

Interpretable Learning Algorithm Based on XGBoost for Fault Prediction in Optical Network

Chunyu Zhang[1], Danshi Wang[1], Chuang Song[1], Lingling Wang[1], Jianan Song[1], Luyao Guan[1], Min Zhang[1]

[1] State Key Laboratory of Information Photonics and Optical Communications, Beijing University of Posts and Telecommunications (BUPT), Beijing, 100876, China;

xyzhang@bupt.edu.cn

Abstract: We propose a fault prediction scheme using interpretable XGBoost based on actual datasets, which not only achieves high accuracy (99.72%) and low positive rate (0.18%), but also reveals the five most remarkable features that caused the fault.

1. Introduction

Recently, artificial intelligence (AI) technology has been explored actively for dealing with fault prediction in optical networks [1, 2]. The neural network (NN) could execute fault prediction through implicitly learning from features (i.e., physical parameter of equipment) [3]. However, the NN acts as a black-box so that difficult to explain why it works and analyze how it operates, which brings serious dilemmas to the network operators [4].

Different from traditional NNs, extreme gradient boosting (XGBoost) adopts the CART regression tree as the base learner and uses the objective function as the basis to find the best split, construct tree structure and calculate leaf score [5]. Inspired by the idea that XGBoost finds the best split through feature splitting continuously, it is valuable to introduce XGBoost to reveal which features are more important for fault prediction.

In this paper, we propose a fault prediction scheme based on the interpretability of XGBoost. By analyzing the internal structure of XGBoost, the five most remarkable features that caused the fault were revealed. The data used in the experiments come from the records of the existing OTN of a telecommunication operator. The classical statistical characteristics (i.e., accuracy and false positive rate) are used to evaluate performance of scheme proposed. Compared with Gradient Boosting Decision Tree (GBDT) and Deep Neural Network (DNN), the proposed scheme achieves higher predictive accuracy (i.e., 99.72%) and lower false positive rate (i.e., 0.18%).

