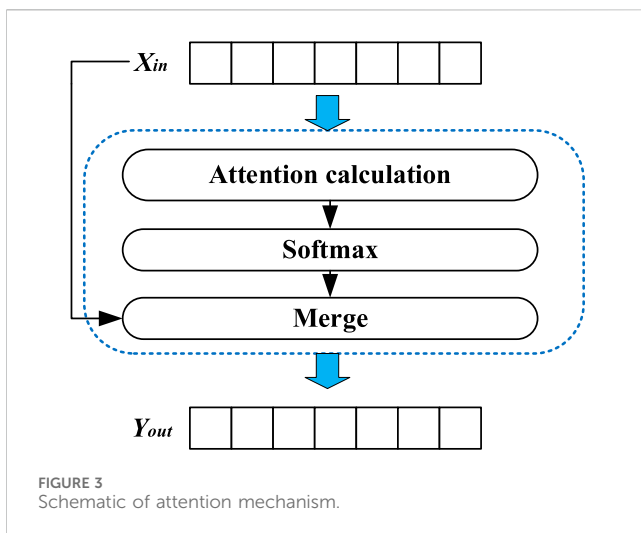


FIGURE 3 Schematic of attention mechanism.

3 Attention-driven CNN-LSTM state prediction algorithm for cable conveyor

3.1 Multivariate prediction problems

Let there are n monitoring quantities that need to be predicted, the value of each monitoring quantity at different moments forms a one-dimensional column vector, and the subscripts are used to indicate the value of each monitoring state at different moments, e.g., x_1^2 indicates the monitoring value of the second monitoring state at the first sampling moment, and n monitoring states containing m time steps can be represented by the matrix X as Equation 6.

$$X = [X_1 \ X_2 \ \dots \ X_m] = \begin{bmatrix} x_1^1 & x_2^1 & \dots & x_m^1 \\ x_1^2 & x_2^2 & \dots & x_m^2 \\ \vdots & \vdots & \vdots & \vdots \\ x_1^n & x_2^n & \dots & x_m^n \end{bmatrix} \tag{6}$$

If the data of multiple monitoring states from the first m time steps are utilized to predict the $m+1$ time step, the expected output is the vector $Y = (x_{m+1}^1, x_{m+1}^2, \dots, x_{m+1}^n)$. In this way, X and Y form a training pair, and the training pair constitutes the training dataset and the validation dataset.

3.2 Algorithm description

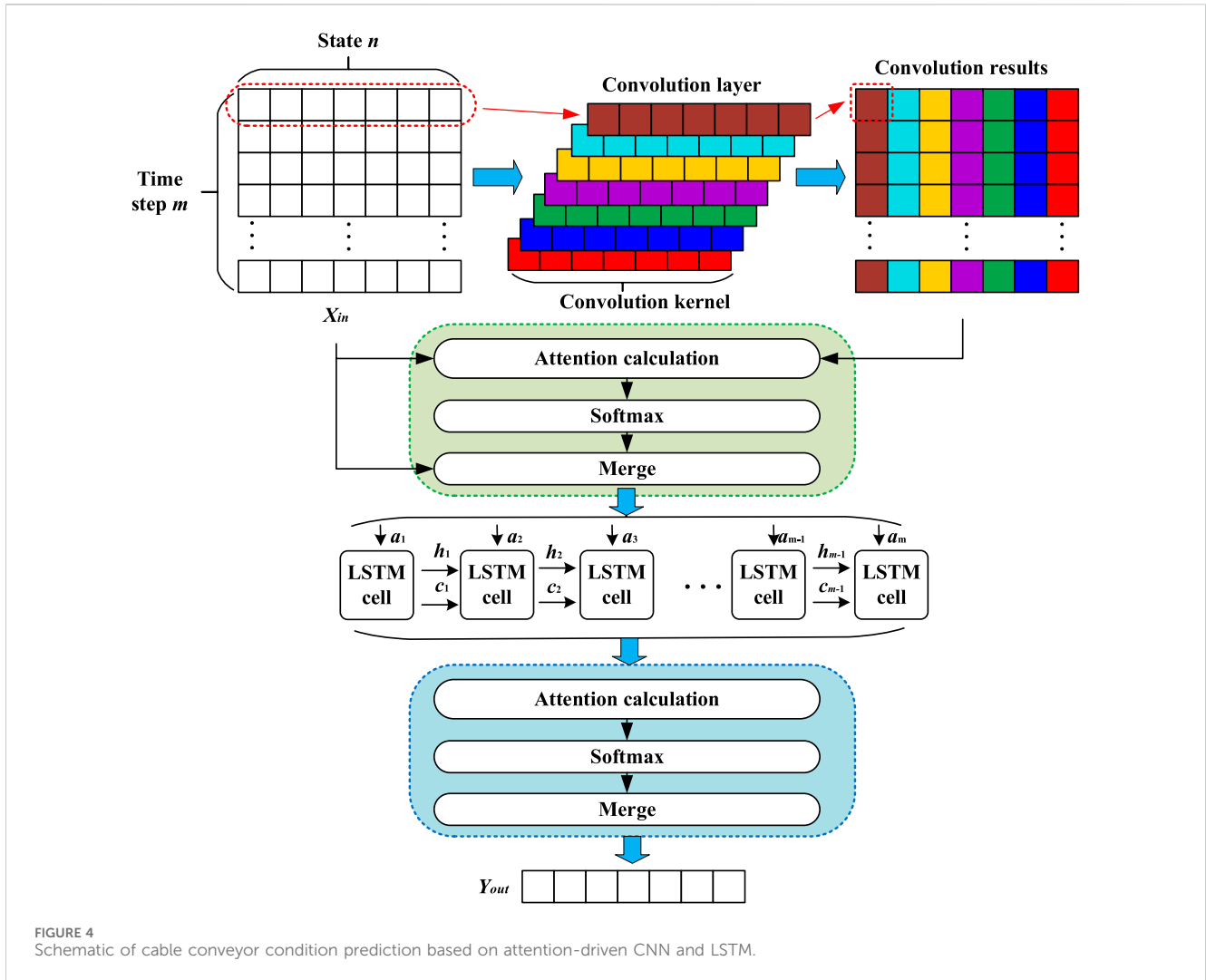
To fully explore the correlation between the monitoring states of cable conveyors and the dependence of the data at different moments of each monitoring state, and accurately predict the change of multiple monitoring states, this paper proposes a prediction model for multiple monitoring states of cable conveyor based on the attention-driven CNN-LSTM. The algorithm structure of the model proposed in this paper is shown in Figure 4. To facilitate the understanding of the overall structure of the algorithm, the proposed algorithm is introduced with a set of training pairs as an example. The proposed multi-monitoring state prediction algorithm for cable conveyors based on attention-driven CNN-LSTM contains a CNN module, a monitoring state attention module, an LSTM module, and a time-step attention module.

