CNN-LSTM-Based QAR Data Feature Extraction and Prediction (Postprint)

Authors: Zhang Peng, Yang Tao, Yanan Liu, Fan Zhiyong, 段照斌

Date: 2018-06-19T00:00:00+00:00

Abstract

To address the challenge that traditional data-driven fault diagnosis methods face in extracting effective features from QAR data, this paper proposes a CNN-LSTM dual-channel fusion model integrating Convolutional Neural Network (CNN) and Long Short-Term Memory (LSTM). CNN and LSTM serve as two separate channels, fused through an attention mechanism to enable simultaneous representation of data features in both spatial and temporal dimensions, with effectiveness verified through time series prediction. Experimental results demonstrate that, compared with single CNN or LSTM models, the dual-channel fusion model can extract data features more effectively, reducing the prediction errors of both single-step and multi-step forecasting by an average of 35.3%. This provides a new research approach for fault diagnosis based on QAR data.

Full Text

Preamble

Feature Extraction and Prediction of QAR Data Based on CNN-LSTM

Zhang Peng[1], Yang Tao[2], Liu Yanan[2], Fan Zhiyong[1], Duan Zhaobin[1]

[1]Engineering Training Center, Civil Aviation University of China, Tianjin 300300, China
[2]College of Electronic Information & Automation, Civil Aviation University of China, Tianjin 300300, China

Abstract: Traditional data-driven fault diagnosis methods struggle to extract effective features from Quick Access Recorder (QAR) data. To address this challenge, this paper proposes a dual-channel fusion model, CNN-LSTM, that integrates Convolutional Neural Networks (CNN) and Long Short-Term Memory (LSTM) networks. CNN and LSTM serve as two separate channels, fused through an attention mechanism, enabling the model to simultaneously capture features in both spatial and temporal dimensions. The effectiveness of the fusion model' s feature extraction is validated through time series prediction. Experimental results demonstrate that, compared with standalone CNN or LSTM models, the dual-channel fusion model extracts data features more effectively, reducing single-step and multi-step prediction errors by an average of 35.3%. This approach provides a novel research direction for fault diagnosis based on QAR data.

Keywords: deep learning; CNN; LSTM; feature extraction; time series prediction

0 Introduction

Quick Access Recorder (QAR) data continuously and comprehensively records various parameters during flight, providing critical information that reflects the actual operational status or failure 征兆 signals of aircraft systems. This data plays a vital role in ensuring flight safety. However, since QAR data primarily consists of direct measurements from aircraft sensors, the relationships between parameters are not immediately apparent, making it difficult to evaluate flight quality directly. Extracting feature vectors that reflect system operational status from QAR data to discover potential correlations between aircraft systems and applying them to fault diagnosis represents a significant challenge in this field.

Consider the Power Control Unit (PCU) in aircraft flight control systems as an example. Fault manifestations include PCU non-tracking, PCU locked valve misalignment, PCU bypass valve misalignment, and PCU actuation reliability test failures. The interconnected systems involved include the power system, hydraulic system, and sensor system. During actual troubleshooting, maintenance personnel must spend considerable time following complex signal analysis logic to eliminate each potential fault cause. This requires not only mastery of the PCU system' s working principles but also understanding of interconnected systems. If we abstract PCU fault diagnosis as a classification problem, extracting features from relevant QAR data that monitors the PCU, establishing a mapping between features and fault categories, and performing intelligent fault diagnosis could significantly improve troubleshooting efficiency and ensure normal flight operations.

