



Tube Inner Circumference State Classification Optimization by Using Artificial Neural Networks, Random Forest and Support Vector Machines Algorithms

Wei-Ting Li, Chung-Wen Hung, and Ching-Ju Chen^(✉)

National Yunlin University of Science and Technology, Douliou, Taiwan
chen.chingju@gmail.com

Abstract. Using Artificial Neural Networks, Random Forest and Support Vector Machines algorithms to optimize Tube inner circumference state classification and accomplish the process of Incoming Quality Control (IQC) is proposed in this paper. However, the traditional classification system is usually set the threshold by the developer in the early stages. The method is time-consuming and tedious to develop the module. In modern, machine learning technology can overcome the shortcomings of tradition classification system. However, machine learning exists a lot of algorithms, such as Artificial Neural Network (ANN), Random Forest (RF), Support Vector Machine (SVM), and so on. And, the different algorithms may cause the different characteristics and efficiencies, so it's necessary to compare the different algorithms at application. This paper will use a method, called grid search to find the best parameter, and compare these algorithms which has the best characteristic, efficiency and the parameter. Finally, it is found from the experimental results that the method of this paper is workable for actual dataset.

Keywords: Machine learning · ANN · RF · SVM and tube inner circumference

1 Introduction

In the past, the accuracy and development time of the traditional classification system is worse than machine learning. As the technology advances, the operational performance of the hardware was already satisfied machine learning with the complicated algorithms now. However, machine learning includes a lot of algorithms, the chopsticks tube data [1] will be used to compare accuracy and speed for ANN, RF, and SVM which are optimized. The contour perimeter, circle radius of circle fitting, two radii of ellipse fitting, maximum inscribed circle, and minimum inscribed circle are used by the developer to classify as “Normal”, “Large”, “Small”, “Deformed” chopstick tube and “Empty” [1]. The accuracy of the result is about 91%. You must put more effort and time to try if you want to improve the accuracy. This paper will use the same training data and the test data to compare accuracy and calculating the speed for ANN, RF, and SVM which are optimized model, and how to find these parameters. In the results, the

accuracy of the test set is higher than the traditional method after the paper uses a grid search [2] to find the optimization parameters of these algorithms. Nevertheless, SVM is the most time-consuming in this paper, then ANN is the slowest at module calculating speed, and RF is the easiest to observe the structure of the algorithm.

2 Methodologies

2.1 Artificial Neural Network, ANN

This structure simulates the characteristics of nervous [3], and the nerves are connected to each other. Figure 1 shows the basic structure of ANN, which includes the input layer, hidden layers and the output layer, and the number of neurons and layer is mainly according to application design. If the module can't satisfy the complicated application, then increase the number of neurons or hidden layer, but the time of training will relatively increase. In the modern, graphics processing unit (GPU) can be used to accelerate parallel computing the complicated module.

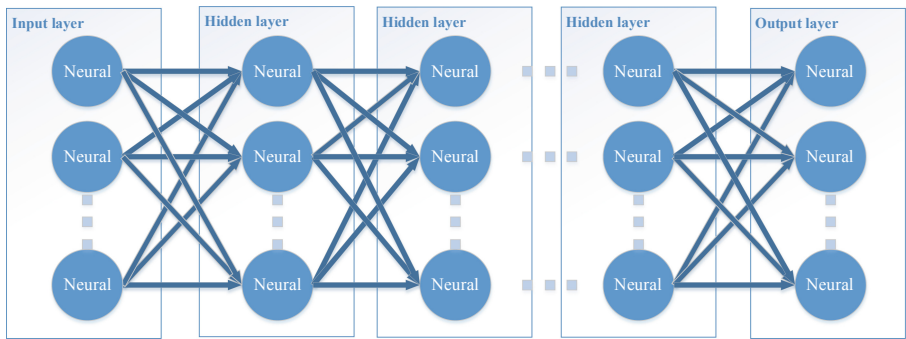


Fig. 1. Artificial Neural Network system architecture

Furthermore, ANN will use the active function [3], this paper used the sigmoid function. However, the value of each layer needs to pass the sigmoid function, and complete the prediction. This paper used cross-entropy [4] and backpropagation [5] to calculate cost and correct weight during training. The structure of ANN is simple, but we cannot understand the meaning for weight.

