A Novel LSTM-GAN Algorithm for Time Series Anomaly Detection

Guangxuan Zhu, Hongbo Zhao[1,2], Haoqiang Liu, Hua Sun
1.Beihang University
2.Hefei Innovation Research Institute, Beihang University
Beijing Haidian, 100191, China
zhugx2013@buaa.edu.cn

Abstract—**Time series anomaly detection is an important part of Prognostic and Health Management (PHM), and has been widely studied and followed with interest. The data with time series features often has non-stationary properties, and its fluctuation amplitude changes with time. Traditional anomaly detection algorithms can achieve the detection of shallow level anomalies when facing such data, however they fail to detect outliers on deep features of time series data. The gate structure of the long short-term memory network (LSTM) shows obvious advantages in processing time series data, while the confrontation training of generative adversarial network (GAN) performs well in detecting and acquiring deep features of data. Therefore, this paper focuses on the anomaly detection of time series data with the fusion model of LSTM and GANs, which is named the LSTM-GAN, and the performance of the algorithm is verified from two sets of time series data. The experimental results demonstrated that the proposed algorithm achieved superior performance in processing time series data compared to conventional algorithms. The research content of this article has profound guiding significance for time series anomaly detection.**

Keywords-time series data, PHM, LSTM, GAN, anomaly detection

I. INTRODUCTION

Prognostic and Health Management (PHM) has been widely applied in many fields, and has made great achievements in system health management and performance monitoring [1]. Due to the wide application of sensor networks and the continuous improvement of data processing technology, anomaly detection of time series data has rose interest all over the world. For the non-stationary properties and randomness of time series data, traditional algorithms for anomaly detection, such as isolation forest and one-class support vector machine, cannot detect deep anomalies in system sensor data, therefore exploring better algorithms for time series data anomaly detection has become a research hotspot in recent years.

Anomaly detection has a wide range of applications in system performance monitoring and health management. Traditional anomaly detection techniques mainly include isolation forest algorithm, local outlier factor detection algorithm, one-class support vector machine algorithm and statistical model. Isolation forest algorithm, which distinguishes outliers by establishing isolation trees and calculating anomaly scores, can effectively distinguish and detect isolated outliers, but it is not applicable, particularly for high-dimensional data [2]. Local outlier factor algorithm can detect abnormal data deviating from mass data based on density, however the selection of the nearest neighbor number and the heavy computation complexity are still waiting to be solved [3]. One-class support vector machine algorithm can obtain the ideal anomaly detection result by constructing the hyperplane model of the positive class data and dividing the data on the other side of the hyperplane into an anomalous class, but the result strongly depends on the selection of the regularization parameter and the kernel function [4]. The statistical model is based on the statistical analysis of data. And the statistical features of data are utilized to construct a model, in which the data points beyond the normal range are assigned to anomaly. However, the obvious deficiency of this algorithm lies in that data must be subordinate to the statistical model [5]. Conventional algorithms have the positive effect on the implementation of shallow anomaly detection, unfortunately, there are still obvious deficiencies in the realization of time series data anomaly detection.

The deep feature extraction capability and the efficient massive data processing of machine learning algorithms make them stand out in the detection of time series anomalies [6]. The long short-term memory network (LSTM) shows superiority in time-series data processing, while the characteristics of confrontation training of generative adversarial network (GAN) can obtain excellent performance in extracting deep features of data. Therefore, this paper combines the two network structures to LSTM-GAN algorithm, and conducts experimental verification of the anomaly detection performance on two sets of time series data. The experimental results demonstrated that the proposed algorithm achieves excellent performance in processing time series data and shows a good prospect of the application.

This paper is organized as followed. Section I mainly introduces relevant anomaly detection algorithms. The theoretical basis of LSTM and GAN are illustrated in Section II. Section III gives detailed structure and anomaly detection process of LSTM-GAN algorithm. Simulation results and discussion are shown with two time series sets in Section IV, followed by conclusion in Section V.

