Proceedings of the 4th International Conference on Software Engineering and Machine Learning
DOI: 10.54254/2755-2721/2026.33932

Multi-class Prediction of Embedded Software Faults Based on Machine Learning Algorithms

Yibing Chen

School of Computer and Information Engineering, Henan University of Economics and Law, Zhengzhou, China
superice666@126.com

Abstract. For the embedded software in safety-critical fields such as aerospace and automotive electronics, which has the characteristics of occasional occurrence, concealment, and strong temporal correlation, while traditional rules and static analysis methods have lagging early warning and high false alarm rates, as well as existing models relying on manual parameter tuning and insufficient generalization, this paper proposes the RBMO-BiLSTM-Attention fusion algorithm. This method combines the improved Magpie Blue Jay optimization algorithm with bidirectional long short-term memory network and attention mechanism. It uses RBMO to achieve automatic optimization of model parameters, utilizes BiLSTM to capture temporal dependencies in both directions, and employs the attention mechanism to focus on key fault features, thereby enhancing the ability to identify complex faults. The experiments are conducted based on the embedded software fault multi-classification dataset. The results show that the proposed algorithm performs the best in the fault classification task, can accurately identify fault types such as deadlock, memory overflow, and task timeout, and provides effective technical support for embedded system fault warning and reliability improvement.

Keywords: Embedded software faults, RBMO, BiLSTM, fault prediction

1. Introduction

Embedded systems have become the core support of key safety fields such as aerospace, automotive electronics and industrial control, and their software reliability directly determines the operation safety of systems [1]. With the increasingly complex functions of embedded software, dynamic faults such as deadlock, memory overflow and task timeout occur frequently. Such faults have the characteristics of occasionality, concealment and strong timing correlation. Traditional methods based on rules and static analysis are difficult to accurately capture the fault evolution law, resulting in lagging fault warning and high false alarm rate, which cannot meet the application requirements of real-time and high reliability [2]. Existing fault prediction mostly relies on manual extraction of features, which is difficult to adapt to the nonlinear and dynamic characteristics of embedded multi-source time series data, which restricts the improvement of fault classification accuracy and system stability [3].

© 2026 The Authors. This is an open access article distributed under the terms of the Creative Commons Attribution License 4.0 (https://creativecommons.org/licenses/by/4.0/).

DOI: 10.54254/2755-2721/2026.33932

Machine learning provides a data-driven solution for embedded software failure prediction, enabling fault intelligent identification by mining potential patterns in operational data [4]. Traditional machine learning algorithms can extract fault-related information from CPU usage, memory occupancy, task scheduling and other features to achieve basic fault classification, but there are limitations in dealing with long-time-series dependence and multi-feature coupling problems [5]. Deep learning models such as BiLSTM can effectively capture the bidirectional dependencies of time series data, and the attention mechanism can dynamically focus on key fault features. The combination of the two significantly improves the recognition ability of complex fault modes. However, the existing model hyperparameters mostly depend on empirical setting, which is easy to fall into local optimum, resulting in insufficient generalization ability in the embedded fault multi-classification scenario, and it is difficult to ensure the prediction accuracy and real-time performance at the same time.

Therefore, this paper proposes the RBMO-BiLSTM-Attention algorithm, which combines the improved RBMO optimization algorithm with the BiLSTM-Attention model to solve the core problem of multi-classification fault prediction in embedded software. The algorithm uses the global optimization ability of RBMO algorithm to automatically optimize the network structure and attention mechanism parameters of BiLSTM, avoiding the subjectivity and inefficiency of manual parameter adjustment. The BiLSTM module fully mines the timing dependence of embedded running data, and the attention mechanism dynamically assigns feature weights, and strengthens the expression of key fault features such as deadlock, overflow and timeout. The model takes into account both time series feature capture and parameter optimization, can accurately identify different fault types, and provides efficient technical support for fault warning and reliability improvement of embedded systems.

2. Data sources

The multi-classification data set of embedded software faults adopted in this paper totals 1728 sample data. Relying on the actual operation mechanism of embedded real-time system and the range of engineering reasonable indicators, the simulation acquisition and construction are completed. The core characteristic indicators are screened around the normal operation process of embedded devices, and the key timing operation characteristics such as CPU usage rate, memory occupancy ratio, task running number, stack resources, consumption interrupt frequency, task conflict number, resource lock waiting time, data transmission rate, watchdog working state and chip working temperature are completely retained. Four types of sample labels are distinguished: system normal operation state, deadlock fault, memory overflow fault and task timeout fault.