2.2 Decision Tree, DT

This algorithm is a tree structure on the application for classification or regression. In the process of the training module, it's mainly used to find the maximum information gain (IG) (1) of every parent node. D_p and D_j are the datasets of the parent node and the child node. The parameter f is the feature to perform the split. N_p is all of the samples at the parent node, N_n is the number of samples in the child node. The parameter I is our impurity measure. Then the parameter I will change with different Algorithm.

The algorithm included Iterative Dichotomiser 3 (ID3), C4.5, C5.0 [6], classification and regression trees (CART). The CART is used in the paper, and this algorithm is a binary method that likes the algorithm of ID3. The impurity measure of CART use Entropy (2) with ID3, parameter p is the proportion of c sample in nodes, but the algorithm of CART changes the impurity measure method to Gini index (3) method in building DT.

$$IG(D_p, f) = I(D_p) - \sum_{j=1}^m \frac{N_j}{N_p} I(D_j) \tag{1}$$

$$I_H(t) = - \sum_{i=1}^c [p(i|t) \log_2 p(i|t)] \tag{2}$$

$$I_G(t) = 1 - \sum_{i=1}^c p(i|t)^2 \tag{3}$$

2.3 Random Forest, RF

This algorithm is the ensemble method of DT, the concept of algorithms mainly uses a lot of DT method to achieve majority voting rules, the performance of RF is better than a DT. The method of Bagging is used in random sampling to train the module, also this method used in this paper. As a result, the every DT can keep accurate results for a part of samples, and improve the problem of overfitting. You do not need to prune the RF if you use the RF method.

2.4 Support Vector Machine, SVM

The SVM [8, 9] is mainly to find the hyperplane (4) that can be divided into two categories such as Fig. 2. The \bar{x} is the feature vector of input, $\bar{\omega}$ is the weight vector, b is the bias of hyperplane, pink line is the hyperplane that classify different sample into red dots and blue forks, red dots and blue forks are support vectors (SV); moreover, the hyperplane has to possess maximum distance (5) between hyperplane and SV.

$$h(\bar{x}) = \bar{\omega}^T \bar{x} + b \tag{4}$$

$$\bar{r} = \frac{h(\bar{x})}{\|\bar{\omega}\|} \tag{5}$$

If the application can't use linear SVM to classify, we can use the kernel function to increase the dimensions. The kernel function included radial basis function kernel (RBF), polynomial kernel, sigmoid kernel, inter kernel and so on. For example, Fig. 3 uses the polynomial kernel to increase two-degrees to three-degrees. The hyperplane can be found in high dimensions by this method, and the best kernel and the related parameters will be found by grid search in this paper.

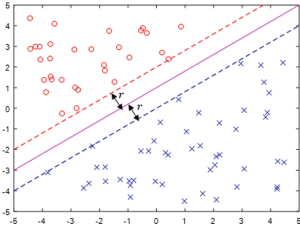


Fig. 2. Schematic that SVM classify the sample (Color figure online)

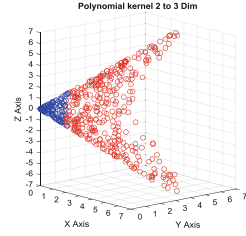
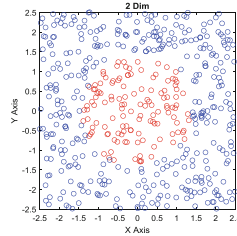


Fig. 3. Use the polynomial kernel to improve the dimensions

3 Experimental Result

In this section, this paper will use tube inner circumference state [1] as a sample to find the optimized algorithm which ANN, RF, and SVM, and improve the traditional method. The tube inner circumference state and feature [1] are displayed in Fig. 4. The white line is the contour of chopstick tube after image analysis, the green line is drawn by circle fitting, a red line is drawn by ellipse fitting, a light blue line is the maximum inscribed circle, and the purple line is the minimum circumscribed circle. Furthermore, the ellipse can get the two feature is a radius. However, the result predicted by [1] are Normal, Large, Small, Deformed chopstick tube and Empty. The above information will be used in this experiment, and the data set will be divided into four to one which are training data and testing data in this experiment. Finally, the Tensorflow and OpenCV will be implemented and compared on NVIDIA TX2 platform in this paper.

