# LSTM-GAN-XGBOOST Based Anomaly Detection Algorithm for Time Series Data

Xin Xu[1], Hongbo Zhao[1,2], Haoqiang Liu[1], Hua Sun[1]
[1]Beihang University
[2]Hefei Innovation Research Institute, Beihang University
Beijing Haidian, 100191, China
buaaxu@buaa.edu.cn, bhzhb@buaa.edu.cn, lhqbuaa@buaa.edu.cn, sun@buaa.edu.cn

***Abstract*—Time series anomaly detection is an important and fundamental task of Prognostic and Health Management (PHM). Traditional anomaly detection algorithms can achieve the detection of shallow features when dealing with the nonstationary time series data, yet those algorithms fail to detect outliers on deep features of massive time series data. In this paper, we proposed a novel fusion algorithm named LSTM-GAN-XGBOOST based on the characteristics of artificial neural networks (ANNs) and ensemble learning (EL). The hybrid approach combines long short-term memory network (LSTM) to extract time dimensional features of time series data, generative adversarial networks (GAN) to extract deep features of normal data effectively, and extreme gradient boosting (XGBOOST) to classify the extracted features and export anomaly scores. Moreover, we proposed an anomaly detection framework of test and evaluation based on LSTM-GAN-XGBOOST to obtain the final anomaly results and evaluation indicators. The proposed algorithm shows obvious advantages in processing features extraction and anomaly detection of time series data. Experimental and classic ball bearing time series datasets have been used to testify the effectiveness of the proposed approach and its superiority over some conventional methods. The experimental results demonstrate that LSTM-GAN-XGBOOST can effectively detect the anomalies of ball bearing time series dataset, and achieves 99.1% in terms of area under ROC curve (AUC) which is a superior performance compared with conventional algorithms, and has high significance for time series anomaly detection.**

***Keywords*—*time series data, PHM, LSTM, GAN, XGBOOST, anomaly detection***

## I. Introduction

Prognostic and Health Management (PHM) has been widely applied in many fields, such as machine health monitoring, with great achievements [1]. Anomaly detection which is a significant part of PHM attracts a lot of researchers. For the nonstationary characteristic and randomness of time series data, traditional algorithms for anomaly detection, such as isolation forest and one-class support vector machine, cannot detect deep dimensional anomalies in sensors data, therefore finding better algorithms for anomaly detection of time series data has been a research hotspot in recent years.

Traditional anomaly detection algorithms include isolation forest, local outlier factor, and one-class support vector machine. Isolation forest is an unsupervised anomaly detection method which identifies outliers by creating isolation trees and calculating anomaly scores without using any metrics. It is suitable for continuous numerical data but not appropriate for high-dimensional feature data [2]. Local outlier factor calculates the local density deviation of a given data point relative to its neighborhood for achieving anomaly detection. It is suitable for non-uniform density datasets under unsupervised learning, but data must have obvious density differences and the calculation is very complicated [3]. One-class support vector machine construct the hyperplane model of the positive class data and divide the data on the other side of the hyperplane into an anomalous class to get the anomaly detection result. It is an unsupervised anomaly detection method and suitable for small sample data with high-dimensional features but strongly depends on the selection of the regularization parameters and its kernel function [4]. The conventional algorithms mentioned above can well deal with anomaly detection of data with shallow features, however, there are obvious shortcomings in detecting outliers for time series data [5].

Machine learning shows superiority in anomaly detection because of the deep feature extraction capability and the efficiency in big data processing [6]. The long short-term memory network (LSTM) performs well at processing data in time dimension; generative adversarial network (GAN) obtains excellent performance in deep features extraction because of the characteristics of confrontation training; and extreme gradient boosting (XGBOOST) specializes in classification and regression of large-scale data with high training speed and great effect [7]. Therefore, this paper combines the three structures to LSTM-GAN-XGBOOST algorithm, and makes the experimental verification of anomaly detection on ball bearing time series data. The experimental results demonstrate that the proposed algorithm achieves excellent performance in deep feature extraction and anomaly detection of time series data and shows a good application prospect.

This remainder of this paper is organized as followed. Section II illustrates the theoretical foundation of LSTM, GAN and XGBOOST. Section III gives detailed structure and anomaly detection of LSTM-GAN-XGBOOST algorithm. Simulation results and discussion are shown in Section IV, followed by conclusion in Section V.

