Available online www.jsaer.com

Journal of Scientific and Engineering Research, 2020, 7(11):133-140

Research Article

ISSN: 2394-2630
CODEN(USA): JSERBR

Fault Classification Model of Railway Freight Car Rolling Bearings based on GBDT Algorithm

Hongkun Wang[1], Honghui Li*[2,3], Yufeng Wang[2], Mengqi He[2]

[1]Shenhua Railway Equipment Co., Ltd. Beijing, China
[2]School of Computer and Information Technology, Beijing Jiaotong University Beijing, China
[3]Engineering Research Center of Network Management Technology for High Speed Railway Ministry of Education Beijing, China hhli@bjtu.edu.cn

Abstract To improve the maintenance accuracy of mechanical fault, further mining of industrial data has become an important means. Through the research on the information collection system of rolling bearings and the in-depth mining of the running state data of rolling bearings of heavy haul freight cars, a GD-XGBoost fault classification model for rolling bearings of freight trains is proposed. This method combines the Gradient Boosting Decision Tree algorithm (GBDT) and the improved Gradient Boosting algorithm (XGBoost) to optimize the slow training and over-fitting problems in training. And using the heavy haul freight cars operating state data collected by the railway safety monitoring system to conduct comparative experiments, it is found that the model improves the accuracy of fault classification, reduces the training time, and has good application value for improving the efficiency of heavy haul freight cars real-time fault classification.

Keywords GBDT; XGBoost; Rolling bearing; Fault classification

1. Introduction

Rolling bearing is a key component to support mechanical rotation. Once it fails, it will directly affect the smooth operation of the whole system, and even cause huge economic losses and safety accidents [1]. The traditional maintenance method of mechanical failure is regular scheduled maintenance [2], which is characterized by time-consuming, laborious and high cost. At present, condition based maintenance (CBM) which is based on the status of parts and components to decide whether to maintain mechanical equipment or not is the mainstream method for large-scale equipment maintenance in various countries.

With the establishment of the ground to vehicle safety monitoring and early warning system (4T) for heavy haul freight cars in China, and the continuous development of big data and deep learning technology in recent years, data-driven fault diagnosis technology [3] can play an important role in the condition based maintenance of heavy haul freight cars. The 4T system is composed of four subsystems: Trackside Acoustic Detection System(TADS), the infrared axle temperature detection system (THDS), Truck Performance Detection System (TPDS), and Trouble of moving Freight car Detection System (TFDS) [4]. This paper makes full use of the running state data of the key parts of heavy haul freight cars from 4T, uses GBDT to sort the importance of features, and combines with XGBoost model, puts forward the GD-XGBoost classification model for rolling bearing fault classification and diagnosis.

2. Materials and Methods

2.1. Gradient Boosting decision Tree

The Gradient Boosting decision Tree(GBDT) algorithm was first completely proposed by Jerome H. Friedman of Stanford University in 1999. The algorithm can realize regression, classification and sorting. There are three main components of GBDT: Regression Decision Tree, Gradient_Boosting, and Shrinkage. The boosting algorithm is composed of a series of "weak learners", which realize a strong learner through a certain linear combination, although the classification or regression effect of these "weak learners" may be only a little better than random classification or regression, the final combination of "strong learners" can output a good prediction result. In the gradient boosting decision tree, the "weak learner" used is the classification regression tree (CART). Since the goal of GBDT fitting is a gradient, which is always a continuous value, it is said that in the gradient boosting decision tree, whether it is classification or regression problems, the base classifier uses the regression tree uniformly [5-6].

2.1.1. Forward distribution algorithm

Suppose a data set$\{(x_1, y_1), (x_2, y_2), \dots, (x_n, y_n)\}$, each data item x_i is a vector representing the characteristic attribute of a transaction, if $y_i \in R$, it is a regression problem. If $y_i \in \{-1,1\}$, it is a classification problem. By (m-1)-th iteration, a set of "weak learners" $\{k_1, k_2, \dots, k_{m-1}\}$is obtained, then "strong learner" C_{m-1} can be obtained from formula (1).

$$C_{m-1}(x_i) = \alpha_1 k_1(x_i) + \cdots + \alpha_{m-1} k_{m-1}(x_i) \quad (1)$$

Among them, α is the weight of k, and m>1. After another iteration, the "strong learner" C_m is obtained, as shown in formula (2).

$$C_m(x_i) = C_{m-1}(x_i) + \alpha_m k_m(x_i) \quad (2)$$

The effect of C_m is better than that of C_{m-1}, because after (m-1)th iteration, the algorithm will increase the weight of the samples with prediction errors, which makes them pay more attention to these samples in mth iteration, so as to achieve the purpose of correcting the error of the previous iteration.