Firstly, a two-dimensional matrix is constructed for n monitoring states at m time steps, and n one-dimensional convolution kernels are used to convolve the input matrix X . The convolution is utilized to compute the links between the n monitoring states of the cable conveyor, and the results of different convolution kernel computations represent the degree of influence of a certain monitoring state on other monitoring states. The convolutional layer operation can be represented by Equation 7.

$$j_t^l = f_c \left(\sum_{i=1}^n x_t^i * k_t^i + b_t^l \right) \tag{7}$$

where j is the post-convolution matrix, f_c is the activation function, k is the convolution kernel weight, and b is the convolution kernel bias.

The post-convolution result is input into the feature attention module along with the original input matrix X . The degree of



influence of each input monitoring state on the other monitoring states at each moment is obtained by attention computation and softmax layer activation which makes the sum of all the weights to be one. Afterward, the attention weights are combined with the input monitoring quantities by weight multiplication fusion operation to obtain the monitoring quantity s_t that contains the connection between each monitoring state as shown in Equation 8.

$$\begin{cases} \epsilon_t = V_\alpha^T \tanh(W_\alpha j_t + U_\alpha x_t) \\ \alpha_t^i = \text{softmax}(\epsilon_t^i) = \frac{\exp(\epsilon_t^i)}{\sum_{i=1}^n \exp(\epsilon_t^i)} s_t^i = \text{merge}(\alpha_t^i, x_t^i) \end{cases} \quad (8)$$

where α is the attention weight, V_α , W_α , and U_α are the full connection weight parameters, and s is the output after the monitoring quantity attention calculation.

The monitoring attention output s_t is fed into the LSTM network by each time step, and the learning is trained to obtain the connection between the time steps to make predictions about the monitoring volume at the subsequent time steps. The computation of the LSTM layer is similar to Equations 1–3, where the input x_t is

replaced by the output s_t that has been computed by the monitoring attention as shown in Equation 9.

$$\begin{cases} f_t = \sigma(W_f \cdot [h_{t-1}, s_t] + b_f) \\ i_t = \sigma(W_i \cdot [h_{t-1}, s_t] + b_i) \\ \tilde{C}_t = \tanh(W_C \cdot [h_{t-1}, s_t] + b_C) \\ C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \\ o_t = \sigma(W_o \cdot [h_{t-1}, s_t] + b_o) \\ h_t = o_t * \tanh(C_t) \end{cases} \quad (9)$$

Where f_t , i_t , and o_t are the outputs of the forgetting, input, and output gates at time t . h_t and C_t are the hidden state vector and the cell state vector, and W and b are the trainable weight matrices of each gating control as well as the bias.

To synthesize the impact of each monitoring state at different time steps on the data at future prediction time steps, the hidden state vector h_t and the cell state vector C_t of the LSTM network are intelligently deployed with time-step weights using the attention mechanism. The degree of influence of each monitoring state at different moments on the monitoring states at other moments is obtained by attention calculation and softmax layer activation to make the sum of all weights 1. After that, the attention weights are combined with the input moment monitoring states through the

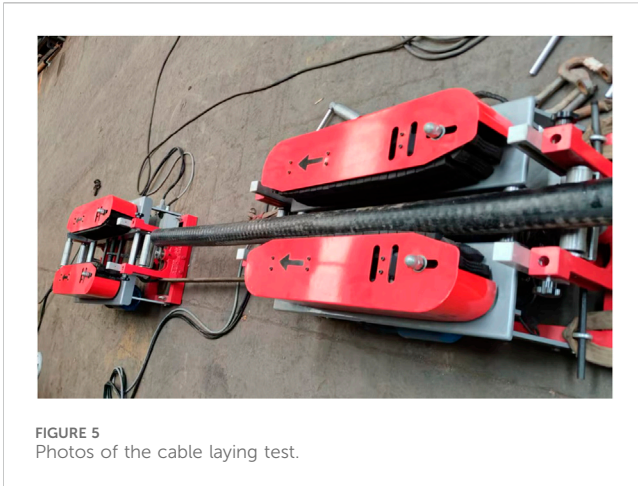


FIGURE 5 Photos of the cable laying test.