Traditional data-driven fault diagnosis methods rely on signal processing and statistical theories to manually extract features from raw data. These approaches suffer from several limitations: they require extensive engineering practice and signal processing expertise, heavily depend on personnel expertise, and struggle to cope with the high-dimensional, massive "big data" characteristics of QAR data. Deep learning theory, proposed by Hinton et al., has emerged as a research hotspot in machine learning, focusing on automatically extracting multi-layer feature representations from data. Recent studies have demonstrated deep learning’ s unique advantages and potential in feature extraction and pattern recognition for complex industrial systems. For instance, researchers have applied Deep Belief Networks (DBN) to aircraft fault diagnosis, LSTM networks to lithium battery Remaining Useful Life (RUL) prediction, Autoencoders to grinding system fault diagnosis with 92.4% accuracy, and CNNs to gearbox fault identification. These studies show that deep learning-based fault diagnosis algorithms outperform traditional methods and can eliminate the complex manual feature extraction process.

Building upon deep learning theory, this paper proposes a dual-channel fusion deep learning model, CNN-LSTM, for QAR data feature extraction. The model’ s feature completeness is validated through QAR data prediction. Specifically, we predict aircraft pitch angle variations over 500 points without using historical values of the prediction parameter itself. Results demonstrate that the fusion model achieves superior feature representation performance with higher accuracy in both single-step and multi-step predictions compared to standalone CNN and LSTM models.

1 Deep Learning Theory

Deep learning originated from artificial neural network research, with its distinguishing characteristic being multi-layer perceptrons with multiple hidden layers. Compared with shallow learning algorithms, deep learning offers better capability to approximate complex functions. Through its multi-hidden-layer architecture, it enables hierarchical data transformation, ensuring the most effective information extraction and feature representation. As a data-driven approach, deep learning eliminates the need for accurate physical system modeling —only historical system operation data is required to obtain optimal feature representations for fault diagnosis, classification, and prediction tasks. This paper focuses on two algorithms: CNN and LSTM.

1.1 Convolutional Neural Network (CNN)

CNN, proposed by LeCun, is a feedforward neural network. The typical LeNet-5 architecture consists of an input layer, convolutional layers (C), pooling layers (S), fully connected layers (F), and an output layer. The essence of CNN lies in constructing multiple filters that extract data features through successive convolution and pooling operations to reveal hidden topological characteristics. As network depth increases, extracted features become increasingly abstract. These abstract features are ultimately merged through fully connected layers and processed by softmax or sigmoid activation functions for classification and regression tasks. A key advantage of CNN is its ability to extract local features from input data and hierarchically combine them into high-level features, effectively enabling fault diagnosis and recognition.

1.2 Long Short-Term Memory Network (LSTM)

LSTM is a recurrent neural network particularly suited for time series data. Compared with traditional Recurrent Neural Networks (RNN), LSTM introduces memory cells that solve the long-term dependency problem plaguing RNNs in practical applications. LSTM consists of an input layer, hidden layer, and output layer, with its structure shown in [Figure 2: see original paper]. In the diagram, x_t represents input units, h_t represents state output units, M_t represents memory cells, while i_t, o_t, and f_t denote input gates, output gates, and forget gates, respectively.

The read, write, and forget operations of M_t are controlled by three gate units. Given an input time series $\{x_1, x_2, ..., x_t\}$ where t is the current time step, the states of each unit can be expressed through the following formulas:

Input unit:

$$g_t = \tanh(W_{xg}x_t + W_{hg}h_{t-1} + b_g)$$

Gate units:

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + b_i)$$
$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + b_f)$$
$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + b_o)$$

Memory cell:

$$M_t = f_t \odot M_{t-1} + i_t \odot g_t$$

State output unit:

$$h_t = o_t \odot \tanh(M_t)$$

where σ represents the sigmoid activation function.

2 Dual-Channel Fusion Model

Both CNN and LSTM are mainstream deep learning algorithms with complementary strengths: CNN excels at spatial expansion, extracting local data features and combining them into abstract high-level representations, while LSTM specializes in temporal expansion with long-term memory capabilities, making it ideal for time series data. QAR data feature extraction requires consideration of both spatial relationships between different parameters and temporal variations in the data. Therefore, this paper proposes a dual-channel fusion model (CNN-LSTM) that combines CNN and LSTM through an Attention mechanism, enabling the model to express features in both temporal and spatial dimensions. The model structure is illustrated in [Figure 3: see original paper].