2.1.2. Gradient boost

Suppose $F(x)$ realize the fitting to the sample (x, y), that is, $y = F(x)$. However, the effect of fitting is not very good, and it cannot achieve error-free fitting to all samples in the sample set, that is, there will always be residuals for each sample, as in formula (3).

$$F(x_1) + h(x_1) = y_1, F(x_2) + h(x_2) = y_2, \dots, F(x_N) + h(x_N) = y_N \quad (3)$$

In the formula (3), $h(x)$ is called the residual of the sample x, and the formula (3) is equivalent to the following formula (4).

$$h(x_1) = y_1 - F(x_1), h(x_2) = y_2 - F(x_2), \dots, h(x_N) = y_N - F(x_N) \quad (4)$$

It can be seen that without changing $F(x)$, only $h(x)$ needs to be optimized to improve the fitting effect. Suppose $F_0(x)$ is the initial fitting function, $h_1(x)$ is the residual obtained after optimization processing, and the new fitting function $F_1(x)$ is formula (5).

$$F_1(x) = F_0(x) + h_1(x) \quad (5)$$

If the fitting effect of $F_1(x)$ is still not up to expectations, it is necessary to continue to optimize the residual $h_2(x)$ for $F_1(x)$ to obtain $F_2(x)$, as in formula (6).

$$F_2(x) = F_1(x) + h_2(x) = F_0(x) + h_1(x) + h_2(x) \quad (6)$$

Suppose a total of m times of iterative optimization, the final fitting function is $F_m(x)$ is the formula (7).

$$F_m(x) = F_{m-1}(x) + h_m(x) = F_0(x) + \sum_{i=1}^{m} h_m(x) \quad (7)$$

Formula (7) shows a process of optimizing residuals to gradually improve performance. Here we can use the method of decision tree to optimize the residual, such as the existing model $F_m(x)$ (i.e. the fitting function), where $h_{m+1}(x)$ needs to be obtained to establish $F_{m+1}(x)$ model, the required data set is: $\{(x_1, h_m(x_1))$,$(x_2, h_m(x_2))$,…,$(x_N, h_m(x_N))\}$. From formula (7), it can be rewritten as formula (8).

$$\{(x_1, y_1 - F_{m-1}(x_1)), \dots, (x_N, y_N - F_{m-1}(x_N))\} \quad (8)$$

Use the data set of formula (8) to construct the decision tree $h_{m+1}(x)$. Compared with the traditional decision tree, the response value corresponding to x in the data of the decision tree $h_{m+1}(x)$ is not its true response value y, but the residual $h_m(x)$. Here we put $h_{m+1}(x)$ is called "pseudo response value". After the decision tree $h_{m+1}(x)$ is constructed, the decision tree is used to predict the sample data, and the pseudo response value $h_{m+1}(x)$ obtained is the residual required by $F_{m+1}(x)$ model. When the loss function is a square loss, the optimization model based on residual is the optimization $F(x)$ based on negative gradient. When the loss function is not a square loss function, suppose J be the loss function. We need to minimize J by changing each sample $F(x_i)$, where J is formula (9).

$$J = \sum_{i=1}^{N} L(y_i, F(x_i)) \tag{9}$$

Put $F(x_i)$ as a parameter, and the partial derivative of J is obtained (10).

$$\frac{\partial J}{\partial F(x_i)} = \frac{\partial \sum_{i=1}^{N} L(y_i, F(x_i))}{\partial F(x_i)} \tag{10}$$

Further, we can get the formula (11).

$$F_m(x) = F_{m-1}(x) + h_m(x) = F_{m-1}(x) - \frac{\partial J}{\partial F_{m-1}(x_i)} \tag{11}$$

Formula (11) can be used to optimize $F(x)$ based on negative gradient. It has been proved that the algorithm using gradient optimization is more general than using residuals, because other functions can be used to define the loss function.

The steps of gradient boosting decision tree algorithm are summarized as follows:

(1) Initialize $F_0(x)$.