weight multiplication fusion operation to obtain the output that contains the connection between different moments of each monitoring state. Finally, through the full connectivity layer, the prediction of multi-monitoring states of the conveyor. The time step attention is calculated by Equation 10.

$$\left\{ \begin{array}{l} \gamma_t = \mathbf{V}_\beta^T \tanh(\mathbf{W}_\beta [\mathbf{C}_{t-1}, \mathbf{h}_{t-1}] + \mathbf{U}_\beta \mathbf{s}_t) \\ \beta_t^i = \text{softmax}(\gamma_t^i) = \frac{\exp(\gamma_t^i)}{\sum_{i=1}^n \exp(\gamma_t^i)} \\ r_t^i = \text{merge}(\beta_t^i, s_t^i) \\ \hat{y}_i = \sigma(\mathbf{W}_d \mathbf{r}_t + b_d) \end{array} \right. \quad (10)$$

where β is the attention weight, \mathbf{V}_β , \mathbf{W}_β , \mathbf{U}_β , \mathbf{W}_d and b_d are the full connectivity parameters, \mathbf{r} is the output after time-step attention calculation, and \hat{y} is the final predicted output. The quadratic loss function L is used to evaluate the gap between the predicted output and the actual output, and then the back-propagation gradient function is used to update and adjust the parameters of the algorithm as shown in Equation 11.

$$L = \sum_{i=1}^n (\hat{y}_i - y_i)^2 \quad (11)$$

4 Example analysis

To verify the effectiveness of the algorithm proposed in this paper, a total of six states of motor speed, current, and side pressure are monitored on two adjacent cable conveyors during 110 kV cable laying, (recorded as speed 1, current 1, pressure 1, speed 2, current 2, pressure 2, respectively). They are set up to simulate the three fault modes of cable jamming, cable delivery asynchrony, and excessive pressure, and the data before and after the three faults are monitored and collected for the training and testing of the model proposed in this paper. The experimental site photos are shown in Figure 5.

A total of 200 sets of three fault simulation data are used to construct the training and validation sets according to the data

TABLE 1 Errors of the proposed model with different time steps.

Time step	5	10	20	40
RMSE	28.387	28.226	28.014	27.815
MAPE (%)	1.049	1.004	1.001	0.973

structure described in the previous section, randomly disrupting the order to avoid systematic errors caused by artificial division, so that the model prediction has wide applicability.

4.1 Model hyperparameter study

The type of monitoring state for the algorithm is $n = 6$ and the time steps are set to $m = 5, 10, 20$ and 40 to investigate the effect of different time steps on the prediction accuracy. The proposed algorithm for cable laying conveyor based on attention-driven CNN-LSTM contains a CNN module, monitoring state attention module, LSTM module, and time step attention module. For the CNN module, six one-dimensional convolution kernels are used to extract monitoring state features from the input matrix, and the size of the convolution kernels is 6×1 . The monitoring state attention module performs attention weight calculation on the features extracted after convolution and the original monitoring state and then multiplies them by the original monitoring state. The LSTM module uses a single-layer LSTM cell, and the number of hidden cells is 64. The time-step attention module performs attention weight calculation on the state of the LSTM cell and the hidden state, and the time-step attention module performs attention weight calculation on the LSTM cell state and the hidden state. Then they are dot-multiply with the monitoring data processed by the monitoring state attention module, and finally output six predicted monitoring states through the fully connected layer. The proposed model uses Adam's algorithm to calculate the backpropagation gradient and update the training parameters, the number of training iterations is 100, and the learning rate uses exponential decay to prevent oscillations, with an initial learning rate of 0.01, a decay rate of 0.96, and a decay step size of 100 steps.

In this paper, two indicators, relative mean square error (RMSE) and mean absolute percentage error (MAPE), are selected to evaluate the effect of model prediction as shown in Equations 12, 13. The smaller the value of both, the more accurate the prediction result.

$$e_{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2} \quad (12)$$

$$e_{MAE} = \frac{1}{n} \sum_{i=1}^n \left| \frac{y_i - \hat{y}_i}{y_i} \right| \quad (13)$$

The computational input model predicts the motor speed, current, and side pressure data of the two conveyors, and the prediction results of the input model with different time steps are compared. Taking the motor speed as an example, the results are shown in Table 1. It can be seen that the longer the input time step, i.e., the more historical data provided by the prediction, the more accurate the prediction results are.

TABLE 2 Errors of the proposed model with different hidden units.

Hidden unit number	8	16	32	64
RMSE	28.200	28.186	28.022	27.945
MAPE (%)	1.002	0.993	0.984	0.973

In addition, the prediction results are also compared for the number of LSTM hidden units, controlling the other parameters unchanged, selecting the best 40 time steps, and setting the number of hidden units to 8, 16, 32, and 64 respectively. Taking the motor speed as an example, the prediction errors are shown in Table 2. It can be seen that the more the number of hidden units, the more accurate the prediction results, but the overall difference in accuracy is not large. This is because more number of hidden units can perform more complex nonlinearity calculations and can learn more complex laws.