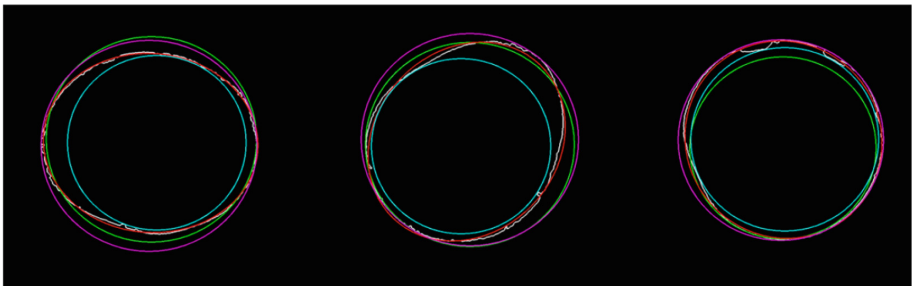


Fig. 4. Schematic diagram of chopsticks after image analysis (Color figure online)

3.1 Artificial Neural Network, ANN

The number of hidden units and hidden layers will be adjusted to optimize the ANN module. The range of hidden units is between 6 and 18, and the range of hidden layers is between 1 and 6. The weight of ANN is random in the beginning, so this training result is the average value of 5 times such as [10]. Furthermore, the parameters in

machine learning are learning rate $\eta = 0.01$ and epochs are 7000. Table 1 indicates the test data accuracy of ANN with different parameters. Hidden units and hidden layers use the optimal solution about 16 units and 3 layers, the maximum accuracy of test data is 0.96857 shown in Table 1. With 12 units and 4 layers are also the optimal solution of hidden units and hidden layers, but the module with 4 hidden layers will spend more calculating time than 3. The module with 16 units and 3 layers be chosen in this paper.

Table 1. The accuracy of test data with the various number of parameter

Net\layer	1	2	3	4	5	6
6	0.56000	0.72000	0.89714	0.84857	0.90000	0.73714
7	0.62571	0.78571	0.84571	0.85429	0.88857	0.87429
8	0.57714	0.81714	0.89429	0.92571	0.88000	0.93714
9	0.61714	0.82286	0.93714	0.94857	0.86286	0.90286
10	0.62286	0.84000	0.94000	0.94571	0.94286	0.86857
11	0.60286	0.87429	0.92857	0.96286	0.95714	0.95429
12	0.68000	0.85714	0.94857	0.96857	0.95143	0.96286
13	0.65143	0.88571	0.96000	0.96572	0.96000	0.96000
14	0.63429	0.88286	0.94857	0.96286	0.96572	0.96000
15	0.63429	0.89429	0.94286	0.95429	0.96000	0.95429
16	0.68571	0.89429	0.96857	0.95429	0.93714	0.96572
17	0.68286	0.89429	0.94857	0.96000	0.95143	0.94857
18	0.68571	0.89714	0.94571	0.94571	0.95429	0.95429

This paper will train the module up to 17000 epochs used the module with 16 hidden units and 3 hidden layers, and the result is shown in Fig. 5. And then, the blue line is the loss from root-mean-square error, the red solid line is a curve of training accuracy, and the red dotted line is a curve of testing accuracy. The training set accuracy and the test set accuracy also arrive 0.985612 and 1. The result can prove the module is not overfitting, and the module is the best for this experiment.

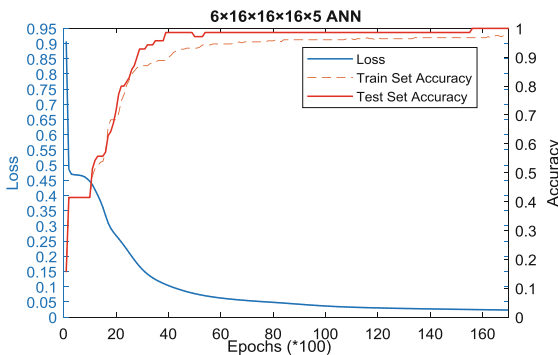


Fig. 5. Schematic diagram of chopsticks after image analysis

3.2 Random Forest, RF

RT mainly adjusted the number of DT and the maximum branch for every DT, this paper will adjust the maximum branch between 20 and 200, and the number of DT between 10 and 100. However, RF is easy to overfit during training and shown in Fig. 6. The red solid line is testing accuracy, the red dotted line is training accuracy, and the blue line is a branch of average. As a result, the testing set accuracy gradually decreases after each training, and this paper used early stopping [9] to improve the problem. Furthermore, RF used the random training set to build DT as the same as bagging algorithm, so this practice will use the average value from 10 times data.