The continuity variables were first analyzed with a violin plot, as shown in Figure 1.

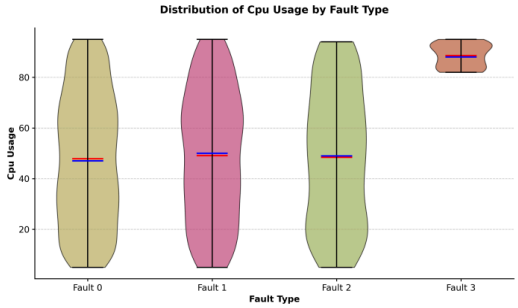


Figure 1. The violin plots

DOI: 10.54254/2755-2721/2026.33932

The correlation analysis was performed on each variable and the correlation heat map was plotted, as shown in Figure 2.

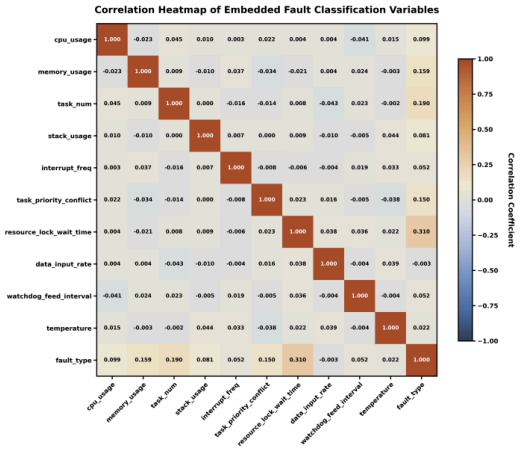


Figure 2. The correlation heat map

3. Method

3.1. RBMO

RBMO algorithm is a meta-heuristic optimization algorithm based on the predation behavior of red-billed blue magpies. The core is to simulate the behavior of birds cooperatively searching, attacking prey and storing food to find the optimal solution. When the algorithm is initialized, multiple individuals are randomly generated in the solution space, and each individual represents a potential solution. The search stage is divided into group search and cluster search, the group is composed of 2 to 5 individuals, and the position is updated by calculating the difference between the average position of the group and the random individual, so as to strengthen the local exploration ability; The cluster is composed of more than 10 individuals, and the global average location information is used to balance the global exploration and local development. In the attack stage, the individual moves towards the current optimal solution, and combines the difference between the average position of the group or cluster and its own position, and adjusts the moving step size through adaptive coefficients. The algorithm focuses on global exploration in the early stage and local fine development in the later stage [6]. The food storage behavior retains the historical optimal solution, avoids the loss of global optimal information, and improves the stability and convergence accuracy of the algorithm. The network structure of the RBMO is shown in Figure 3.

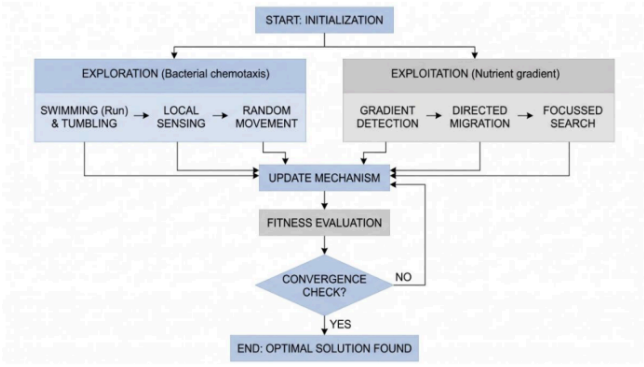


Figure 3. The network structure of the RBMO

DOI: 10.54254/2755-2721/2026.33932

3.2. BiLSTM

BiLSTM is a sequence modeling model built on the basis of LSTM. Its core is to capture the forward and reverse dependencies of sequences simultaneously, and comprehensively capture the context information. Its structure consists of two independent LSTM layers in parallel, and the forward LSTM layer processes inputs from the beginning to the end of the sequence in chronological order to extract historical information; The reverse LSTM layer processes inputs in reverse chronological order from the end of the sequence to the start, capturing future context information. Each LSTM unit controls the flow of information through a forget gate, an input gate, and an output gate. The forget gate determines the retention degree of information in the cell state at the previous moment, the input gate controls the write ratio of the current input information, and the output gate screens the cell state output to the current hidden state, effectively solving the long-term dependency problem of traditional RNN [7]. At each time step, the BiLSTM stitches the hidden states of the forward and reverse LSTM to form a final representation of the fused bidirectional information. The network structure of BiLSTM is shown in Figure 4.