For classification problems, $F_0(x)$ is set to 0;

(2) The iterative process m-1 to M is optimized.

1) Calculate the negative gradient, which is the pseudo-residual $h_m(x)$;

2) $h_{m+1}(x)$ is constructed from the data set composed of the residuals;

3) The new pseudo-residual $h_{m+1}(x)$ is predicted by the decision tree $h_{m+1}(x)$;

4) Get the new model $F_{m+1}(x)$.

(3) Output $F_M(x)$.

When using GBDT to predict samples, for regression problem, the prediction result of sample x is $y = F_M(x)$ is shown in formula (12).

$$F_M(x) = F_{M-1}(x) + vh_M(x) = F_0(x) + \sum_{m=1}^{M} vh_m(x) \tag{12}$$

In the formula, $F_0(x)$ and v are the initial fitting function and shrinkage factor used when constructing the GBDT model, respectively. $h_m(x)$ is the "weak learner". In the multi-classification task, for the sample x, $\mathrm{F}_M^{(k)}(x)$of each sub-category of all K categories is first obtained, as shown in formula (13).

$$F_M^{(k)}(x) = F_{M-1}^{(k)}(x) + vh_M(x) = F_0^k(x) + \sum_{m=1}^{M} vh_m^{(k)}(x), k = 1,2,\ldots,K \tag{13}$$

The classification corresponding to the maximum value in $\mathrm{F}_M^{(k)}(x)$ is the prediction result of sample X.

2.2. Research on Fault Classification Model

2.2.1. Feature importance ranking

When selecting features, machine learning algorithms often rely on manual calculation, such as Pearson correlation coefficient, covariance, etc. this method not only has low calculation efficiency, but also only considers the association relationship between features and target classes, which cannot measure the correlation relationship between features and target classes as a whole. The principle of GBDT has been mentioned in the first section of this article. And its advantages are strong algorithm integration, high accuracy, and interpretability. This article uses GBDT algorithm to process the input features, calculates the importance of each feature through model training, and then selects the features according to the feature importance score of the model output, which lays a good foundation for the classification of XGBoost model in the next step.

The method for GBDT to calculate the importance of features is to first calculate the importance of each feature on that tree on each tree, observe how much each feature contributes to each tree, and then calculate the importance of this feature in all trees. The contribution of the tree is averaged, and the importance ranking of the feature in the entire model is obtained [7].

When using GBDT for classification, the $\mathrm{Gini(D)}$ is usually used to calculate the importance of features on each tree. The calculation formula of the characteristic Gini index of node a is as formula (14).

$$\mathrm{Gini(a)} = 1 - \sum_{k=1}^{|K|} p_{ak}^2 \tag{14}$$

Here, K indicates that there are currently K categories in total, and p_{ak} represents the proportion of feature k in node a. Generally speaking, $\mathrm{Gini(a)}$ can be understood as the probability that two samples are randomly taken from node a, and the two samples do not belong to the same category. The importance of feature X_i on node a is represented by the reduction of feature impurity. See formula (15).

$$FIM_{ia}^{Gina} = N_a \times Gina(a) - N_l \times Gini_l - N_r \times Gini_r \tag{15}$$

Among them, FIM represents Feature Importance Measures, N_a represents the number of samples contained in node a, N_1 represents the number of samples of left child l of node a, and N_r represents the number of samples of right child r of node a.

Assuming that the set of nodes that feature X_i appears in the p-th decision tree is M, the feature importance score of feature X_i at node M can be expressed as formula (16).

$$\mathrm{FIM}_{pi} = \sum_{m \in M} FIM_{im} \tag{16}$$

The above is the feature calculation method of computing features on a single decision tree. It is extended to n decision trees and the score of N trees is obtained. See formula (17).

$$\mathrm{FIM}_i = \sum_{j=1}^{n} FIM_{ji} \tag{17}$$

Finally, the calculated feature importance score is normalized and the formula (18) is obtained.

$$\mathrm{FIM}_i = \frac{FIM_i}{\sum_{j=1}^{c} FIM_j} \tag{18}$$

Where c is the total number of features. Through the above steps, the importance score of each feature in the sample can be obtained.