4.2 Relevance analysis and attention mechanism validation

The motor speed, current, and side pressure of a conveyor during cable laying have correlation characteristics. The motor speed and current of the conveyor are conventionally positively correlated, When the motor is blocked or the load is increased, the current increases and the speed decreases. When the two conveyors are not synchronized, the cable sheath and the conveyor are displaced relative to each other, which can further damage the cable sheath. When the cable is bent too much or there is additional force, the side pressure of the conveyor changes, which can reflect the abnormal force damage during cable laying.

To investigate the correlation between conveyor state quantities during cable laying, the correlation of each monitored state quantity was analyzed using the Pearson correlation coefficient as shown in Equation 14.

$$Corr = \frac{N \sum_{i=1}^N x_i y_i - \sum_{i=1}^N x_i \sum_{i=1}^N y_i}{\sqrt{N \sum_{i=1}^N x_i^2 - \left(\sum_{i=1}^N x_i\right)^2} \sqrt{N \sum_{i=1}^N y_i^2 - \left(\sum_{i=1}^N y_i\right)^2}} \quad (14)$$

where x and y are two kinds of measurement data respectively and N is the number of data.

The heat map of correlation coefficients of conveyor states among each other during cable laying is shown in Figure 6. The lighter color of the color block in the graph represents the higher degree of correlation. It can be seen that each variable presents different correlation characteristics except that each variable is completely correlated with itself. The higher brightness of the cable conveyor motor speed and current squares shows a stronger correlation. In addition, the strong correlation between the same type of state quantities of the two conveyors indicates that the cable laying situation can be simultaneously represented in the states of the two neighboring conveyors when the two conveyors are transporting cables in series.

To verify whether the attention mechanism can reflect the relationship between the states, the CNN attention correlation coefficients in the last step of the prediction are calculated, and the attention weight α in Equation 8 is calculated as the final correlation score between the state quantities to be predicted and the other relevant feature coefficients, and the correlation is quantified in this way. The heat map of the calculation result is shown in Figure 7. According to the calculation rule of attention weight, the sum of all correlation weights in each column is 1. Since all the information of each state quantity itself is retained, the attention weight is calculated without counting itself, and

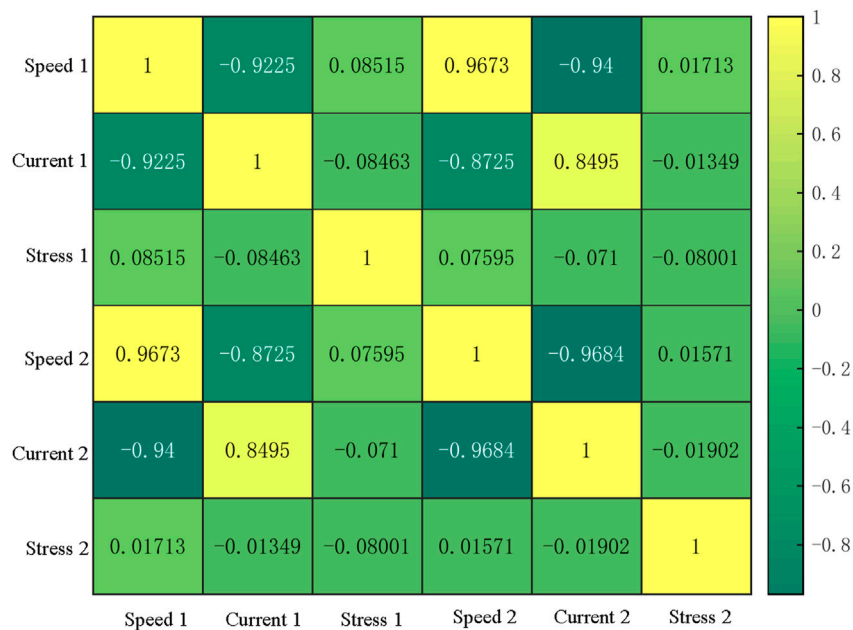


FIGURE 6 The calculation results of Pearson relation coefficients of cable conveyor states.

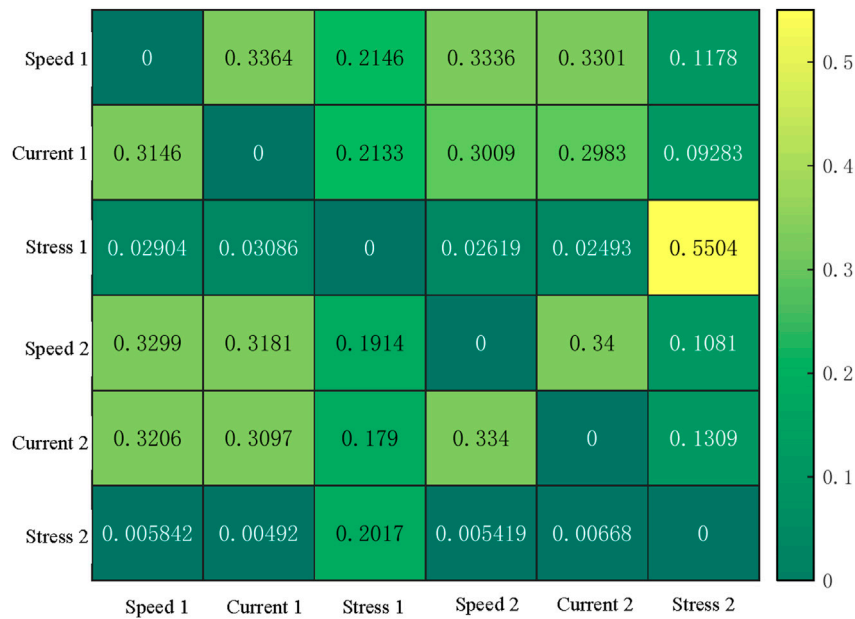


FIGURE 7 Calculation results of attention weights for each state of cable conveyor.