2166-5656/20/$31.00 ©2020 IEEE
DOI 10.1109/PHM-Jinan48558.2020.00066

## II. Research Background And Theoretical Basis

### A. Long short-term memory network foundation

The long short-term memory network (LSTM) is a gated recurrent neural networks (RNNs) that effectively solves the problems of long-term dependence and the phenomena of gradient disappearance and gradient explosion of RNNs [8]. With the addition of forget gate, input gate and output gate, LSTM can selectively obtain important information in time series according to data features and ignore irrelevant information to improve the processing ability of time series data [9]. LSTM has been widely used in time series data processing, mainly including text categorization, sentence generation and machine translation [10].

### B. Generative adversarial network foundation

The generative adversarial network (GAN) is constructed on the confrontation training of the generator and the discriminator. The generator generates the generated samples with the features similar to the time series dataset, and the discriminator distinguishes whether the test data has such features or is generated by the generator [11]. With the confrontation training, GAN continuously improves the performances of generator and discriminator, thus the generator can generate samples with features more like training data and discriminator can discriminate the generated samples and training data [12]. Due to its efficient features extraction ability, GAN has been widely used in image generation, image restoration, video generation and so on [13].

### C. Exterme Gradient Boosting foundation

The extreme gradient boosting (XGBOOST) is an optimized distributed gradient boosting library which implements machine learning algorithms under the Gradient Boosting framework and provides a parallel tree boosting (also known as GBDT, GBM) therefore many data science problems can be solved in a fast, accurate, efficient, flexible and portable way [14]. XGBOOST uses Taylor expansion to expand the loss function into a binomial function, and its objective function optimization uses the second derivative of the loss function with respect to the function to be sought. The advantages lie that it can process large-scale data, support multi-thread parallelization, have multiple strategies to prevent overfitting, have fast training speed, and make good effect [15].

## III. LSTM-GAN-XGBOOST Anomaly Detection Algorithm

LSTM-GAN-XGBOOST algorithm is divided into 2 sub-models: the first one is LSTM-GAN trained by normal class dataset, and the second one is Dis1-XGBOOST trained by the features extracted by the discriminator of LSTM-GAN from time series data. The fusion algorithm combining LSTM, GAN, and XGBOOST which can classify normal and abnormal time series data is described below.

### A. LSTM-GAN

LSTM-GAN model is the first sub-model of LSTM-GAN-XGBOOST algorithm. LSTM-GAN have the abilities that can extract the data features in the time dimension by LSTM and can get the deep feature extraction of time series data by GAN. These abilities help construct a generator that can generate generated samples with the same features as normal class data and a discriminator that can extract the features of normal class data and discriminate normal class data.

The generator input is the Gaussian noise and export the generated sample which is similar to normal class data and is the input of the discriminator. The aim of the discriminator is to discriminate the generated sample as false, and the purpose of the generator is to deceive the discriminator to judge the generated samples as true.

The training data of LSTM-GAN is a normal class data set. After the training is completed, the data generated by the generator will have the same features as the normal class training data, and the discriminator will have the excellent ability to extract such features and identify the data is normal or not. The training diagram of LSTM-GAN is shown in Fig. 1.

In Fig. 1, the discriminator of GAN with N layers totally is divided into two parts: Dis1 and Dis2. Dis1 consists of the first layer to the penultimate layer of the discriminator and it can extract 128 features of a time series data sample; Dis2 is the last layer of the discriminator, and it can export the abnormal score according to the relationship among the extracted 128 features. The features extracted by Dis1 will be used to train Dis1-XGBOOST which is the second sub-model of LSTM-GAN-XGBOOST.

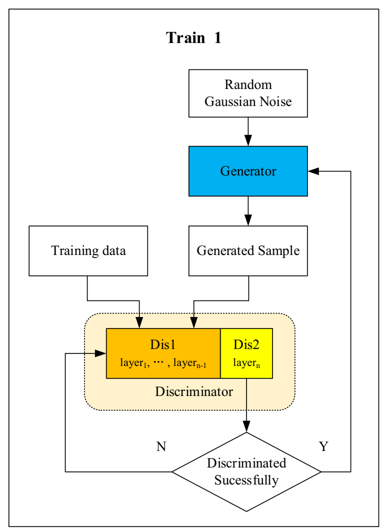


Figure. 1 Training of LSTM-GAN

The hierarchical structures of the generator and discriminator are shown in Fig. 2 and Fig. 3. Three convolutional layers are the main structure of generator. And there are three convolutional layers and one LSTM layer in the discriminator model.