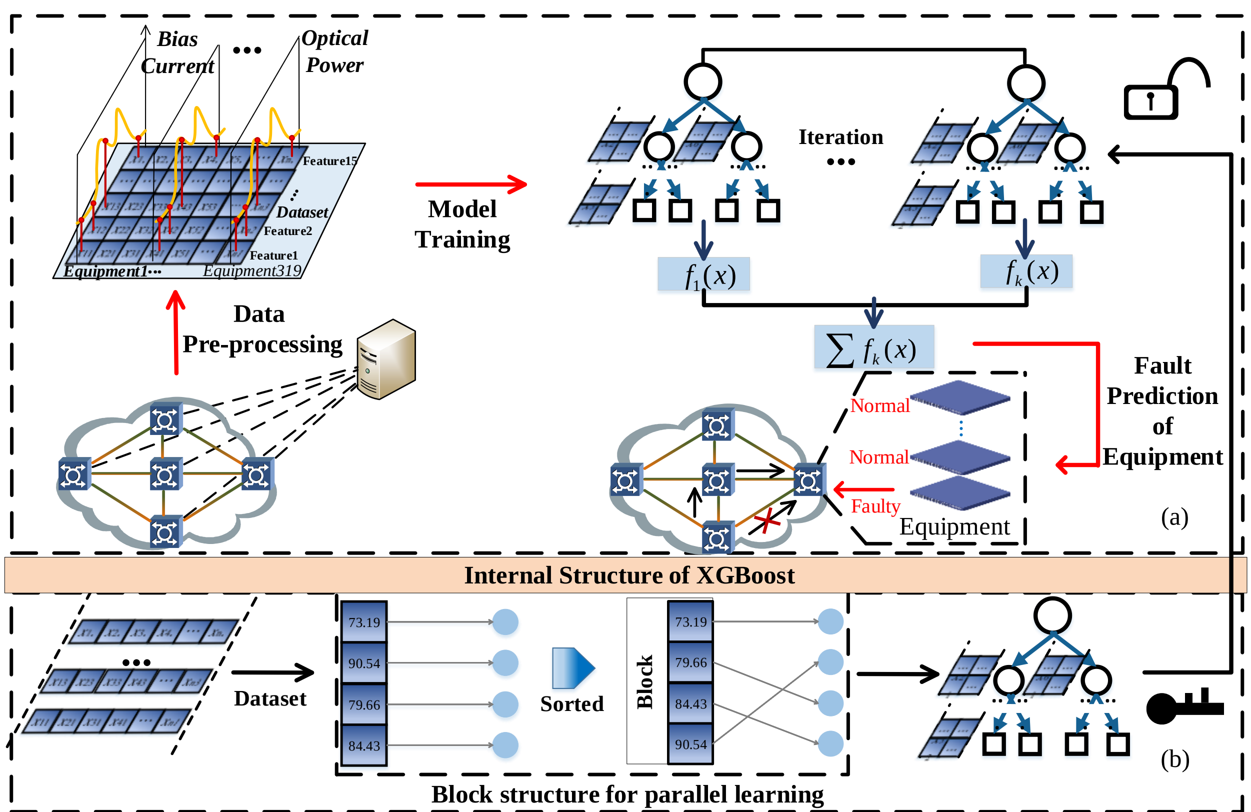

Fig. 1. Fault prediction of equipment based on interpretable XGBoost in optical network.

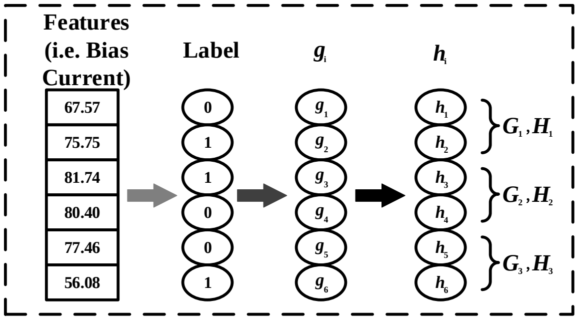

Fig. 2. Split finding.

Table 1. Experimental Parameters

Parameters	Value
Max_depth	11
Min_weight	2
Gamma	0.1
Reg_lambda	1.5
Learning_rate	0.01
Subsample	0.9
N_estimators	1800

2. Fault Prediction Based on Interpretable XGBoost

The schematic diagram of the fault prediction based on XGBoost is shown in Fig. 1 (a), which includes data pre-processing, model training and fault analyzing. Data pre-processing is to prepare for model training. XGBoost is mainly used for model training. Fault analyzing is used to explain the physical mechanism of equipment fault.

2.1 Feature Scoring

Feature scoring can be seen as the visual presentation of XGBoost interpretability. The XGBoost algorithm counts out the importance by “gain”, “frequency”, and “cover” [6]. We choose “frequency” to calculate feature importance score, and the “frequency” is the number of times that a feature is used to split node across all trees. We know that XGBoost needs to enumerate features when splitting note, and sort out the feature which makes objective function achieve the best value [7]. Therefore, the more times a feature splits, the higher correlation is obtained between the feature and the fault state, and the higher score is achieved. In addition, XGBoost sorts the feature values before finding the best split and saves it as the “block”. Meanwhile, the “block” can help XGBoost enumerate features in parallel. It is seen in Fig. 1 (b) that each column in a block is sorted by the corresponding feature value and a linear scan over one column in the block is sufficient to enumerate all the split points [5]. Especially, the sorting process takes place only once, which can reduce the complexity of the model.