In the model, p_i represents the i-th parameter in the input sample, T represents the sample time length, C_i^r represents the i-th feature extracted by the CNN network, and h_i represents the output of the i-th hidden layer of LSTM.

Samples are fed into the network through left and right channels. The left channel is the CNN network, whose final layer outputs an n-dimensional feature vector through n neurons:

$$\mathbf{C}^r = [c_1^r, c_2^r, ..., c_n^r]$$

The right channel is the LSTM network with sequence length T and hidden layer output dimension m, producing an $m \times n$-dimensional feature vector:

$$\mathbf{H} = [h_1, h_2, ..., h_n]$$

After obtaining feature representations from both channels, an Attention mechanism generates the fused feature map. The Attention layer structure is shown in [Figure 4: see original paper]. The process can be described by the following formulas:

$$\varphi(c_i^r, h_i) = \tanh(c_i^{rT} W_a h_i + b_a)$$

$$\alpha_i = \frac{\exp(\varphi(c_i^r, h_i))}{\sum_{j=1}^{n} \exp(\varphi(c_j^r, h_j))}$$

$$f_{map} = \sum_{i=1}^{n} \alpha_i h_i$$

where W_a is an $m \times n$-dimensional weight matrix, b_a is a bias term, both learned during training, and c_i^{rT} is the transpose of c_i^r.

Equation (9) fuses CNN' s feature vector with LSTM' s feature vector into a set of "weights," which are normalized to the range [0,1] through the Softmax activation function in Equation (10). Finally, Equation (11) multiplies LSTM' s hidden layer output vectors h_i at different time points with their corresponding "weights" α_i and sums them to produce the final feature representation f_{map}. This approach allows CNN' s ability to combine abstract local features to modulate LSTM' s temporal feature representation through the attention mechanism, strengthening features in certain regions while weakening others, thereby endowing the model with both temporal and spatial feature expression capabilities.

3 Experiments and Results

3.1 Dataset Construction

This study focuses on the pitch channel of the flight control system, extracting features from relevant parameters to predict pitch angle variations. Based on the flight control system working principles [15], 22 relevant parameters were selected from QAR data with a sampling interval of 1 second. The parameter descriptions are listed in .

The pitch angle (PITCH ATTITUDE) serves as the prediction parameter, while the remaining parameters are used as training features. The sample construction method is illustrated in [Figure 5: see original paper]. Normalized data are arranged chronologically to form a two-dimensional data matrix, with training sets constructed using a sliding window approach. The prediction labels are the values of the prediction parameter at future time steps $y_1, y_2, ..., y_N$, as shown in the figure. Considering this experiment as a time series prediction task, to ensure temporal continuity of the prediction parameter, the step size is set to $s = 1$. To facilitate convolution operations in CNN, the sliding window size is selected to match the number of training parameters, making each sample a 21×21 square matrix. To evaluate both single-step and multi-step prediction performance, N values of 1, 5, and 10 are selected, yielding datasets S_1, S_2, and S_3 through the described construction method.

In the CNN-LSTM model, data in the left (CNN) channel first passes through a 1D convolutional layer (Conv1D) with 32 kernels of size 3×1, using ReLU activation, transforming the tensor shape to $(?, 19, 32)$. This is followed by a max pooling layer (Maxpooling1D) with window size 2, changing the tensor dimension to $(?, 9, 32)$. Another convolutional layer with 16 kernels of size 3×1 further transforms the tensor to $(?, 7, 16)$. A subsequent max pooling layer with window size 2 reduces the dimension to $(?, 3, 16)$. Finally, a Flatten layer "flattens" the data, which then passes through a fully connected layer (Dense) with output dimension 21, resulting in a tensor dimension of $(?, 21)$.