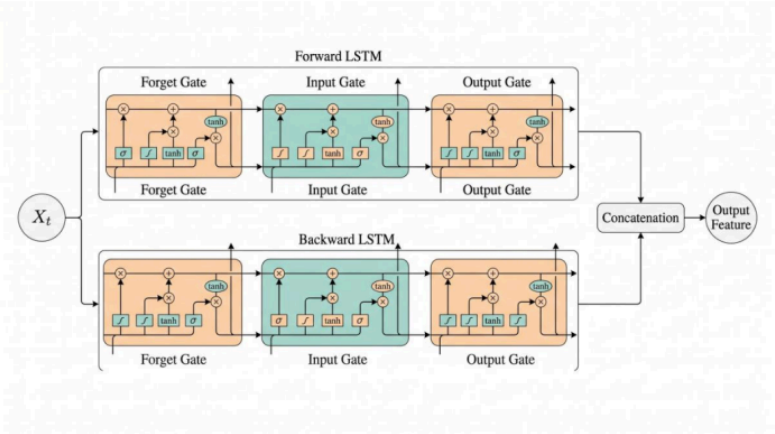


Figure 4. The network structure of BiLSTM

3.3. Attention

Attention algorithm is the Attention mechanism. Its core is to simulate the cognitive process of human selective Attention, assign differentiated weights to different position information of the input sequence, and let the model focus on key information. The calculation process is divided into three steps. First, the input data is converted into query vector, key vector and value vector. The query vector represents the current task requirements, the key vector is the identifier of the input feature, and the value vector corresponds to the specific content of the input [8]. Then the similarity between the query and the key is calculated, and the attention score is obtained by dot product, cosine similarity and other methods, and then the score is normalized by Softmax function to obtain the attention weights with a sum of 1, so as to ensure that the weights are explainable and stable. Finally, the value vectors are weighted and summed by the attention weights, and the feature vectors which fuse key information are output, which realizes the strengthening of important information and the weakening of irrelevant information. The network structure of Attention is shown in Figure 5.

DOI: 10.54254/2755-2721/2026.33932

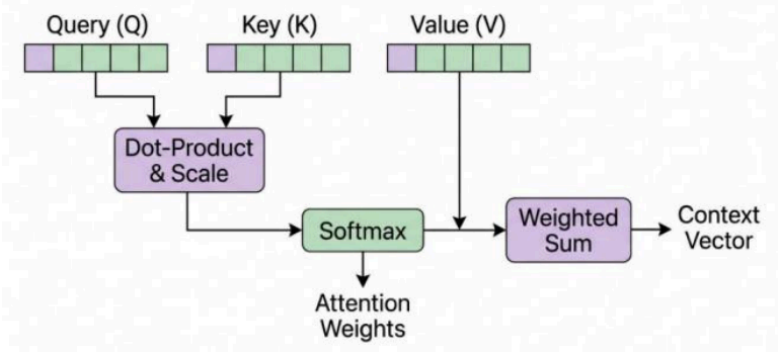


Figure 5. The network structure of Attention

3.4. RBMO-BiLSTM-Attention

RBMO-BiLSTM-Attention algorithm is a combination model that integrates Red-billed Blue Magpie optimizer, bidirectional long-short-term memory network and Attention mechanism, and realizes efficient modeling and prediction of sequence data through three-layer collaboration. First, the RBMO algorithm is responsible for optimizing the key hyperparameters of the BiLSTM-Attention model, including the number of hidden layer nodes, initial learning rate, regularization coefficient, etc. By simulating the group behavior of the red-billed blue magpie, it efficiently searches for the optimal combination in the parameter space to avoid the model falling into the local optimum and improve the overall performance [9]. Secondly, the BiLSTM layer, as the core feature extractor, processes the input sequence bidirectionally while capturing historical and future contextual dependencies, retains long-term effective information through a gating mechanism, and outputs a hidden state sequence containing rich timing features. Subsequently, the Attention layer weighted the output of BiLSTM, calculated the Attention weight of the hidden state at each time step, automatically focused the most critical timing information for the current task, filtered redundant noise, and enhanced the effective feature expression [10]. Finally, the weighted fused features are input into the fully connected layer to complete the final prediction of classification or regression task. The network structure of RBMO-BiLSTM-Attention is shown in Figure 6.