978-1-7281-0861-2/19/$31.00 ©2019 IEEE

II. Research Background And Theoretical Basis

A. Long short-term memory network basis

The long short-term memory network is derived from the recurrent neural network, and the gradient disappearance and gradient explosion phenomena of the recurrent neural network are improved. Due to the forgotten gates, input gates and output gates that it is added, the long short-term memory network can enhance the memory of important information by selectively forgetting irrelevant information according to appropriate parameters [7]. Therefore, the long short-term memory network can selectively acquire important information in the time series according to the data features while ignoring the insignificant information, thereby improves the processing capability of the time series data. Long-short term memory network obtains a wide range of applications in the processing of time series data, which mainly include the fields of text categorization [8], statement generation and machine translation [9].

B. Generative adversarial network basis

The generative adversarial network is constructed based on the confrontation training of the generator and the discriminator, in which the generator extracts the data features and the discriminator distinguishes the anomalous data [10]. Because of the confrontation training idea, generative adversarial network continuously improves the performances of generator and discriminator according to the learned features, enhances the generator's ability to generate real data and discriminator's ability to discriminate the generated data and real data, and finally achieves the extraction of data features and the construction of anomaly detection models. Due to its efficient data features extraction ability, generative adversarial network has been widely used in image generation, image restoration [11], video generation and so on.

III. LSTM-GAN Anomaly Detection Algorithm

LSTM has significant advantages in processing time series data, while GAN has excellent capabilities in extracting data features and building data models. Therefore, this paper attempts to combine LSTM and GAN, extracts the time series characteristics of time series data by using LSTM, and extracts the deep feature furtherly and constructs normal class data model by using GAN. When the model output of the testing data exceeds the threshold of the normal data model, it is judged as an anomalous class, thereby realize anomaly detection on time series data.

Compared with the GAN model for processing two-dimensional images, the main improvements of the LSTM-GAN anomaly detection algorithm are as follows:

1) Introducing one-dimensional convolution layer into the GAN to implement feature extraction and reconstruction of the time series data.

2) Introducing the LSTM layer into the GAN to achieve the extraction of data temporal characteristics.

3) The generate residual of the generator and the discriminate loss of the discriminator are integrated as an evaluation criterion of the anomaly detection score.

The structure of the LSTM-GAN anomaly detection algorithm is displayed in Fig. 1.

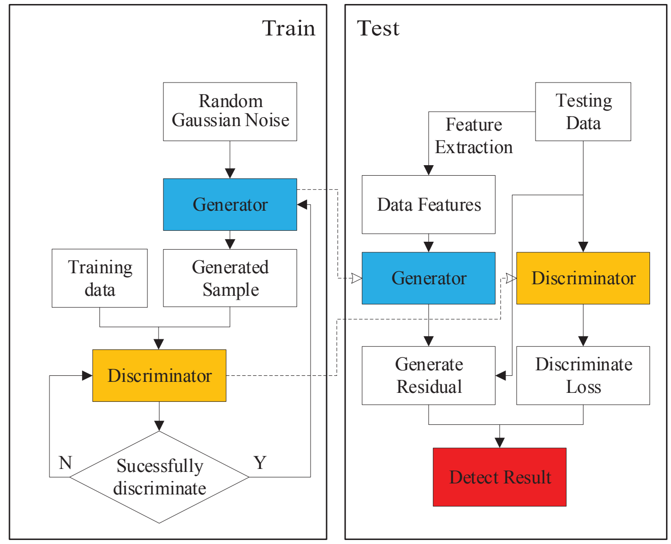

Figure. 1 Structure of LSTM-GAN anomaly detection algorithm

Among them, the combination of LSTM and convolutional layers constitutes the structure of generator and discriminator of the LSTM-GAN algorithm. Specifically, the hierarchical structures of the generator and discriminator are shown in Fig. 2 and Fig. 3.

In the structure of the LSTM-GAN algorithm, the *LeakyRelu* layer is aimed at learning the inverse gradient of the neurons, the *Conv1D* and *LSTM* layers are used to extract the data temporal features, and the *BatchNormalization* layer keeps the input of each layer of the network structure the same distribution, *Dropout* layer helps prevent the overfitting of the trained model.

The structure of the generator is the same as the discriminator in most layers, while the discriminator adopts *LSTM* layer to maintain the temporal feature of time series data and *Dropout* layer to prevent the overfitting.