In the hierarchical structure of LSTM-GAN, *LeakyRelu* layers are used to realize learning under the inverse gradient of the neurons, *Conv1D* layers are used to extract multidimensional features of time series, and *LSTM* layer is used to extract the data features in time dimension. *BatchNormalization* layers maintain the inputs of each layer in the network structure the same distribution and speed up the training process, and *Dropout* layers can prevent overfitting.

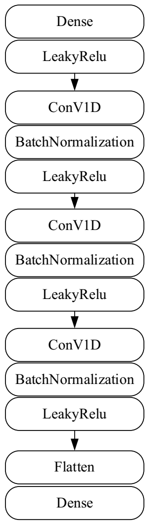


Figure. 2 Structure of generator

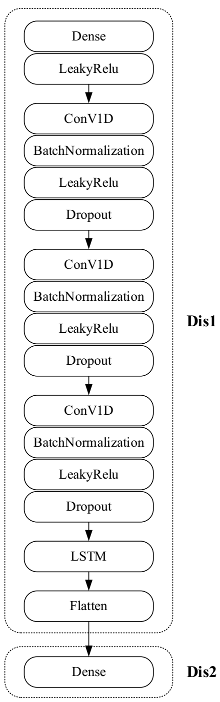


Figure. 3 Structure of discriminator

### *B. Dis1-XGBOOST*

Dis1-XGBOOST is the second sub-model of LSTM-GAN-XGBOOST algorithm. The trained Dis1 of the discriminator in the first sub-module is required in the training process of this sub-model.

Fig. 4 shows the training chart of Dis1-XGBOOST. In the data preprocessing for XGBOOST, normal data and abnormal data each accounts for 50% which can avoid classification imbalance problem, and are imported into Dis1 to obtain feature set 1 of normal data and feature set 2 of abnormal data. Both feature set 1 and feature 2 have 128 feature dimensions. Use 0 for normal class labels as label set 1, and use 1 for abnormal labels as label set 2. And use feature set 1, label set 1, feature set 2, and label set 2 as the training dataset of XGBOOST. Train XGBOOST to achieve the purpose of classification of normal and abnormal.

It should be emphasized that in the process of constructing the XGBOOST, the feature extraction of Dis1 of the discriminator for normal data and abnormal data is a very important step.

Fig. 5 shows the top 40 features in Feature Importance Ranking of the trained XGBOOST on the ball bearing dataset. There are 128 features in total extracted by Dis1 from time series data. As can be seen from Fig. 5, in fact, a time series data sample can be described by 34 deep features whose importance score is greater than 5 points.

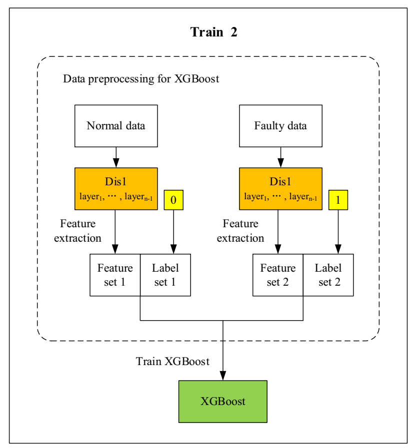


Figure. 4 Training of Dis1-XGBOOST

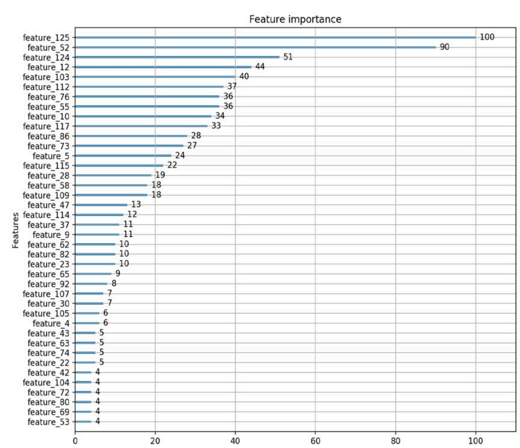


Figure. 5 TOP 40 features in Feature Importance Ranking of XGBOOST

### *C. Test of LSTM-GAN-XGBOOST*

After training LSTM-GAN and Dis1-XGBOOST, the structure of anomaly detection in Fig. 6 for LSTM-GAN-

XGBOOST is proposed to get the detection result which evaluates the abnormal score of time series data.