2.2.2. GD-XGBoost fault classification model

GBDT has achieved better classification effect than traditional machine learning algorithms such as Support Vector Machine, Logistic Regression, Native Bayes, etc. by incorporating integrated learning, gradient boosting, and forward learning mechanisms. However, GBDT still has the following problems in classification: (1) The GBDT algorithm only multiplies each base classifier by a coefficient λ (0<λ<1) to reduce the influence of each classifier on the whole model to prevent over fitting, but does not consider the inclusion of the correctness within each base classifier, and there is still the risk of over fitting. (2) GBDT can calculate the first derivative of loss function. Although the iteration is simple, the computation and storage are small, the problem of slow convergence is existed in the first derivative. Finally, GBDT is completely serial in the process of tree building, and the training speed is relatively slow.

XGBoost is an improved gradient boosting tree algorithm [8]. Compared with the traditional gradient boosting decision tree algorithm, it has the following advantages [9]:

(1) The regularization term is introduced into the objective function, as shown in formula (19).

$$\mathrm{Obj} = \sum_{i=1}^{n} l(y_i , \hat{y_i}) + \sum_{k=1}^{K} \Omega(f(k)) \tag{19}$$

In the formula, the first term is the training loss, that is, the difference between the actual value and the predicted value, and the second term is the regular term added, which is expanded into formula (20).

$$\Omega(f_t) = \lambda T + \frac{1}{2}\lambda \sum_{j=1}^{T} w_j^2 \tag{20}$$

In the formula, the first term is the penalty intensity, where T represents the number of leaf nodes, the second term is the L2 penalty term, and wj represents the prediction score of the j-th leaf node. Therefore, the XGBoost model further improves the generalization ability of the model by adding regular terms to each subtree.

(2) Rewrite the objective function. According to the second-order Taylor expansion formula (21).

$$f(x + \Delta x) = f(x) + f^{'}(x)\Delta(x) + \frac{1}{2} f''(x)\Delta x^2 \tag{21}$$

Simplify the objective function to formula (22).

$$Obj = -\frac{1}{2}\sum_{j=1}^{T} \frac{G_j^2}{H_j + \lambda} + \gamma T \tag{22}$$

Among them:

$$G_j = \sum_{i \in I_j} g_i\ ,\quad H_j = \sum_{i \in I_j} h_i \tag{23}$$

$$g_i = \partial_{\hat{y}(t-1)} l(y_i, \hat{y}(t-1)),\quad h_i = \partial^2_{\hat{y}(t-1)} l(y_i, \hat{y}(t-1) \tag{24}$$

In formula (24), g_i, h_i is the first derivative and the second derivative of the error function respectively. By solving the second derivative, the convergence speed is accelerated. At the same time, we can see that there is no correlation between each sample in the process of solving the first derivative and second derivative, which greatly improves the parallelism of the algorithm.

The GBDT algorithm can calculate the importance of each feature by scoring the input features. Compared with other feature selection methods, this method does not require complicated calculation details and can mine the complex relationships between features. This paper combines GBDT algorithm and XGBoost algorithm, and proposes a GD-XGBoost algorithm to realize the fault classification of rolling bearing.

In GD-XGBoost algorithm, firstly, the data after feature preprocessing is transferred to GBDT model for training, and then the importance score of each feature is calculated by GBDT model, and the features are classified according to the feature score; then the features to be classified are transferred to XGBoost. According to the training results of XGBoost model, the prediction probability of various rolling bearing fault types is output and the fault classification is completed.

3. Experiment and Results

3.1. Experiment

In this experiment, we first extract the state data of the freight car parts from different sensors, simultaneous interpreting the missing values, outliers and redundant values from three railway freight cars detection systems of THDS, TPDS and TADS.

Fig. 1 shows the temperature distribution on each bearing of the truck with heat shock alarm and vehicle NO. 0005976 passing through each detection station in sequence. It can be seen from the figure that the temperature of axle 6 on the right side is significantly higher than that of other axles when the truck passes the section from Dongzhi to Xibanpo. From Fig. 2, it can also be seen that the temperature rise and temperature rise offset of the axle on the section from Dongzhi to Xibanpo are larger than those measured on other sections.