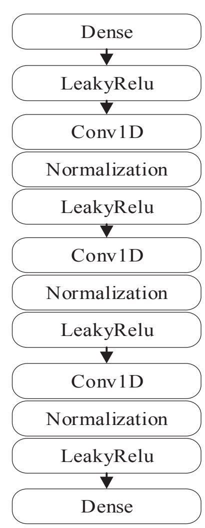

Figure. 2 Structure of generator

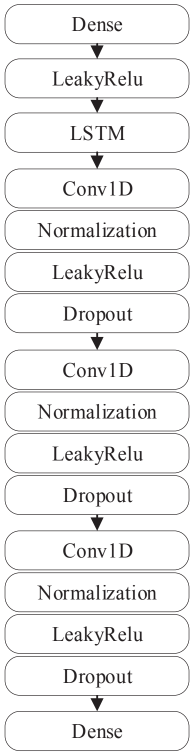

Figure. 3 Structure of discriminator

The generator and discriminator are trained first and then applied to detect anomaly. The anomaly detection process of the proposed LSTM-GAN algorithm is consisted of train and test phases, and the detail of the algorithm is expressed by pseudo-code as follows.

In the pseudo-code, *genr* and *disr* represent the generator and discriminator respectively, *uniform(0,1)* indicates the generation of the random gaussian noise with a mean of 0 and a variance of 1, *traindisr* and *traingenr* represents the parameter training process to promote the performance of discriminator and generator, the *mse* function calculates the root mean square error to get the generate residual, the *abs* function calculates the absolute value of the discrimination result to obtain the discrimination loss, and the *calauc* function plots the ROC curve of the detection result.

The parameters involved in pseudo-code mainly include the training epochs *n*, the error integration factor β, and the abnormal threshold θ .The training epochs *n* determines the pros and cons of the generator and discriminator training results, which is determined according to the actual characteristics of the training data set. The error integration factor β affects the determination of the anomaly detection score, and is selected according to the model parameters of the generator and the discriminator. The abnormal threshold θ determines the performance of the abnormal detection, and is set according to the empirical value.

In the training phase, the normal class data and the generated samples of the random gaussian noise are firstly used as the input of the fusion model discriminator, so that the discriminator realizes the distinction between the normal data and the generated samples, and then fixes the discriminator parameters and adjusts the generator parameters to make the generator's generated sample spoof discriminator and be judged as a normal class.

In the testing phase, the testing data feature is firstly extracted to obtain the feature sequence, and is also used as a generator input for data reconstruction. The root mean square error between the reconstructed data and the testing data is used as the generator's generate residual, and then the absolute value of difference between the normal output 1 and the testing data in the discriminator output is calculated as the discriminator's discriminate loss, and finally the generate residual and discriminate loss are combined as an anomaly detection score result. When the score result exceeds the threshold θ , the testing data is detected as anomalous.

Algorithm 1 LSTM-GAN for anomaly detection

```
Inputs: traindata,testdata,n, β, θ
Outputs: auc
function TRAIN(traindata, 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 TEST(testdata,β, θ)
    testdata → feature
    data← genr(feature)
    res← mse(data,testdata)
    dis← abs(disr(testdata)-1)
    loss← res∗β+dis∗(1 − β)
    auc ← calauc(res,dis)
    if loss> θ then
        testdata→ anomaly
    end if
    return auc
end function
```

When the testing data is normal, the theoretical value of the generate residual is 0, and the theoretical value of the discrimination loss is 0. When the testing data is an anomalous class, the generate residual is greater than 0, and the theoretical value of the discrimination loss is close to 1. Therefore, the reasonable threshold can be set according to the integration error to realize the distinction between the normal class and the anomalous class. The anomaly detection performance of the algorithm is verified from two sets of time series data in the following.

IV. SIMULATION RESULTS AND ANALYSIS

The simulation experiment of the LSTM-GAN algorithm is carried out as follows: In the training phase, only the normal class data is used as the training data, so that the generator and the discriminator learn the model features of the normal class data. In the testing phase, normal class data and anomalous class data are combined as the testing data, and the normal class and the anomalous class data are distinguished according to the trained model.

According to the network structure of the model, the input training data and the input testing data need to be normalized before being transmitted into the model. The data values are normalized to the interval of [-1, 1] to facilitate the parameter transmission and update of the network layer.

The experimental simulation is carried out on the following two sets of data:

1) *ECG time series data:* each data in the data set records the electrical activity during a heartbeat, with 96 sampling points on each data, that is 96 data values. The data is divided into normal heartbeat and myocardial infarction anomaly. The proportion of anomalous data is 36%.