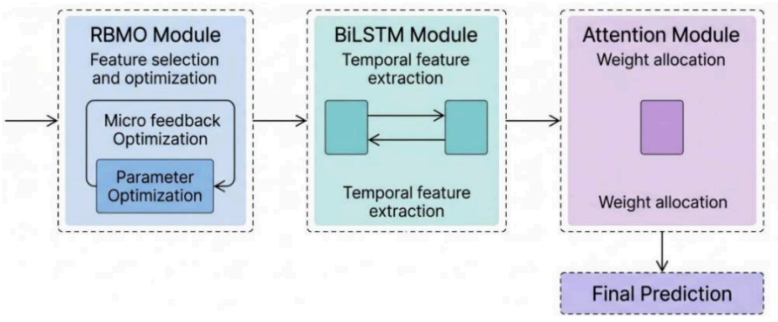


Figure 6. The network structure of RBMO-BiLSTM-Attention

4. Result

The parameter setting of this project includes two parts: optimization algorithm and neural network. The optimization algorithm Red-billed Blue Magpie optimizer sets 12 search agents and 6 maximum iteration times. The lower limits of the three hyperparameters to be optimized are 1 times 10 to the negative 5th power, 0.0001 and 10, and the upper limits are 0.01, 0.002 and 50, respectively. The

DOI: 10.54254/2755-2721/2026.33932

maximum training times of the neural network part are set to 500 times, the gradient threshold is set to 1, the learning rate is adjusted by a factor of 0.2 after 400 training times, and the vacancy rate is set to 0.01.

A variety of machine learning algorithms were used for comparative analysis, and the results are shown in Table 1.

Table 1. The results of the comparative experiment

Model	Accuracy	Recall	Precision	F1	AUC
ExtraTrees	0.954	0.954	0.947	0.946	0.995
CatBoost	0.938	0.938	0.924	0.927	0.981
Decision tree	0.954	0.954	0.955	0.954	0.965
Random forest	0.952	0.952	0.945	0.946	0.984
XGBoost	0.973	0.973	0.973	0.973	0.993
AdaBoost	0.751	0.751	0.746	0.714	0.935
BP neural network	0.684	0.684	0.47	0.557	0.8
Logistic regression	0.753	0.753	0.717	0.726	0.919
RBMO-BiLSTM-Attention	0.987	0.987	0.987	0.987	0.997

Output the index comparison histogram of each algorithm, as shown in Figure 7.