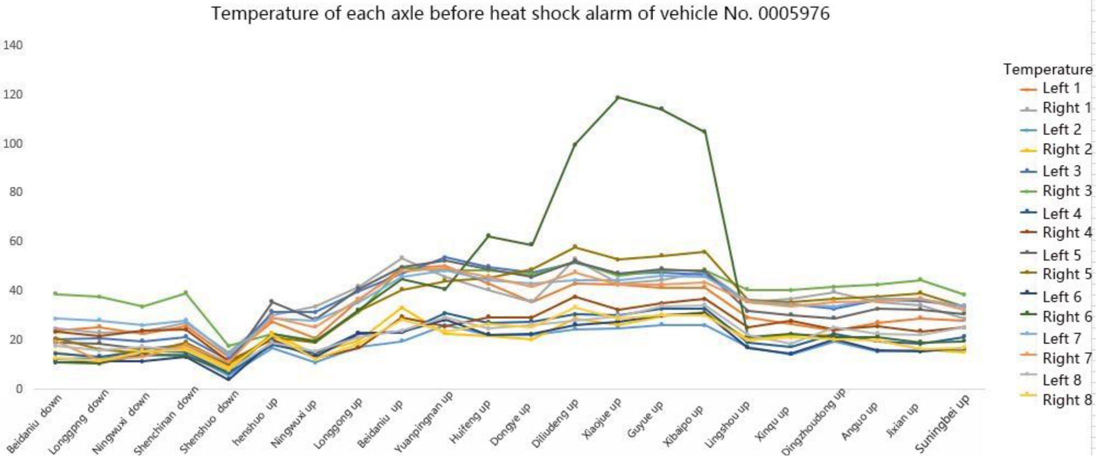

Figure 1: Bearing temperature fluctuation

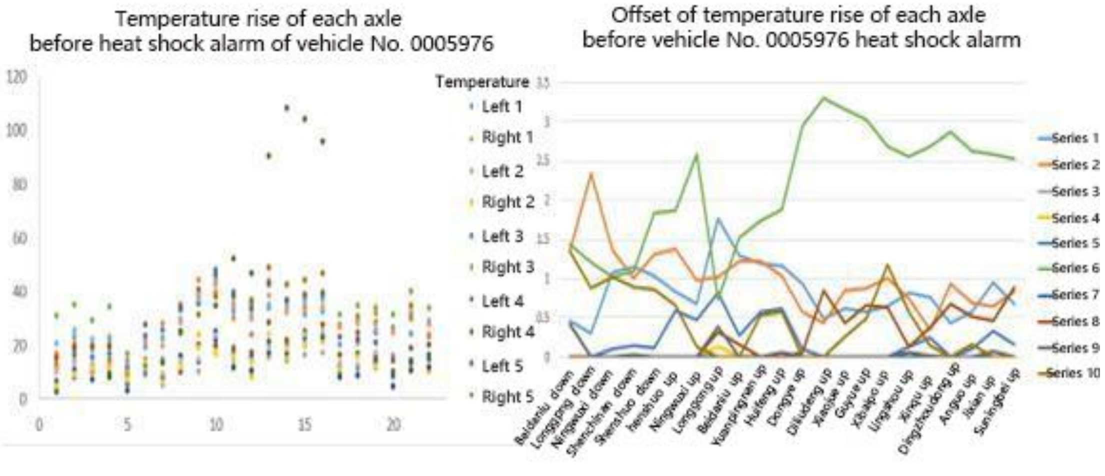

Figure 2: Bearing temperature rise and offset fluctuation

Then, the data after feature preprocessing is transferred to GBDT model for training, and the importance score of each feature is output through GBDT model. At this time, the features with feature score greater than 120 are selected and transferred to XGBoost model for training. The specific parameter configuration of XGBoost is shown in Table 1, and finally output the predicted probabilities of the model for various rolling bearing failure types. The specific implementation process is shown in Fig. 3, where NM (Normal Condition) indicates that the bearing state is normal, IF (Inner Fault) refers to inner ring fault, OF (Outer Fault) refers to outer ring fault, and BF (Ball Fault) refers to rolling element fault.