In the testing phase in Fig. 6, the features of testing data are extracted to be the generator input for data reconstruction. The root mean square error between the reconstructed data and the testing data is used as the reconstruction residual of generator; the absolute error between the abnormal scores of reconstructed data and the testing data is used as the discrimination loss of discriminator. When the test data is normal, the theoretical values of reconstruction residual and discrimination loss are 0; when the test data is abnormal, reconstruction residual is greater than 0 and theoretical discrimination loss is 1. We set a parameter $\beta$ as 0.5 to balance the influence of reconstruction residual and discrimination loss, and to combine them as a comprehensive anomaly detection score. When the anomaly detection score exceeds the threshold $\theta$ which is set as 0.5, the test data will be judged as abnormal class data. Therefore anomaly detection of time series data is realized. The threshold $\theta$ and factor $\beta$ are a set of adjustable parameters that affect each other and anomaly detection result. We set their values as 0.5 according to the general logic and plan to dig the relationship between these two parameters in future work.

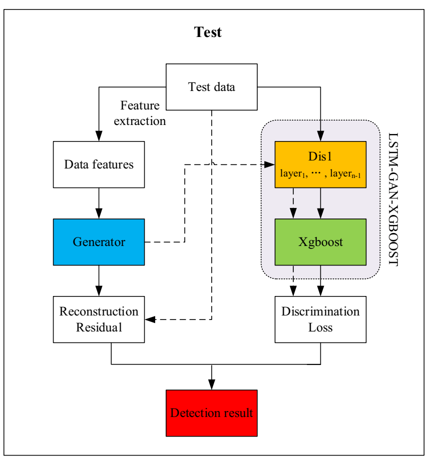


Figure. 6 Anomaly detection framework of test and evaluation based on LSTM-GAN-XGBOOST

### D. Pseudo code of LSTM-GAN-XGBOOST

Anomaly detection of the proposed LSTM-GAN-XGBOOST algorithm is expressed by pseudo code as follows.

In the function TRAIN_GAN which represents the training of the first sub-model LSTM-GAN, *genr* and *disr* represent the generator and discriminator respectively, *uniform(0,1)* indicates the random Gaussian noise with a mean of 0 and a variance of 1, and *traindisr* and *traingenr* represents the training processes of generator and discriminator.

In the function TRAIN_XGB which represents the training of the second sub-model Dis1-XGBOOST, *Dis1* is made up of the first layer to the penultimate layer of the discriminator, *trainxgboost* represents the training of Dis1-XGBOOST, and *LGX* represents the trained LSTM-GAN-XGBOOST.

In the function TEST which represents the calculation of the anomaly detection result of test data, *mse* calculates the root mean square error to get the reconstruction residual, *abs* calculates absolute error to get the discrimination loss, calauc plots the ROC curve of the anomaly detection result.

In the pseudo code, three core parameters of GAN training are the epochs $n$ which influences the performance of generator and discriminator, the integration factor $\beta$ which fuses the reconstruction residual and discrimination loss, and the threshold $\theta$ which determines whether the detection result is normal or abnormal.

**Algorithm 1** LSTM-GAN-XGBOOST for anomaly detection

```
Inputs: traindata_gan, traindata_xgb_1, traindata_xgb_2, testdata, n, β, θ
Outputs: auc
function TRAIN_GAN(traindata_gan, n)
    while i < n do
        z ← uniform(0, 1)
        gendata ← genr(z)
        traindisr(traindata) → 1
        traindisr(gendata) → 0
        while disr(gendata) < 1 do
            traingenr(z) → gendata
        end while
        i ← i + 1
    end while
end function

function TRAIN_XGB(traindata_xgb_1, traindata_xgb_2)
    disr → Dis1
    feature1 ← Dis1(traindata_xgb_1)
    label1 ← 0
    feature2 ← Dis1(traindata_xgb_2)
    label2 ← 1
    LGX ← trainxgboost(feature1, feature2, label1, label2)
end function

function TEST(testdata, β, θ)
    testdata → feature
    data ← genr(feature)
    res ← mse(data, testdata)
    dis ← abs(LGX(testdata) − LGX(data))
    loss ← β ∗ res + (1 − β) ∗ dis
    auc ← calauc(loss)
    if loss > θ then
        testdata → anomaly
    end if
    return auc
end function
```

## IV. Simulation Results And Analysis

The simulation experiments of LSTM-GAN-XGBOOST is carried out as follows:

1) In the training process 1, only normal class data is used as training data, thus the generator and discriminator learn the features of the normal class data.