The right (LSTM) channel processes data through an LSTM unit with output dimension 50, taking its hidden layer output to obtain a tensor of dimension $(?, 21, 50)$. The two channel tensors are fused by the Attention layer implemented according to Equation (9), producing a tensor shape of $(?, 1, 50)$ representing the fused feature expression. This passes through a fully connected layer with output dimension 32, followed by a Dropout layer to prevent overfitting, and finally a fully connected output layer with dimension 1 using Sigmoid activation.

The LSTM model is similar to the right channel of CNN-LSTM, with input data passing through a 50-dimensional LSTM unit but taking only the final hidden layer output, resulting in a tensor dimension of $(?, 1, 50)$. Subsequent processing mirrors the post-Attention layers of the CNN-LSTM model. The CNN model follows a structure similar to the left channel of CNN-LSTM.

To evaluate model performance, we adopt Mean Absolute Error (MAE), Mean Absolute Percentage Error (MAPE), Root Mean Square Error (RMSE), and Root Mean Square Logarithmic Error (RMSLE) as evaluation metrics [16], calculated as:

$$MAE = \frac{1}{n} \sum_{t=1}^{n} |F_t - A_t|$$

$$MAPE = \frac{1}{n}\sum_{t=1}^{n}\left|\frac{F_t - A_t}{A_t}\right| \times 100\%$$

$$RMSE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(F_t - A_t)^2}$$

$$RMSLE = \sqrt{\frac{1}{n}\sum_{t=1}^{n}(\log(1+F_t) - \log(1+A_t))^2}$$

where F_t represents predicted values and A_t represents actual values. Smaller values for all four metrics indicate predictions closer to actual values, demonstrating better model performance and stronger feature expression capability. The training, validation, and test set splits for datasets S_1, S_2, and S_3 are shown in .

3.2 Experiments and Results

To validate the effectiveness of the proposed dual-channel fusion model, we implemented it on the Keras deep learning platform based on TensorFlow and conducted comparative experiments with standalone LSTM and CNN models on datasets S_1, S_2, and S_3. The model structures and parameters are illustrated in [Figure 6: see original paper].

The prediction results of the three models on the three test sets are shown in Figure 7: see original paper-(c), with error metrics calculated in . Comparing the results reveals that on dataset S_1 (single-step prediction), all three models accurately predict pitch angle values due to the short prediction interval, though the CNN model begins deviating after the 350th point while maintaining the general trend. On dataset S_2 (5-step prediction), all models deviate from actual values to varying degrees, with CNN-LSTM showing the smallest deviation. On dataset S_3 (10-step prediction), all models exhibit further deviation and overall lag effects. After the 450th point, both CNN and LSTM show oscillations deviating from the true trend, while only CNN-LSTM maintains consistency with the actual curve.

The error metrics in demonstrate that CNN-LSTM significantly outperforms CNN and LSTM on datasets S_2 and S_3 across all four error indicators. On dataset S_1, CNN-LSTM performance is comparable to LSTM, with equivalent MAE and slightly higher MAPE than LSTM. LSTM consistently outperforms CNN across all three datasets, indicating its superior suitability for time series data. Overall, CNN-LSTM reduces errors by an average of 69.8% compared to CNN and 38.2% compared to LSTM for single-step prediction, 51.0% compared to CNN and 33.1% compared to LSTM for five-step prediction, and 19.6% compared to CNN and 1.5% compared to LSTM for ten-step prediction. On average, the proposed CNN-LSTM model reduces single-step and multi-step prediction errors by 35.3% compared to standalone CNN and LSTM models.

These results demonstrate that the fusion approach effectively combines the respective strengths of CNN and LSTM to extract more comprehensive features from QAR data.

4 Conclusion

Effective analysis and utilization of QAR data are crucial for improving flight operational efficiency and ensuring flight safety. Constrained by the high-dimensional, massive sample characteristics of QAR data, traditional feature extraction methods prove inadequate. Current QAR data analysis primarily focuses on statistical approaches. This paper proposes a deep learning-based dual-channel fusion model, CNN-LSTM, which combines CNN and LSTM through an Attention mechanism to more comprehensively extract QAR data features, providing a more effective utilization method for QAR data.