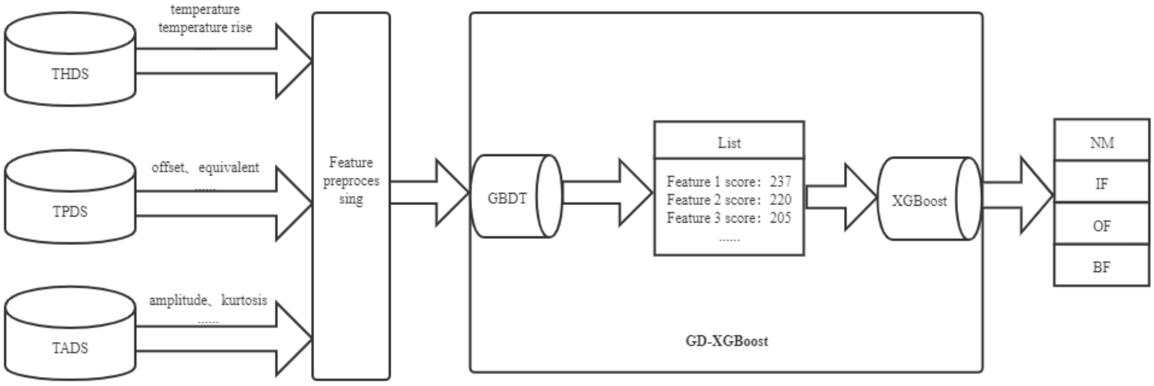

Figure 3: GD-XGBoost fault diagnosis flowchart

Table 1: XGBoost parameters

Parametric variable	Parameter meaning	Parameter value
eta	Learning rate	0.03
min_child_weight	Sum of minimum leaf node weight	1
gamma	Minimum loss function	0.02
subsample	sampling rate	0.7
max_depth	The maximum depth of tree	5
max_leaf_nodes	Maximum number of leaf nodes	8

3.2. Results Analysis

Fig. 4 shows the change curves of loss value under different algorithms when using GBDT, XGBoost and GD-XGBoost algorithms to classify rolling bearing fault. It can be seen that due to the use of second-order Taylor expansion and optimization in parallel operations, XGBoost and GD-XGBoost have a faster convergence rate than GBDT. GBDT converges at the 25th epoch, XGBoost converges at the 20th epoch, and GD-XGBoost converges at the 18th epoch. At the same time, compared with XGBoost, GD-XGBoost can train the loss function to a lower level, about 0.2.

Table 2 shows the comparison of the training time and classification accuracy of the three models of GBDT, XGBoost, and GD-XGBoost. Compared with GBDT and XGBoost, the accuracy of GD-XGBoost is increased by 8% and 5% respectively. In terms of training time, GD-XGBoost and XGBoost are significantly improved compared with GBDT, and GD-XGBoost has the largest improvement, reaching more than 81%. The above analysis shows that the GD-XGBoost algorithm has faster training speed and higher classification accuracy.

Table 2 shows the comparison of the training time and classification accuracy of the three models of GBDT, XGBoost, and GD-XGBoost. Compared with GBDT and XGBoost, the accuracy of GD-XGBoost is increased by 8% and 5% respectively. In terms of training time, GD-XGBoost and XGBoost are significantly improved compared with GBDT, and GD-XGBoost has the largest improvement, reaching more than 81%. The above analysis shows that the GD-XGBoost algorithm has faster training speed and higher classification accuracy.

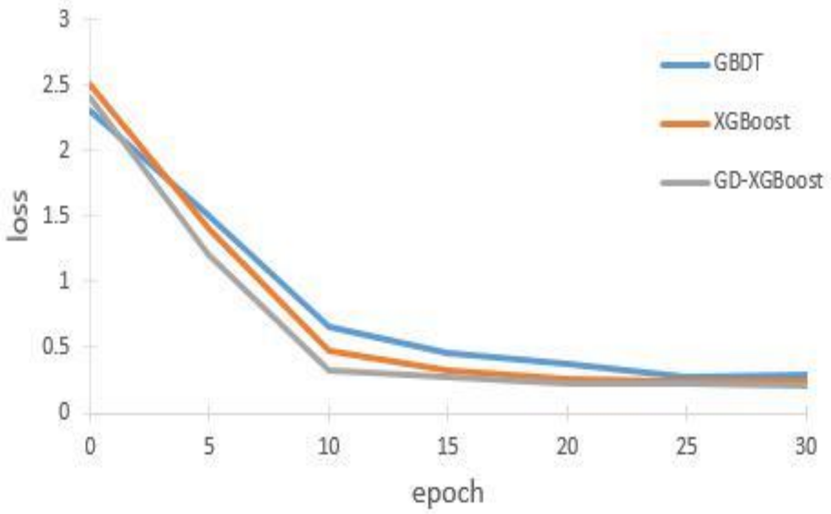

Figure 4: Loss under different models

Table 2: Accuracy and training time under different models

Models	Accuracy	Training time
GBDT	0.81	300s
XGBoost	0.85	70s
GD-XGBoost	0.89	55s