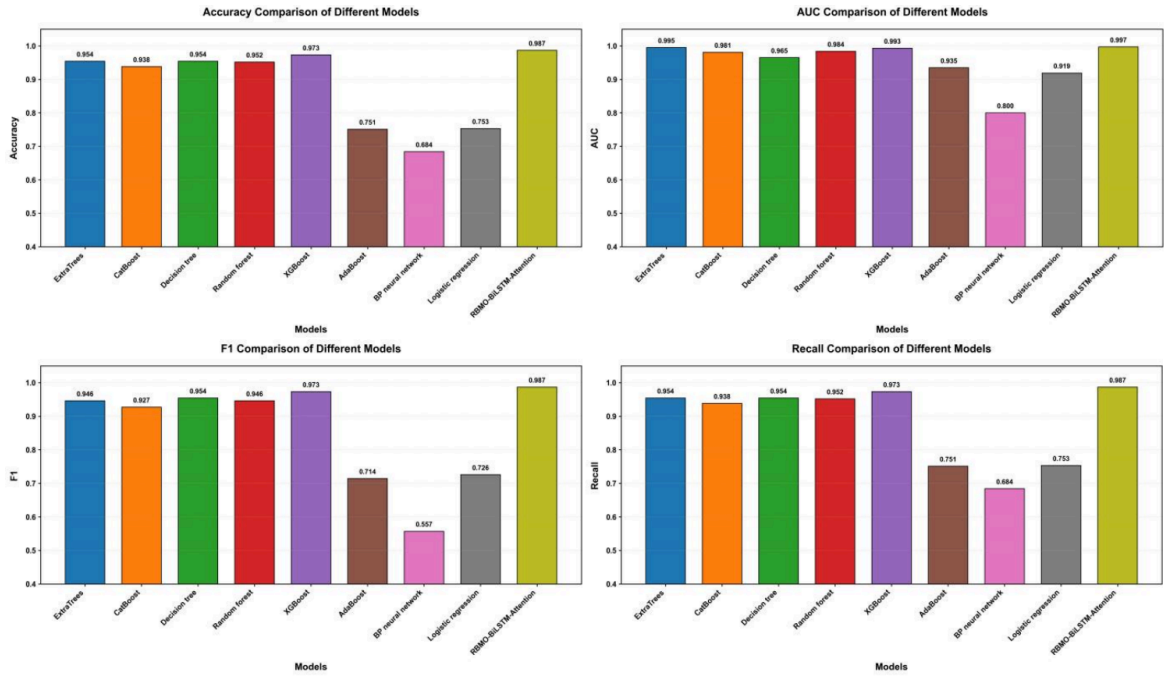


Figure 7. The index comparison results of the RBMO-BiLSTM-Attention test set

The RBMO-BiLSTM-Attention algorithm proposed in this paper shows the best performance in all evaluation indexes, and its accuracy, recall, accuracy, F1 score and AUC values reach 98.7%, 98.7%, 98.7%, 98.7% and 99.7% respectively. The overall classification ability and feature discrimination effect are significantly better than other comparison algorithms. In the traditional ensemble learning algorithm, the comprehensive performance of XGBoost is second only to the

DOI: 10.54254/2755-2721/2026.33932

proposed model, and all indicators are stable and maintained above 97%. The overall performance of ExtraTrees, Decision Tree and Random Forest is similar, and the accuracy rates are all maintained at about 95%. The overall performance of CatBoost is slightly weaker, and all indicators are concentrated in the range of 93% to 98%.

5. Conclusion

Embedded system is the core support of key security domain, and its software reliability directly affects system operation security. With the increasingly complex functions of software, the frequent occurrence of dynamic faults and the difficulty of effective detection by traditional methods, the existing machine learning models have limitations in long-time processing and parameter optimization, and it is difficult to meet the requirements of high precision and real-time performance. In this paper, the RBMobilsTMAttention algorithm is constructed, which integrates Red-billed Blue Magpie optimization, bidirectional long-short-term memory network and Attention mechanism, and automatically optimizes the network structure and key parameters, enhances the time series feature extraction and key fault information focusing. Comparative experiments show that the algorithm is better than the ensemble learning methods such as ExtraTrees, CatBoost, Decision Tree, Random Forest and XGBoost in classification accuracy, recall rate, accuracy rate, F1 score and AUC, and also significantly better than shallow models such as AdaBoost, logistic regression and BP neural network. Among them, XGBoost performs second, the traditional tree model has similar performance, and the shallow model has obvious shortcomings. This research effectively solves the problems of insufficient feature mining, weak time series correlation capture and insufficient model generalization ability in multi-classification fault prediction of embedded software, provides an efficient and feasible scheme for intelligent fault diagnosis of embedded systems, and has important engineering application value for improving the running stability and fault early warning ability of embedded software in key safety areas.

References

[1] Sahu, Santosh Kumar, et al. "A LSTM-FCNN based multi-class intrusion detection using scalable framework." Computers and Electrical Engineering 99 (2022): 107720.
[2] Dhibi, Khaled, et al. "Improved Machine Learning for Multiclass Fault Classification in Industrial Processes." IEEE Access 13 (2025): 211747-211764.
[3] Ruan, Hang, et al. "Deep learning-based fault prediction in wireless sensor network embedded cyber-physical systems for industrial processes." Ieee Access 10 (2022): 10867-10879.
[4] McLaughlin, Connor, and Yi Lu. "Multi-class vulnerability prediction using value flow and graph neural networks." Neural Computing and Applications 36.25 (2024): 15869-15891.
[5] Kerarmi, Abdelouadoud, Assia Kamal-Idrissi, and Amal El Fallah Seghrouchni. "Optimized Machine Learning Tree-Based Fuzzy Logic Model for Multi-Class Classification Using Integrated Truth Table." SN Computer Science 6.6 (2025): 683.
[6] Chen, Xiaofang, Zengli Mao, and Chong Wu. "Multi-class financial distress prediction based on feature selection and deep forest algorithm." Computational Economics 66.4 (2025): 2715-2754.
[7] Vohra, Rajan, et al. "Multi-class classification algorithms for the diagnosis of anemia in an outpatient clinical setting." Plos one 17.7 (2022): e0269685.
[8] Ismail, Heba, et al. "RL-ECGNet: resource-aware multi-class detection of arrhythmia through reinforcement learning: Ismail et al." Applied Intelligence 53.24 (2023): 30927-30939.
[9] Rajagopalan, Sudhar, Jaskaran Singh, and Ashish Purohit. "VMD-based ensembled SMOTEBoost for imbalanced multi-class rotor mass imbalance fault detection and diagnosis under industrial noise." Journal of Vibration Engineering & Technologies 12.2 (2024): 1457-1478.
[10] Katkam, Srinivas, et al. "Multi-Class Diagnosis of Neurodegenerative Diseases Using Effective Deep Learning Models With Modified DenseNet-169 and Enhanced DeepLabV 3+." IEEE Access 13 (2025): 29060-29080.