Pitch angle prediction during flight demonstrates that the fused model combining CNN and LSTM achieves more effective feature representation and higher prediction accuracy than single models for both single-step and multi-step predictions. Additionally, this paper introduces a dataset construction methodology suitable for industrial big data, adapting window size, step length, and time intervals to different practical problems. This research provides a novel approach for subsequent fault diagnosis studies based on QAR data.

References

[1] Geng Hong, Li Pingping. Application of QAR data processing in aerodynamic moment parameter identification [J]. Avionics Technology, 2012, 43 (03): 25-29.

[2] Ju Jianbo, Hu Shenglin, Zhu Chao, et al. A deep learning based method for equipment fault diagnosis [J]. Electronics Optics & Control, 2018, 25 (02): 103-106.

[3] Hinton G E, Osindero S, Teh Y W. A fast learning algorithm for deep belief nets [J]. Neural Computation, 2006, 18 (7): 1527-1554.

[4] Zhang Junyang, Wang Huili, Guo Yang, et al. Review of Deep Learning [J/OL], Application Research of Computers, 2018, [2018-03-21]. http://kns.cnki.net/kcms/detail/51.1196.tp.20170818.1709.084.html.

[5] Reng Hao, Qu Jianfeng, Chai Yi, et al. Deep learning for fault diagnosis: The state of the art and challenge [J]. Control and Decision, 2017, 32 (8): 1345-1358.

[6] Jiang Hongkai, Shao Haidong, Chen Xinxia, et al. Aircraft Fault Diagnosis Based on Deep Belief Network [C]// Proc of International Conference on Sensing, Diagnostics, Prognostics, and Control. 2017: 123-127.

[7] Zhang Yongzhi, Xiong Rui, He Hongwen, et al. A LSTM-RNN method for the lithium-ion battery remaining useful life prediction [C]// Proc of Prognostics and System Health Management Conference. 2017: 1-4.

[8] Hochreiter S, Schmidhuber J. Long Short-term memory [J]. Neural Computation, 1997, 9 (8): 1735-1780.

[9] Qu Xingyu, Zeng Peng, Fu Dongdong, et al. Autoencoder-based fault diagnosis for grinding system [C]// Proc of Chinese Control and Decision Conference. 2017: 3867-3872.

[10] Chen Zhiqiang, Li Chuan, Sanchez R V. Gearbox fault identification and classification with convolutional neural networks [J]. Shock & Vibration, 2015, 2015 (2): 1-10.

[11] Lecun Y, Bottou L, Bengio Y, et al. Gradient-based learning applied to document recognition [J]. Proceedings of the IEEE, 1998, 86 (11): 2278-2324.

[12] Jia Feng, Lei Yaguo, Lin Jing, et al. Deep neural networks: a promising tool for fault characteristic mining and intelligent diagnosis of rotating machinery with massive data [J]. Mechanical Systems & Signal Processing, 2016, 72-73: 303-315.

[13] Zhang Qingchen, Yang L T, Chen Zhikui. Deep computation model for unsupervised feature learning on big data [J]. IEEE Trans on Services Computing, 2016, 9 (1): 161-171.

[14] Bahdanau D, Cho K, Bengio Y. Neural machine translation by jointly learning to align and translate [J]. Computer Science, 2014.

[15] Xu Jun. Flight Control System: Design, Prototype and Semi Physical Simulation Experiment [M]. Beijing: Beijing Institute of Technology Press, 2015.

[16] Zhang Chengzhao. A Deep Learning Model for Financial Market Forecasting: FEPA Model [D]. Chengdu: University of Electronic Science and Technology of China, 2016.

Note: Figure translations are in progress. See original paper for figures.