Table 3: Accuracy under different fault types

Algorithm	NM	IF	OF	BF	Average
GBDT	0.83	0.78	0.79	0.84	0.81
XGBoost	0.87	0.83	0.84	0.86	0.85
GD-XGBoost	0.91	0.87	0.88	0.90	0.89

4. Conclusion

This paper uses GBDT's ability to rank feature importance for feature purification, the GD-XGBoost fault classification model is proposed by combining GBDT with XGBoost model, and the GD-XGBoost model is used for rolling bearing fault classification. Experiment with the features obtained from THDS, TPDS and TADS monitoring systems. Through experimental analysis, compared with GBDT and XGBoost algorithm, GD-XGBoost algorithm has faster training speed and higher accuracy. In future research, we are committed to applying the method to more fields, and aiming at the problems in practical application, to improve and modify it.

Acknowledgment

This research was supported by Scientific and Technological Innovation Project of Shenhua Group (Grant No. SHGF-17-56).

References

[1]. Chi Yongwei, Yang Shixi, Jiao Weidong. Multi label classification method for rolling bearing fault based on lstm-rnn [J]. *Vibration, test and diagnosis, 2020,40 (3):* 562-571

[2]. Wu Guangning, Li Xiaonan, Yang Yan, et al. Research progress on fault prediction and health management of on board transformer [J]. *High voltage technology, 2020,46 (3):* 876-889

[3]. Zhang Ni, Che Lizhi, Wu Xiaojin. The current situation and Prospect of data driven fault diagnosis technology [j]. *Computer science, 2017,44 (z1):* 37-42.

[4]. Zhou Ke, LV min, Wang Gang, et al. Monitoring data transmission technology of 5T system based on message [J]. *China Railway Science, 2008,29 (5):* 112-118

[5]. Jerome H. Friedman. Greedy Function Approximation: A Gradient Boosting Machine[J]. *The Annals of Statistics, 2001, 29(5):*1189-1232.

[6]. Jerome H. Friedman. Stochastic gradient boosting[J].*Computational statistics & data analysis,2002,38(4):*367-378.

[7]. Liu Jinshuo, Liu Biwei, Zhang Mi, et al. Fault prediction of power metering equipment based on GBDT [J]. *Computer science, 2019,46 (z1):* 391-396.

[8]. Chen T, Guestrin C. Xgboost: A scalable tree boosting system[C]//*Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining. ACM, 2016:*785-794.

[9]. Zhang Yu, Chen Jun, Wang Xiaofeng, et al. Application of Xgboost in rolling bearing fault diagnosis [J]. *Noise and vibration control, 2017,37 (4):* 166-170,179

[10]. Wang, Xiukang, Yue, Wenjun, Lu, Xianghui,et al. Evaluation of SVM, ELM and four tree-based ensemble models for predicting daily reference evapotranspiration using limited meteorological data in different climates of China[J].*Agricultural and Forest Meteorology,2018,263:*225-241.

[11]. Yang Shaoqing. Technical development of heavy haul freight cars in China [J]. *Rolling stock, 2009, 47 (12):* 1-5.

[12]. Hao Junhu, Hu Yi. Research on bearing fault diagnosis and early warning method based on XGBoost and autoregressive model [J]. *Modular machine tool and automatic machining technology, 2020, (2):* 140-142,157

[13]. Jiang Hui, Yu Bingchun, Liu chunhuang, et al. Research on operation safety monitoring system of railway passenger and freight vehicles [C] / / *the 9th China Intelligent Transportation annual meeting. 2014:* 807-813.

[14]. Jiang Shaofei, Wu Tianji, Peng Xiang, et al. Data driven fault diagnosis method based on xgboost feature extraction [J]. *China Mechanical Engineering, 2020,31 (10):* 1231-1239

[15]. Yang F, Xiao D. Model and fault inference with the framework of probabilistic SDG[C]//*2006 9th International Conference on Control, Automation, Robotics and Vision. IEEE, 2006:* 1-6.