the color of each of the remaining squares indicates the correlation corresponding to other state quantities. The lighter the color in a square, the stronger the correlation between the two parameters corresponding to that square.

It can be seen that the overall color distributions presented in Figures 6, 7 are the same, indicating a consistent relationship between the distribution of attention weights and the correlation of the data, which shows that the introduced attention weight mechanism can characterize the correlation of each state parameter of the conveyor during cable laying.

4.3 Comparison of single-step prediction results

To compare the effectiveness of the proposed attention algorithm with widely used prediction algorithms, two representative AI algorithms for data prediction tasks, the TCN and CNN-LSTM algorithms without using the attention mechanism, are selected and compared with the attention-driven CNN-LSTM algorithm proposed in this paper. The convolutional kernel size of TCN is set to 6, and the number of hidden layers is set to 4, and the number of hidden units in the CNN-LSTM algorithm without the attention mechanism is 64. The remaining parameters of the CNN-LSTM algorithm are the same as the algorithm proposed in this paper. To compare under the same conditions, the time steps are all set to 20 to predict the next data. The Adam algorithm is used to compute the backpropagation gradient and update the training parameters, the number of training iterations is 1,000, and the learning rate is exponentially decayed to prevent convergence oscillations, with an initial learning rate of 0.01, a decay rate of 0.96, and a decay step size of 1,000 steps.

The prediction results of the three methods for the six states of the two cable conveyors under three fault states are shown in Figure 8. For

the cable jamming fault, the motor speed of the two conveyors dropped to zero relatively quickly after the fault and was accompanied by motor blocking and a rapid increase in current. The pressure monitoring shows no major changes. Different prediction algorithms show different prediction effects for the cable jamming fault data. All three algorithms were able to predict more accurately the pressure data that changed gently with small changes. For some of the monitoring states such as motor speed and current that suddenly increase or decrease, both TCN and CNN-LSTM algorithms that do not use the attention mechanism are unable to accurately predict them, while the attention-driven CNN-LSTM prediction algorithm proposed in this paper not only learns the relationship between the parameters and their laws over time using the attention mechanism, but also acquires the sudden change trend from the connection between the parameters, and provides accurate predictions for the cable conveyor multi-monitoring states for accurate prediction. A similar situation occurs in the case of cable conveyor asynchrony and excessive side pressure faults. In the case of cable transport asynchrony faults, a conveyor’s speed decreases during operation, which corresponds to a change in the motor drive current. For the overpressure fault, the pressure perceived by a conveyor increases rapidly and then slowly with the cable transport process. Although the prediction results of the TCN and CNN-LSTM algorithms without using the attention mechanism also show an increasing trend, their errors are still larger than those of the proposed attention-driven CNN-LSTM algorithm.

The prediction results of these three prediction algorithms are compared, and the RSME and MAPE calculation results of the three algorithms for the prediction of the cable conveyor states are shown in Table 3. The prediction error of the attention-driven CNN-LSTM joint prediction algorithm proposed in this paper is generally smaller than the other algorithms, which proves the effectiveness of the attention mechanism and shows that the proposed attention-driven CNN-LSTM prediction algorithm has better results for the prediction of