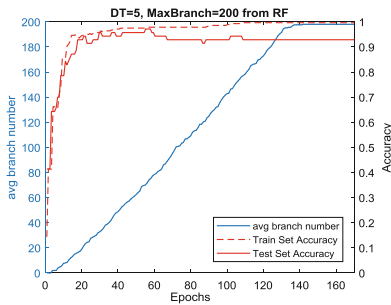


Fig. 6. The curve of DT training

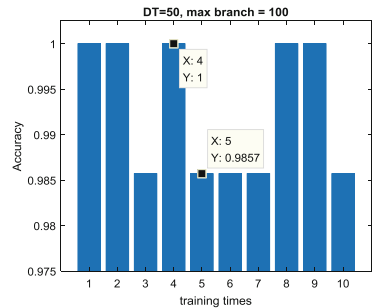


Fig. 7. Test accuracy of 50 DT and 100 branch

The result is shown in Table 2, the row and column are the numbers of DT and maximum branch. The range of table is large, so only show the quantity of DT between 20 to 80. There are 3 accuracies with optimal parameter close to 0.99286. The first, the number of DT and maximum branch are 70 and 160, the second are 30 and 180, and the third are 50 and 100. The module with parameters are 50 and 100 is more simple than others, so this module will be selected to reduce the complexity of this paper.

Table 2. Testing set recode from RT training.

MB\DT	20	30	40	50	60	70	80
20	0.97143	0.96286	0.96857	0.96429	0.95571	0.96429	0.97571
40	0.97571	0.98714	0.98857	0.98429	0.98142	0.98429	0.97857
60	0.98571	0.99000	0.98570	0.98000	0.98571	0.99000	0.98857
80	0.98429	0.98571	0.98429	0.98714	0.98429	0.99000	0.98571
100	0.99000	0.98571	0.98571	0.99286	0.98714	0.98571	0.98571
120	0.98571	0.99143	0.99143	0.98571	0.98714	0.98571	0.98429
140	0.99000	0.98571	0.98857	0.98429	0.99000	0.98857	0.98429
160	0.99000	0.98857	0.98714	0.98714	0.99000	0.99286	0.98571
180	0.98571	0.99286	0.98714	0.99000	0.98571	0.98571	0.98714
200	0.98571	0.99000	0.98571	0.98429	0.98571	0.98714	0.98571

Figure 7. displays the testing accuracy at each time of training, and the parameter of the module is 30 DT and 180 maximum branches. The x-axis is a quantity of training times. The y-axis is testing accuracy. As a result, the best module can easily arrive at in 100% for the testing accuracy in this experimental.

3.3 Support Vector Machine, SVM

The SVM mainly adjusted the kernel function and the parameters of function for the application, this paper will fix the cost penalization and iterations to look for the optimal module. Furthermore, this paper will try these kernel function: linear kernel, RBF, polynomial kernel, sigmoid kernel and inter kernel, to find the optimal parameters.

The training set accuracy and the test accuracy of a linear kernel and inter kernel showed in Table 3, a linear kernel is a normal linear SVM, inter kernel can reference the function (6), these two kernels don't have parameters to adjust.

$$K(\bar{x}, \bar{x}') = \min(\bar{x}, \bar{x}') \tag{6}$$

Table 3. Linear and inter kernel result

Information\kernel	Linear	Inter
Train set accuracy	0.838129	1.0
Test set accuracy	0.814286	0.985714

Polynomial kernel had to adjust two parameters (7), R and d are constant and the degree of a kernel function, setting the degree to 2, because of the high degree is very easy to overfit. In Table 4, this paper looks up for R between 0 to 1, but the data is too large, so only show the data between 0.545 to 0.55, and the optimal R is 0.548 for the test accuracy that is the highest value.