2) *nyc_taxi time series data:* the data set is the statistics of taxi traffic in New York City in the second half of 2014. The data value is the passenger flow every half hour. There are 48 sampling points on each data, that is, the passenger flow within one day. The data is divided into normal classes and anomalous classes, the latter appears in the marathon, Thanksgiving, Christmas, New Year and snowstorm periods. The proportion of anomalous data is 8.3%.

It should be noted that simulations in this paper are conducted with Lenovo R720 (i7-7700HQ, 8GB DDR3 Memory) and PyCharm 2018.1. The experimental simulation results were presented in two forms, ROC curve and abnormal evaluation index, as shown below.

A. ROC curve performance

The ROC curve refers to the receiver operating characteristic curve, which is a comprehensive indicator reflecting the continuous variables relationship of sensitivity and specificity. And the composition method reveals the relationship between sensitivity and specificity. ROC curve is plotted by calculating a series of sensitivity and specific values based on setting the continuous variable to a number of different critical values, with the sensitivity as the ordinate and the specificity as the abscissa. The larger the area under curve (AUC), the higher the diagnostic accuracy [12].

On the ECG dataset, the anomaly detection performances of LSTM-GAN algorithm and isolation forest algorithm, local outlier factor algorithm, one-class support vector machine algorithm, and gaussian statistical model are compared. The epochs n of the LSTM-GAN algorithm is 1000, and the β is set to 1. The isolation forest, local outlier factor, one-class support vector machine and gaussian statistical model algorithms are all implemented in the sklearn library in Python. The parameter constraints of isolation forest, local outlier factor and gaussian statistical model are set to 0.01, the kernel function of one-class support vector machine is KBF, and the parameter γ is set to 0.1.

The ROC curves drawn based on the test results are shown in Fig. 4. According to the simulation results, the LSTM-GAN algorithm achieves the highest AUC value among the five algorithms, which is the best detection accuracy, followed by the isolation forest, local outlier factor and gaussian statistical model algorithms, their detection accuracies are slightly lower than the LSTM-GAN algorithm, the one-class support vector machine algorithm obtains the poorest performance on this data set.