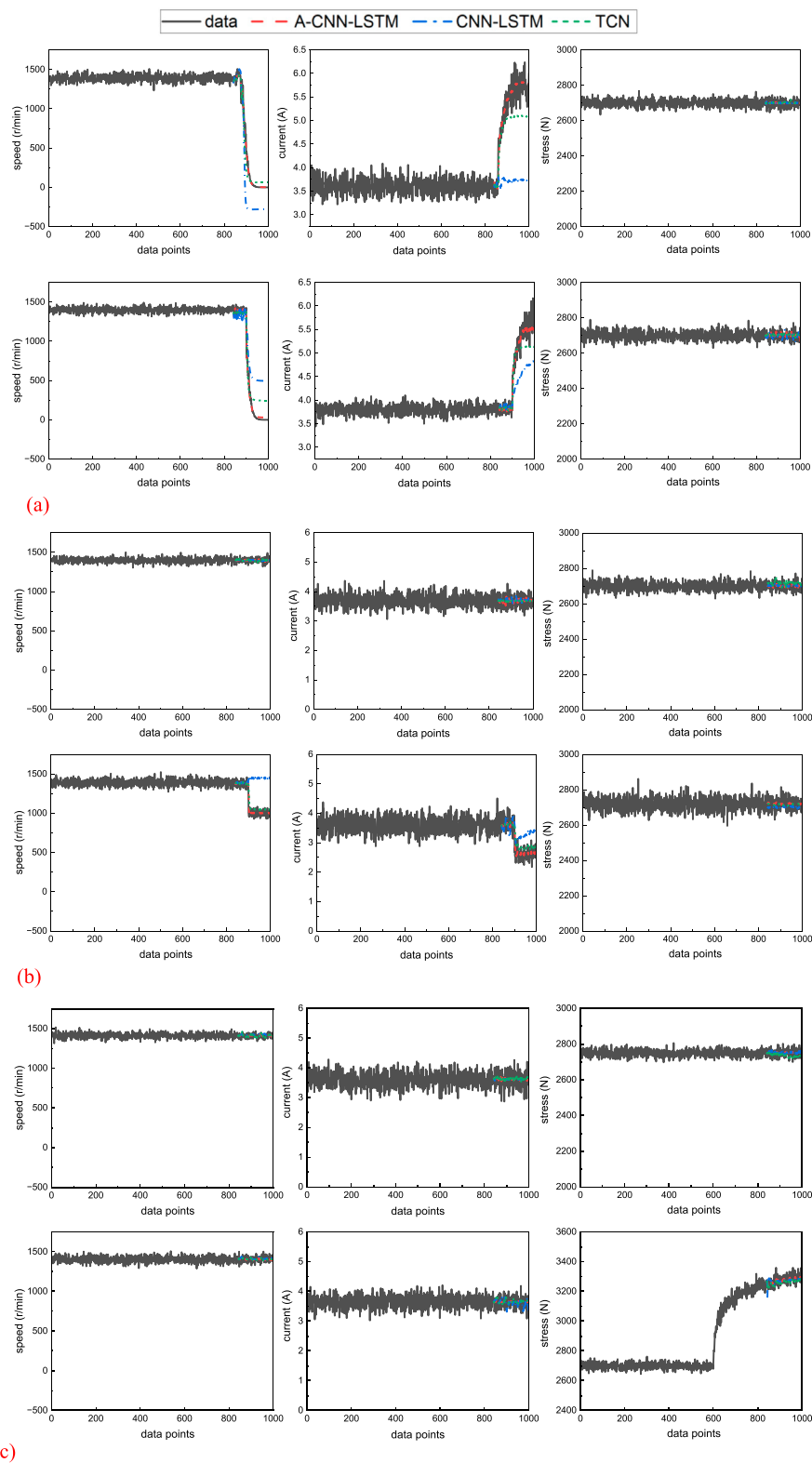


FIGURE 8 The prediction results of cable conveyor state with different algorithms (data for real measured results, A-CNN-LSTM for proposed algorithm, TCN for TCN, CNN-LSTM for the normal CNN-LSTM without attention). **(A)** prediction of cable jamming fault (speed 1, current 1, stress 1, speed 2, current 2, stress 2). **(B)** prediction of cable asynchrony fault (speed 1, current 1, stress 1, speed 2, current 2, stress 2). **(C)** prediction of excessive pressure fault (speed 1, current 1, stress 1, speed 2, current 2, stress 2).

TABLE 3 Prediction errors of different algorithms for each state of cable conveyor (A-C-L for A-CNN-LSTM, C-L for CNN-LSTM, T for TCN).

jamming fault	States	Speed 1			Current 1			Pressure 1		
	Algorithm	A-C-L	C-L	T	A-C-L	C-L	T	A-C-L	C-L	T
	RMSE	45.8	3320	107.7	0.211	1.71	0.25	18.9	21	20.1
	MAPE (%)	7.29	121	118	3.12	28.47	6.2	0.3	0.64	0.81
	States	Speed 2			Current 2			Pressure 2		
	Algorithm	A-C-L	C-L	T	A-C-L	C-L	T	A-C-L	C-L	T
	RMSE	46.7	354	169	0.156	0.651	0.341	24.4	29.6	24.6
	MAPE (%)	7.99	622	446	2.22	10.07	4.11	0.72	0.88	0.73
asynchrony fault	States	Speed 1			Current 1			Pressure 1		
	Algorithm	A-C-L	C-L	T	A-C-L	C-L	T	A-C-L	C-L	T
	RMSE	28.6	30.6	31.2	0.202	0.213	0.209	24.6	24.6	25.3
	MAPE (%)	1.61	1.69	1.99	4.43	4.78	4.51	0.71	0.73	1.05
	States	Speed 2			Current 2			Pressure 2		
	Algorithm	A-C-L	C-L	T	A-C-L	C-L	T	A-C-L	C-L	T
	RMSE	43.8	354	67.4	0.221	0.581	0.287	33.5	37.6	34.2
	MAPE (%)	2.75	28.8	3.45	5.6	18.8	7.05	1.02	1.11	1.09
Excessive pressure fault	States	Speed 1			Current 1			Pressure 1		
	Algorithm	A-C-L	C-L	T	A-C-L	C-L	T	A-C-L	C-L	T
	RMSE	26.5	31.1	27.7	0.263	0.264	0.272	20.8	21.9	24.6
	MAPE (%)	1.51	1.77	1.60	5.94	5.98	6.22	0.62	0.66	0.69
	States	Speed 2			Current 2			Pressure 2		
	Algorithm	A-C-L	C-L	T	A-C-L	C-L	T	A-C-L	C-L	T
	RMSE	36.8	37.9	38.2	0.176	0.208	0.186	26.8	31.4	44.8
	MAPE (%)	2.11	2.19	2.20	3.08	4.49	3.86	0.66	0.77	1.02

interconnected multi-monitoring states of the cable conveyor. To verify the feasibility of using this proposed algorithm for prediction and forewarning, the algorithm is placed in an embedded AI computing device—an Nvidia Jetson TX2 module. The time to make one prediction is 80 ms, which is much less than the prediction data length.