2) In the training process 2, the output of Dis1 of the discriminator is used as the feature set and the label set is given, therefore LSTM-GAN-XGBOOST has the ability to classify normal and abnormal data.

3) In the testing process, normal data and abnormal data are used as test data, and the normal and abnormal types are detected and distinguished according to LSTM-GAN-XGBOOST.

The experimental simulation is carried out on *Ball Bearing time series data*. This dataset was filtered from of Bearing Data

Center Seeded Fault Test Data of the Bearing Data Laboratory of Case Western Reserve University. The filtered dataset records the acceleration data of the bearing driving end. Each sample has 400 sampling points which is the statistics of acceleration data at the driving end with one rotation of the bearing. The dataset is divided into the normal category with no fault on the bearing and the abnormal category with the inner race fault on the bearing. The filtering method is shown in Tab. I. The filtered data scatter diagram is shown in Fig. 7. The filtered dataset has enough data to support two different training set for LSTM-GAN and XGBOOST, and a test set where abnormal data accounts for 10% to evaluate algorithms.

TABLE I. FILTERING METHOD FOR BALL BEARING DATASEt

| Selected File | Category | Selected Field | Number of data | Fault Diameter |
|---|---|---|---|---|
| 98.mat | Normal | X098_DE_time | 483903 | 0.000" |
| 106.mat | Faulty | X106_DE_time | 121991 | 0.007" |
| 170.mat | Faulty | X170_DE_time | 121846 | 0.014" |
| 210.mat | Faulty | X210_DE_time | 121556 | 0.021" |
| Common parameters | Sampling Rate | 12 KHz | | |
| | Motor Speed | 1772 rpm | | |
| | Fault Position | Inner Race | | |
| | Fault Depth | 0.011" | | |
| | Bearing Manufacturer | SNF (Svenska Kullager-Fabriken) | | |

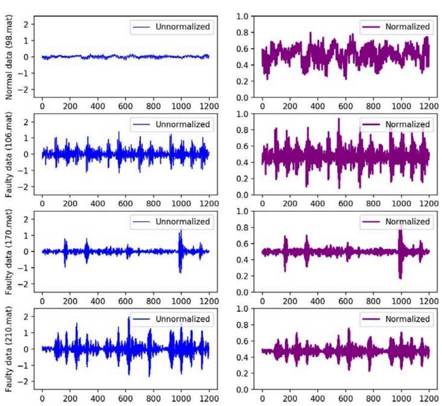


Figure. 7 Scatter diagram of ball bearing dataset

On the ball bearing dataset, the anomaly detection performances of LSTM-GAN-XGBOOST algorithm, isolation forest algorithm (IF), local outlier factor algorithm (LOF), and one-class support vector machine algorithm (OCSVM) are compared. The epochs *n* of LSTM-GAN-XGBOOST is 1000. In the training of XGBOOST, the *number of trees* is set to 160, *maximum depth* is set to 5, *gamma* is set to 0.1 and *learning rate* is set to 0.1. The IF, LOF and OCSVM are all based on the sklearn library in Python. In IF, the parameter *contamination* is set to 0.1, *n_estimators* is set to 35 and *max_features* is set to 27. In LOF, *contamination* is set to 0.1, *n_neighbor*s is set to 35 and *leaf_size* is set to 30. The *kernel* function of OCSVM is set to RBF, *gamma* is set to 0.5 and *nu* is set to 0.15. By traversing all parameter combinations to maximize the AUC value of each algorithm, the above parameter combinations reach the optimal setting.

It should be noted that simulations in this paper are conducted with MacBook Pro 2019 (i7-9750H, 32GB DDR4 Memory), PyCharm 2020.2 (Community Edition) and TensorFlow 1.15.3. The experimental simulation results are shown below in ROC curve and anomaly evaluation indicators.

### *A. ROC curve performance*

Receiver Operating Characteristic (ROC) is a plot of False positive rate (FPR) against True positive rate (TPR) of binary classifiers, which is a comprehensive indicator reflecting the continuous variables relationship between sensitivity and specificity of classifier. FPR is the proportion of negative data points mistakenly predicted positive to all negative data points and TPR is the proportion of positive data points that are correctly predicted positive to all positive data points. ROC curve is a very important and common statistical analysis method and the larger area under ROC curve (AUC) means the higher diagnostic accuracy [16].