2.2 Split Finding

We know that XGBoost uses the objective function as the basis for finding the best split. So how to find the best split? XGBoost uses the same idea as the CART regression tree, which sorts out the largest gain by enumerating all the feature with using the greedy algorithm [8]. However, the greedy algorithm is just suitable for situations where the amount of data is not too large. When the amount of data is too large to fit entirely into memory, the exact greedy algorithm will run very slowly. So, the approximate algorithm is introduced. Different from the greedy algorithm, the approximate algorithm proposes candidate split points based on weighted quantile sketch, which need not to enumerate all splits. Firstly, the feature values need to be sorted, the candidate points will be found based on percentiles of feature distribution; then, the value of continuous feature will be mapped into buckets split by these candidate points and the best solution among proposals based on the aggregated statistics will be determined [5].

$$Gain = \frac{1}{2}\left[\frac{\left(\sum_{i\in L} g_i\right)^2}{\sum_{i\in L} h_i + \lambda} + \frac{\left(\sum_{i\in R} g_i\right)^2}{\sum_{i\in R} h_i + \lambda} - \frac{\left(\sum_{i\in I} g_i\right)^2}{\sum_{i\in I} h_i + \lambda}\right] - \gamma \quad \Rightarrow \quad Gain = \frac{1}{2}\left[\frac{G_L^2}{H_L + \lambda} + \frac{G_R^2}{H_R + \lambda} - \frac{(G_L + G_R)^2}{H_L + H_R + \lambda}\right] - \gamma \tag{1}$$

The gain of XGBoost is shown in Eq (1). It is seen that node will be split when the difference of objective function between after splitting the node and before splitting the node is higher than the threshold (γ). According to Eq (1), we choose the candidate point with the largest gain as the best split.

Where L and R are the subsets of left and right nodes after the instance set I splitting, g_i and h_i are the first and second order gradient statistics on the loss function. In addition, γ and λ are the tunable parameters used to adjust the degree of regularization.

The process of selecting the best split is described in details by using the first six feature values of laser bias current. The laser bias current is one of parameters collected from experimental data. As can be seen from Fig. 2, the value of the bias current has three candidate points. Then, the values of the bias current is mapped into the three buckets, and the G, H can be achieved based on the aggregated statistics. The Eq (2) can be obtained from Eq (1), and the candidate point with the largest gain as the best split.

$$Gain = max\left\{Gain,\ \frac{G_1^2}{H_1 + \lambda} + \frac{G_{23}^2}{H_{23} + \lambda} - \frac{G_{123}^2}{H_{123} + \lambda} - \gamma,\ \frac{G_{12}^2}{H_{12} + \lambda} + \frac{G_3^2}{H_3 + \lambda} - \frac{G_{123}^2}{H_{123} + \lambda} - \gamma\right\} \tag{2}$$

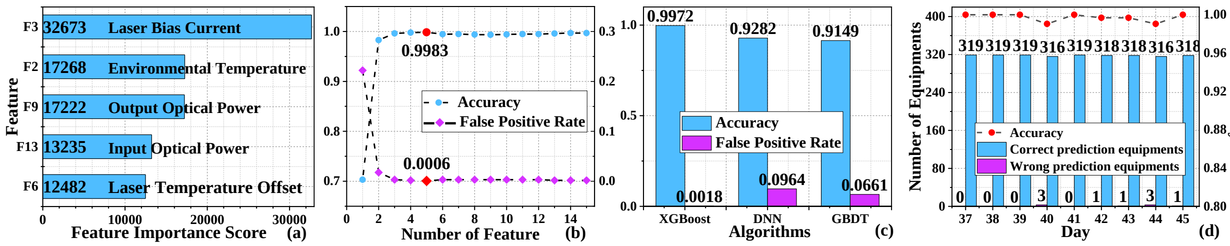

Fig. 3. (a) Feature importance score; (b) Accuracy and false positive rate change with the number of feature sorted; (c) Accuracy and false positive rate of different algorithms; (d) Number of fault equipments (Accuracy) vs. Day.