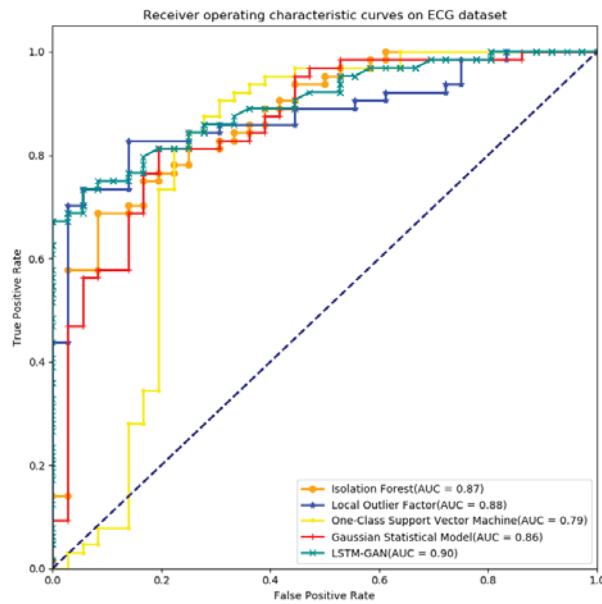

Figure. 4 ROC curves of five algorithms on ECG dataset

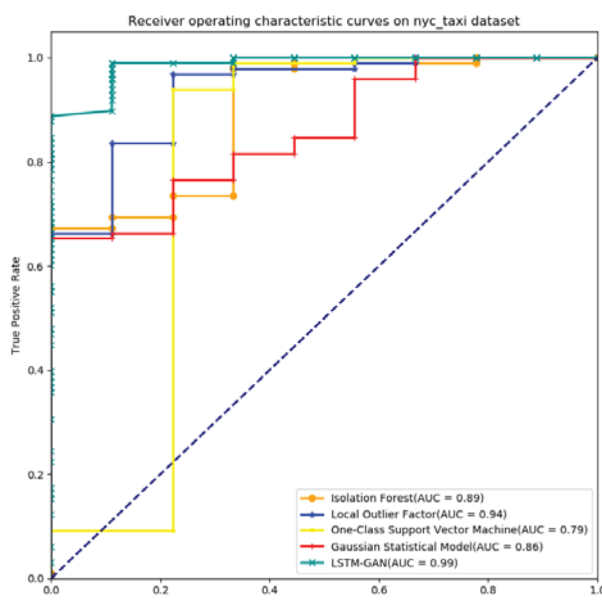

Figure. 5 ROC curves of five algorithms on nyc_taxi dataset

On the nyc_taxi dataset, the anomaly detection performances of LSTM-GAN algorithm and isolation forest algorithm, local outlier factor algorithm, one-class support vector machine algorithm, and gaussian statistical model are compared. The epochs n of the LSTM-GAN algorithm is 1000, and the β is set to 0.8. The isolation forest, local outlier factor, one-class support vector machine and gaussian statistical model algorithms are all implemented in the sklearn library in Python. The parameter constraints of isolation forest, local outlier factor and gaussian statistical model are set to 0.01, the kernel function of one-class support vector machine is KBF, and the parameter γ is set to 0.1.

The ROC curves drawn based on the test results are shown in Fig. 5. According to the simulation results, the LSTM-GAN algorithm achieves the highest AUC value among the five algorithms, the detection accuracies of the local outlier factor and gaussian statistical model algorithms are slightly lower than the LSTM-GAN algorithm, and the isolation forest and one-class support vector machine algorithms obtains the poorest performance on this data set.

B. Anomaly evaluation index performance

Anomaly evaluation indicators mainly include precision rate, recall rate, F1-measure value and accuracy. The precision rate indicates the ratio of the number of samples detected as positive and actually positive to the number of samples that are positively detected. The recall rate indicates the ratio of the number of samples detected as positive and actually positive to the number of samples that are actually positive. F1-measure is an indicator that comprehensively considers the precision rate and the recall rate. The accuracy rate indicates the ratio of the number of samples whose test results are the same as the actual type to the total number of samples. The larger the indicator values, the better the detection performance. Among them, the positive class is the target class of the anomaly detection process, that is, the anomalous class, and the negative class is the normal class.

TABLE I. Anomaly evaluation indicators of algorithms on ECG dataset

Algorithms	Precision	Recall	F1	Accuracy
Isolation Forest	1.0000	0.1111	0.2000	0.7200
Local Outlier Factor	0.8889	0.1127	0.2000	0.7100
One-Class SVM	0.7222	0.3250	0.4483	0.8000
Gaussian Statistical Model	0.3600	1.0000	0.5294	0.3600
LSTM-GAN	0.7429	0.3210	0.4483	0.8100

On the ECG dataset, the anomaly detection performances of LSTM-GAN algorithm and isolation forest algorithm, local outlier factor algorithm, one-class support vector machine algorithm, and gaussian statistical model are compared. The epochs n of the LSTM-GAN algorithm is 1000, and the β is set to 1, the threshold θ is set to 0.215. The isolation forest, local outlier factor, one-class support vector machine and gaussian statistical model algorithms are all implemented in the sklearn library in Python. The parameter constraints of isolation forest, local outlier factor and gaussian statistical model are set to 0.01, the kernel function of one-class support vector machine is KBF, and the parameter γ is set to 0.1.

The anomaly evaluation indicators of the test results are shown in Tab. I. The experimental results show that LSTM-GAN outperforms other algorithms in accuracy, and isolation forest algorithm performs best in precision. On the recall rate and F1-measure, gaussian statistical model algorithm shows superiority.

On the nyc_taxi dataset, the anomaly detection performances of LSTM-GAN algorithm and isolation forest algorithm, local outlier factor algorithm, one-class support vector machine algorithm, and gaussian statistical model are compared. The epochs n of the LSTM-GAN algorithm is 1000, and the β is set to 0.8, the threshold θ is set to 0.25. The isolation forest, local outlier factor, one-class support vector machine and gaussian statistical model algorithms are all implemented in the sklearn library in Python. The parameter constraints of isolation forest, local outlier factor and gaussian statistical model are set to 0.01, the kernel function of one-class support vector machine is KBF, and the parameter γ is set to 0.1.