Fig. 8 shows the results by ROC curves. According to the simulation results, the AUC value of LSTM-GAN-XGBOOST algorithm is 99.10% which achieves the largest AUC among the four algorithms. The AUC value of LOF is slightly lower than that of LSTM-GAN-XGBOOST, and IF and OCSVM have the general performance on ball bearing dataset.

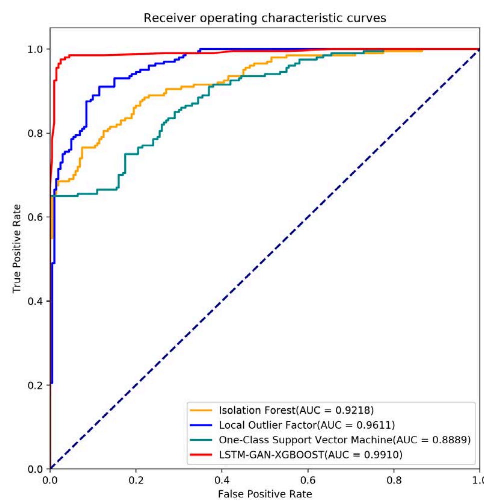


Figure. 8 ROC curves of four algorithms on ball bearing dataset

### *B. Anomaly evaluation indicators performance*

Anomaly evaluation indicators mainly include Precision, Recall, F1-score and Accuracy. Precision is the ratio of the number of positive and actual positive samples to the number of positive samples. Recall is the ratio of the number of positive and actual positive samples to the actual number of positive samples. F1-score is a comprehensive indicator considering Precision and Recall. Accuracy is the ratio of the number of samples with the same test results to the total number of samples. The larger evaluation indicator value means the better detection performance [12], [13].

TABLE II. ANOMALY EVALUATION INDICATORS OF FOUR ALGORITHMS ON BALL BEARING DATASET

| Algorithms | Accuracy | Recall | F1-score | Precision |
|---|---|---|---|---|
| **Isolation Forest** | 0.8275 | 0.8900 | 0.8376 | 0.7911 |
| **Local Outlier Factor** | 0.8975 | 0.9100 | 0.8988 | 0.8878 |
| **One-Class SVM** | 0.7950 | 0.6500 | 0.7602 | 0.9155 |
| **LSTM-GAN-XGBOOST** | **0.9725** | **0.9800** | **0.9727** | **0.9655** |

Tab. II shows the results of anomaly evaluation indicators. The experimental results on ball bearing dataset show LSTM-GAN-XGBOOST algorithm performs best in terms of the Accuracy, Recall, F1-score and Precision compared with conventional algorithms. In terms of Accuracy and F1-score, LOF performs better than IF and OCSVM; in terms of Recall, the performance of OCSVM is much behind; in terms of Precision, LOF and OCSVM work better than IF. In conclusion, LSTM-GAN-XGBOOST achieve best performance, LOF and IF perform mediocre, and OCSVM have the worst evaluation indicators.

## V. CONCLUSION

In this paper, we proposed a novel LSTM-GAN-XGBOOST algorithm for time series anomaly detection. Firstly, we used LSTM as a part of generator of GAN to learn the timing relationships of normal data. Secondly, we used LSTM-GAN to train the discriminator, therefore Dis1 of discriminator can effectively extract 128 deep features of time series data. Thirdly, we trained XGBOOST with the features extracted by Dis1 and constructed Dis1-XGBOOST to classify such features. Finally, we proposed a Dis1-XGBOOST-based detection framework which combined the reconstruction residual and discrimination loss as an anomaly detection result. The algorithm was verified by ball bearing dataset, and by comparing the detection results of LSTM-GAN-XGBOOST and traditional methods, it is verified that LSTM-GAN-XGBOOST algorithm can more effectively extract the deep features of time series data and significantly improve the accuracy of anomaly detection. Compared with traditional methods, evaluation indicators as Accuracy, Recall, F1-score, Precision and AUC of LSTM-GAN-XGBOOST algorithm all achieve the best performance. Moreover the test framework of LSTM-GAN-XGBOOST algorithm is suitable for time series data processing, however, the integration factor $\beta$ and the threshold $\theta$ in the proposed algorithm still should be adjusted in terms of specific dataset. How to select appropriate values for these parameters scientifically is our future work.