4.4 Effectiveness of multi-step prediction

To reserve more time for predicting the state of the cable conveyor and preventing equipment failures, the model needs to make multiple time-step predictions and even long-term predictions. Usually, there are three kinds of strategies for multiple time-step prediction: direct strategy, recursive strategy, and multi-output strategy as shown in Equations 15–17.

$$\begin{cases} \hat{y}_{t+1} = Model_1(x_t, x_{t-1}, \dots, x_{t-n}) \\ \hat{y}_{t+2} = Model_2(x_t, x_{t-1}, \dots, x_{t-n}) \end{cases} \quad (15)$$

$$\begin{cases} \hat{y}_{t+1} = Model(x_t, x_{t-1}, \dots, x_{t-n}) \\ \hat{y}_{t+2} = Model(\hat{y}_{t+1}, x_t, \dots, x_{t-n+1}) \end{cases} \quad (16)$$

$$(\hat{y}_{t+1}, \hat{y}_{t+2}, \dots, \hat{y}_{t+m}) = Model(x_t, x_{t-1}, \dots, x_{t-n}) \quad (17)$$

TABLE 4 Errors of the proposed model with different prediction time steps.

Multi-time step	5	10	30	50
RMSE	28.116	28.323	28.810	29.151
MAPE (%)	0.980	0.996	1.045	1.887

where x is the measured data and \hat{y} is the predicted data. The direct strategy uses a different model to predict the data at each time step, and it has the disadvantage of not being able to model the dependencies between predictions. The recursive strategy involves using the same single time-step model multiple times, with the predictions from the previous time-step used as inputs to make predictions for the next time-step length. However, it is difficult for recursive strategies to avoid the accumulation of prediction errors, and thus the prediction accuracy decreases rapidly as the prediction time increases. The multi-output strategy involves only one model that is capable of predicting multiple time steps from historical data in a one-time fashion. Multi-output models are more complex and therefore slower to train and require more training data to achieve satisfactory accuracy. In this paper, a multi-output strategy is used to perform multi-time-step long-term prediction, i.e., building a model with multiple inputs to multiple outputs.

Prediction is performed using the proposed attention-driven CNN-LSTM algorithm on data after 5, 10, 30, and 50 time steps, with the input time step set to 20 and the number of hidden layer nodes set to 64. The Adam algorithm is used to compute the back-propagation gradient and update the training parameters, with 1,000 training iterations, and an exponential decay is applied to the learning rate to prevent convergence oscillations, with the initial learning rate of 0.01. The initial learning rate is 0.01, the decay rate is 0.96, and the decay step size is 1000 steps, and the results are compared using MAPE and RMSE, as shown in Table 4. As the prediction time grows, the prediction error slightly improves, but none of them exceeds 10%, proving that the proposed attention-driven algorithm and multi-output strategy can also predict the multi-long-term results more accurately. It should be mentioned that the proposed model can be used for other cable laying tasks as long as the model is retrained using the new data which is related to the exact task.

5 Conclusion

To explore the connection between each monitoring quantity of cable conveyor and make a more accurate prediction, this paper proposes an attention-driven CNN-LSTM cable conveyor multi-monitoring state prediction algorithm based on the states of cable conveyor in three cable laying fault situations, and the results show that:

- (1) The CNN network can mine the connection between the states of the cable conveyor and intelligently assign weights to each monitoring state through the attention mechanism. The LSTM network can mine the law of the change of each state of the cable conveyor over time and intelligently assign time-step weights to each variable through the attention mechanism.
- (2) The proposed cable conveyor state prediction algorithm based on attention-driven CNN-LSTM predicts better than TCN, and CNN-LSTM algorithm without attention mechanism.
- (3) The proposed cable conveyor state prediction algorithm based on attention-driven CNN-LSTM can also predict long-term changes relatively accurately.

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.

References

- Arora, P., Jalali, S. M. J., Ahmadian, S., Panigrahi, B. K., Suganthan, P. N., and Khosravi, A. (2023). Probabilistic wind power forecasting using optimized deep auto-regressive recurrent neural networks. *IEEE Trans. Industrial Inf.* 19 (3), 2814–2825. doi:10.1109/tii.2022.3160696
- Choi, J.-K., Yokobiki, T., and Kawaguchi, K. (2018). ROV-based automated cable-laying system: application to DONET2 installation. *IEEE J. Ocean. Eng.* 43 (3), 665–676. doi:10.1109/joe.2017.2735598
- Diban, B., Mazzanti, G., and Seri, P. (2022). Life-based geometric design of HVDC cables—Part I: parametric analysis. *IEEE Trans. Dielectr. Electr. Insulation* 29 (3), 973–980. doi:10.1109/tdei.2022.3168369
- Fan, C., Peng, Y., Shen, Y., Guo, Y., Zhao, S., Zhou, J., et al. (2024). Variable scale multilayer perceptron for helicopter transmission system vibration data abnormality

Author contributions

LJ: Writing—original draft, Writing—review and editing. YL: Writing—review and editing. DY: Writing—review and editing. JH: Writing—review and editing. JuX: Writing—review and editing. JiX: Writing—review and editing. YC: Writing—original draft, Writing—review and editing.