TABLE II. Anomaly evaluation indicators of algorithms on nyc_taxi dataset

Algorithms	Precision	Recall	F1	Accuracy
Isolation Forest	0.6667	0.0400	0.0755	0.9346
Local Outlier Factor	0.5385	0.0707	0.1250	0.9252
One-Class SVM	0.1489	0.1077	0.1250	0.6075
Gaussian Statistical Model	0.5000	0.0306	0.0577	0.9159
LSTM-GAN	0.8571	0.0583	0.1091	0.9626

The anomaly evaluation indicators of the test results are shown in Tab. II. The experimental results show that LSTM-GAN outperforms other algorithms in terms of accuracy and precision. On the recall rate, one-class support vector machine algorithm performs best. Local outlier factor and one-class support vector machine algorithms has advantage over other algorithms on F1-measure.

V. CONCLUSION

Conventional algorithms have the positive effect on the implementation of shallow anomaly detection, unfortunately, there are still obvious deficiencies in the realization of time series data anomaly detection. The proposed LSTM-GAN algorithm, which combines the advantage of LSTM in processing time series data and the advantage of GAN in extracting data depth features, can achieve superior performance compared to traditional anomaly detection methods. The detection framework of the LSTM-GAN algorithm is especially suitable for time series data processing, however, the error integration factor β, and the abnormal threshold θ of the proposed algorithm still need to be tuned according to specific data sets. How to scientifically select appropriate values for these parameters is the next research direction.

REFERENCE

[1] C. L. Liu, W. H. Hsaio and Y. C. Tu, “Time Series Classification with Multivariate Convolutional Neural Network,” *IEEE Transactions on Industrial Electronics*, vol. 66, no. 6, pp. 4788-4797, Jan, 2019.

[2] F. T. Liu, K. M. Ting and Z. H. Zhou, “Isolation Forest,” *2008 Eighth IEEE International Conference on Data Mining*, pp. 413-422, 2008.

[3] H. H. Ma, Y. Hu, and H. B. Shi, “Fault Detection and Identification Based on the Neighborhood Standardized Local Outlier Factor Method,” *Industrial & Engineering Chemistry Research*, vol. 52, no. 6, pp. 2398-2402, Feb, 2013.

[4] S. Mahadevan, S. L. Shah, “Fault Detection and Diagnosis in Process Data using One-Class Support Vector Machines,” *Journal of Process Control*, vol. 19, no. 10, pp. 1627-1639, Dec. 2009.

[5] S. Fortunati, F. Gini and M. S. Greco, et al, “An Improvement of the State-of-the-art Covariance-Based Methods for Statistical Anomaly Detection Algorithms,” *Signal Image and Video Processing*, vol. 10, no. 4, pp. 687-694, Apr. 2016.

[6] R. Habeeb, F. Nasaruddin, and A. Gani, et al, “Real-time Big Data Processing for Anomaly Detection: A Survey,” *International Journal of Information Management*, vol. 45, pp. 289-307, Apr. 2019.

[7] K. Greff, R. K. Srivastava, and J. Koutnik, et al, “LSTM: A Search Space Odyssey,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 28, no. 10, pp. 2222-2232, Oct, 2017.

[8] Y. H. Gu, M. Gu, and Y. Long, et al, “An Enhanced Short Text Categorization Model with Deep Abundant Representation,” *World Wide Web-Internet and Web Information Systems*, vol. 21, no. 6, pp. 1705-1719.

[9] L. H. Baniata, S. Park, and S. B. Park, “A Multitask-Based Neural Machine Translation Model with Part-of Speech Tags Integration for Arabic Dialects,” *Applied Sciences-Basel*, vol. 8, no. 12, pp. 1-18, Dec, 2018.

[10] I. Goodfellow, J. P. Abadie, and M. Mirza, et al, “Generative Adversarial Nets,” *Advances in Neural Information Processing System*, pp. 2672-2680, 2014.

[11] Y. Tao, J. P. Muller, “Super-Resolution Restoration of MISR Images Using the UCL MAGiGAN System,” *Remote Sensing*, vol. 11, no. 1, pp. 52-84, Dec, 2018.

[12] T. Fawcett, “an Introduction to ROC Analysis,” *Pattern Recognition Letters*, vol. 27, no. 8, pp. 861-874, Jun. 2006.