3. Experimental Results and Analysis

The performance of the proposed scheme is evaluated and analyzed, by using the data from the physical state records of the existing OTN of a telecommunication operator. The observed original data count up to 444,320, which consists of 15 features, including the optical power, the laser current, the environmental temperature, the unusable time and so on. The “unusable time” represents the failure state of the equipment and we assume a board as “failed” when the value of “unusable time” is larger than 40,000. After data pre-processing, totally 14,355 samples during 45 days are used as experimental data. Data in the first 36 days are used to train the XGBoost model and in the last 9 days are used in the XGBoost model to judge the equipment failure state.

In order to get the best training model for fault prediction, we perform repeated experiments by adjusting the XGBoost parameters. Firstly, we adjust the parameter of “max_depth” and “min_weight”, the higher prediction accuracy can be obtained when the parameters are set to 11 and 2 respectively. The “max_depth” and “min_weight” represent the maximum depth of a tree and the minimum sum of instance weight (hessian) needed in a child respectively, which are the important parameter that optimizes the structure of the tree. Then, the parameters of “gamma”, “reg_lambda”, “learning_rate”, “subsample” and “n_ estimators” are adjusted in turn, which are used to prevent overfitting. The process of parameter adjustment will stop if the XGBoost training model achieves the optimal training model, the optimal value of these parameters are shown in Table 1.

Based on the feature scoring of XGBoost, the score of the top 5 features are shown in Fig. 3 (a), in which, for example, the most related feature is the laser bias current when the equipment is close to fault. Feature selecting is to sort out the features highly correlative with the equipment fault, which can simplify the predictive model and improve the prediction accuracy in some degree. As shown in Fig. 3 (b), when selecting top 5 dimensional features as input, the scheme based on XGBoost has the highest prediction accuracy and the lowest false positive rate.

Accuracy and false positive rate are recognized as the important statistical characteristics in evaluating the performance of fault prediction. It will bring high economic loss to the actual network operator if the false positive rate is high. To evaluate the performance of the XGBoost, we compare it with the DNN and GBDT. It is seen in Fig. 3 (c) that XGBoost model can achieve higher prediction accuracy while lower false positive rate than DNN and GBDT models do. The average prediction accuracy and the false positive rate of our proposal is 97.72% and 0.18% respectively. As shown in Fig. 3 (d), from Day 37 to Day 45, we predict the equipment operating state and check the predicting results with the actual records, then record the prediction accuracy of each day. Most of these fault equipment are correctly predicted by using the scheme proposed and the average prediction accuracy is 97.72% according to calculating. The prediction accuracy is greater than 99%.

4. Conclusion

We proposed a fault prediction scheme based on interpretable XGBoost, which achieved high prediction accuracy while low false positive rate for fault prediction of equipment in optical networks. Moreover, by analyzing the physical mechanism of equipment fault using the results of fault prediction, the most significant causes of fault were obtained.

5. References

[1] Z. Wang et al., "Failure prediction using machine learning and time series in optical network", Opt. Exp., vol. 25, no. 16, 2017.
[2] X. Zhao et al., "Accurate fault location based on deep neural evolution network in optical networks for 5G and beyond", in OFC, M3J5, 2019.
[3] D. Cireșan et al., "Multi-column deep neural network for traffic sign classification", Neural Networks, vol. 32, 2012.
[4] Y. Wang, et al., "Short-term load forecasting with multi-source data using Gated Recurrent Unit neural networks", Energies, vol. 11, 2018.
[5] T. Chen et al., "Xgboost: A scalable tree boosting system", Proc. SIGKDD, pp. 785-794, 2016.
[6] J. Friedman, et al., "The elements of statistical learning", Berlin, vol. 1, 2001.
[7] X. Ren et al., "A Novel Image Classification Method with CNN-XGBoost Model", Lecture Notes in Computer Science, vol. 10431, 2017.
[8] L. Rutkowski, et al., "The CART decision tree for mining data streams", Int. J. Inform. Sci., vol. 266, 2014.