Funding

The author(s) declare that financial support was received for the research, authorship, and/or publication of this article. This research was funded by the Science and Technology Project of State Grid Jiangsu Electric Power Co., Ltd., grant number J2023041. The funder was not involved in the study design, collection, analysis, interpretation of data, the writing of this article, or the decision to submit it for publication.

Conflict of interest

Authors LJ, YL, DY, JH, JuX, and JiX were employed by State Grid Wuxi Power Supply Company.

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

Generative AI statement

The author(s) declare that no Generative AI was used in the creation of this manuscript.

Publisher's note

All claims expressed in this article are solely those of the authors and do not necessarily represent those of their affiliated organizations, or those of the publisher, the editors and the reviewers. Any product that may be evaluated in this article, or claim that may be made by its manufacturer, is not guaranteed or endorsed by the publisher.

beyond efficient recovery. *Eng. Appl. Artif. Intell.* 133, 108184. doi:10.1016/j.engappai.2024.108184

Ge, L., Xian, Y., and Wang, Z. (2021). Short-term load forecasting of regional distribution network based on generalized regression neural network optimized by grey wolf optimization algorithm. *CSEE J. Power Energy Syst.* 7 (5), 1093–1101. doi:10.17775/CSEEJPES.2020.00390

Ghorbani, H., Jeroense, M., Olsson, C.-O., and Saltzer, M. (2014). HVDC cable systems—highlighting extruded technology. *IEEE Trans. Power Deliv.* 29 (1), 414–421. doi:10.1109/tpwrd.2013.2278717

Hewage, P., Behera, A., Trovati, M., Pereira, E., Ghahremani, M., Palmieri, F., et al. (2020). Temporal convolutional neural (TCN) network for an effective weather forecasting using time-series data from the local weather station. *Soft Comput.* 24, 16453–16482. doi:10.1007/s00500-020-04954-0

- Hochreiter, S., and Schmidhuber, J. (1997). Long short-term memory. *Neural Comput.* 9 (8), 1735–1780. doi:10.1162/neco.1997.9.8.1735
- Hou, M., and Han, X. (2010). Constructive approximation to multivariate function by decay RBF neural network. *IEEE Trans. Neural Netw.* 21 (9), 1517–1523. doi:10.1109/tnn.2010.2055888
- Li, J., and Ma, L. (2022). Short-term wind power prediction based on soft margin multiple kernel learning method. *Chin. J. Electr. Eng.* 8 (1), 70–80. doi:10.23919/cjee.2022.000007
- Li, S., Peng, Y., and Bin, G. (2023). Prediction of wind turbine blades icing based on CJBm with imbalanced data. *IEEE Sensors J.* 23 (17), 19726–19736. doi:10.1109/jsen.2023.3296086
- Mirowski, P., and LeCun, Y. (2012). Statistical machine learning and dissolved gas analysis: a review. *IEEE Trans. Power Deliv.* 27 (4), 1791–1799. doi:10.1109/tpwrd.2012.2197868
- Montanari, G. C., Seri, P., Lei, X., Ye, H., Zhuang, Q., Morshuis, P., et al. (2018). Next generation polymeric high voltage direct current cables—a quantum leap needed? *IEEE Electr. Insul. Mag.* 34 (2), 24–31. doi:10.1109/mei.2018.8300441
- Sun, J., Hu, C., Liu, L., Zhao, B., Liu, J., and Shi, J. (2022). Two-stage correction strategy-based real-time dispatch for economic operation of microgrids. *Chin. J. Electr. Eng.* 8 (2), 42–51. doi:10.23919/cjee.2022.000013
- Wang, Y., Wang, Y., Wu, J., and Yin, Y. (2018). Research progress on space charge measurement and space charge characteristics of nanodielectrics. *IET Nanodielectrics* 1 (3), 114–121. doi:10.1049/iet-nde.2018.0009
- Wang, Y., Xia, Q., and Kang, C. (2011). Secondary forecasting based on deviation analysis for short-term load forecasting. *IEEE Trans. Power Syst.* 26 (2), 500–507. doi:10.1109/tpwrs.2010.2052638
- Yang, Y., Zhang, H., Peng, S., and Su, S. (2023). Wind power probability density prediction based on quantile regression model of dilated causal convolutional neural network. *Chin. J. Electr. Eng.* 9 (1), 120–128. doi:10.23919/cjee.2023.000001
- Zhao, W., Xia, X., Li, M., Huang, H., Chen, S., Wang, R., et al. (2019). Online monitoring method based on locus-analysis for high-voltage cable faults. *Chin. J. Electr. Eng.* 5 (3), 42–48. doi:10.23919/cjee.2019.000019
- Zhu, X., Wu, J., Wang, Y., and Yin, Y. (2019). Characteristics of electrical tree defect during the growth period in high-voltage DC cable under stepped DC voltage. *IET Generation, Transm. and Distribution* 13 (15), 3195–3201. doi:10.1049/iet-gtd.2019.0078