## REFERENCES


[1] L. Zhang, J. Lin, B. Liu, Z. Zhang, X. Yan and M. Wei, "A Review on Deep Learning Applications in Prognostics and Health Management," in IEEE Access, vol. 7, pp. 162415-162438, 2019.

[2] S. Li, K. Zhang, P. Duan and X. Kang, "Hyperspectral Anomaly Detection With Kernel Isolation Forest," in IEEE Transactions on Geoscience and Remote Sensing, vol. 58, no. 1, pp. 319-329, Jan. 2020.

[3] Z. Xu, D. Kakde and A. Chaudhuri, "Automatic Hyperparameter Tuning Method for Local Outlier Factor, with Applications to Anomaly Detection," 2019 IEEE International Conference on Big Data (Big Data), Los Angeles, CA, USA, 2019, pp. 4201-4207.

[4] F. Borges et al., "An Unsupervised Method based on Support Vector Machines and Higher-Order Statistics for Mechanical Faults Detection," in IEEE Latin America Transactions, vol. 18, no. 06, pp. 1093-1101, Jun 2020.

[5] Z. Zhao, Y. Zhang, X. Zhu and J. Zuo, "Research on Time Series Anomaly Detection Algorithm and Application," 2019 IEEE 4th Advanced Information Technology, Electronic and Automation Control Conference (IAEAC), Chengdu, China, 2019, pp. 16-20.

[6] M. Hu et al., "Detecting Anomalies in Time Series Data via a Meta-Feature Based Approach," in IEEE Access, vol. 6, pp. 27760-27776, 2018.

[7] Q. Du, F. Yin and Z. Li, "Base station traffic prediction using XGBoost-LSTM with feature enhancement," in IET Networks, vol. 9, no. 1, pp. 29-37, 1 2020.

[8] K. Lee, J. Kim, J. Kim, K. Hur and H. Kim, "Stacked Convolutional Bidirectional LSTM Recurrent Neural Network for Bearing Anomaly Detection in Rotating Machinery Diagnostics," 2018 1st IEEE International Conference on Knowledge Innovation and Invention (ICKII), Jeju, 2018, pp. 98-101.

[9] H. Shi, L. Guo, S. Tan and X. Bai, "Rolling Bearing Initial Fault Detection Using Long Short-Term Memory Recurrent Network," in IEEE Access, vol. 7, pp. 171559-171569, 2019.

[10] D. Park, Y. Hoshi and C. C. Kemp, "A Multimodal Anomaly Detector for Robot-Assisted Feeding Using an LSTM-Based Variational Autoencoder," in IEEE Robotics and Automation Letters, vol. 3, no. 3, pp. 1544-1551, July 2018.

[11] Y. Choi, H. Lim, H. Choi and I. Kim, "GAN-Based Anomaly Detection and Localization of Multivariate Time Series Data for Power Plant," 2020 IEEE International Conference on Big Data and Smart Computing (BigComp), Busan, Korea (South), 2020, pp. 71-74.

[12] G. Zhu, H. Zhao, H. Liu and H. Sun, "A Novel LSTM-GAN Algorithm for Time Series Anomaly Detection," 2019 Prognostics and System Health Management Conference (PHM-Qingdao), Qingdao, China, 2019, pp. 1-6.

[13] T. Ganokratanaa, S. Aramvith and N. Sebe, "Unsupervised Anomaly Detection and Localization Based on Deep Spatiotemporal Translation Network," in IEEE Access, vol. 8, pp. 50312-50329, 2020.

[14] J. Pardede, B. Sitohang, S. Akbar and M. L. Khodra, "Improving the Performance of CBIR Using XGBoost Classifier with Deep CNN-Based Feature Extraction," 2019 International Conference on Data and Software Engineering (ICoDSE), Pontianak, Indonesia, 2019, pp. 1-6.

[15] D. Zhang, L. Qian, B. Mao, C. Huang, B. Huang and Y. Si, "A Data-Driven Design for Fault Detection of Wind Turbines Using Random Forests and XGboost," in IEEE Access, vol. 6, pp. 21020-21031, 2018.

[16] S. Naseer et al., "Enhanced Network Anomaly Detection Based on Deep Neural Networks," in IEEE Access, vol. 6, pp. 48231-48246, 2018.