$$K(\bar{x}, \bar{x}') = (\bar{x}^T \bar{x} + R)^d \tag{7}$$

Table 4. Polynomial kernel result

R	0.545	0.546	0.547	0.548	0.549	0.55
Train set accuracy	0.87050	0.94245	0.93525	0.95684	0.91727	0.96043
Test set accuracy	0.87143	0.97143	0.98571	1	0.94286	0.97143

The kernel of RBF (8) can infinitely increase dimensions for a sample, γ is constant that influenced Euclidean Distance for the sample, and only γ between 0.903 and 0.908 be shown in Table 5 in this paper because the data range is too large. However, the optimal γ is 0.906 for the test accuracy that is the highest value.

$$\mathbb{K}(\bar{x}, \bar{x}') = e^{-\gamma \|\bar{x} - \bar{x}'\|^2} \quad (8)$$

Table 5. RBF kernel result

Gamma	0.903	0.904	0.905	0.906	0.907	0.908
Train set accuracy	0.98561	0.98561	0.98201	0.98201	0.98561	0.98561
Test set accuracy	0.97143	0.97143	0.98571	1.0	0.97143	0.97143

In finally, sigmoid kernel (8) has to adjust the constant R , a part of the result shown in Table 6, a sigmoid kernel is more suitable for the regressive application, so the accuracy is lower than other kernels.

$$\mathbb{K}(\bar{x}, \bar{x}') = \tanh(\bar{x}^T \bar{x}' + R) \quad (9)$$

Table 6. Sigmoid kernel result

R	1	2	3	4	5	6
Train set accuracy	0.24101	0.34532	0.37050	0.37770	0.39928	0.33094
Test set accuracy	0.21429	0.37143	0.37143	0.37143	0.41429	0.31429

In conclusion, the optimal module of SVM will choose RBF with $\gamma = 0.906$, and the test accuracy is up to 0.982014.

4 Conclusion

The accuracy of testing can increase to 100% when this structure is complicated, and the accuracy is better than a traditional system is 91%. However, the complexity is not necessarily proportional to testing accuracy. The complex model is usually overfitting for the training set. When training the module, ANN and RF are simple, but the prediction of ANN is time-consuming. However, the training of SVM is more time consuming than ANN and RF, because has to set the parameter of a different kernel, and this method will be easy to do the hyperparameters. In the task of prediction, the calculating speed of SVM faster than and easier other. As a result, SVM is more suitable than other in an application of tube detection. Furthermore, we can select module are ANN and RF, if we do not care speed of predict.

The ratio is 4:1 for training and testing data in this experiment. The noise of testing data is lower than training data, and testing accuracy can easily increase to 100% because the sample is few and noise is not uniform. In part of the evaluation, there are still many places needed to improve. Such as: evenly distributed the noise, and optimize value is hyperparameters. The above are all important topics that is worth to discuss.

Acknowledgments. This work is partially supported by the Ministry of Science and Technology, ROC, under contract No. MOST 106-2221-E-224-025, and 106-2218-E150-001.

This work was financially supported by the “Intelligent Recognition Industry Service Center” from The Featured Areas Research Center Program within the framework of the Higher Education Sprout Project by the Ministry of Education (MOE) in Taiwan.

References

1. Hung, C.-W., Jiang, J.-G., Wu, H.-H.P., Mao, W.-L.: An automated optical inspection system for a tube inner circumference state identification. In: ICAROB (2018)
2. Yuanyuan, S., Yongming, W., Lili, G., Zhongsong, M., Shan, J.: The comparison of optimizing SVM by GA and grid search. In: 2017 13th IEEE International Conference on Electronic Measurement & Instruments (ICEMI), pp. 354–360, Yangzhou (2017). <https://doi.org/10.1109/icemi.2017.8265815>
3. Ross, J.: Fuzzy Logic with Engineering Application, 3rd edn, pp. 179–183. Wiley-Blackwell (2010)
4. Cross entropy available. http://en.wikipedia.org/wiki/Cross_entropy
5. Yusong, P.O.: Generalization of the cross-entropy error function to improve the error backpropagation algorithm. In: Proceedings of International Conference on Neural Networks (ICNN'97), vol. 3, pp. 1856–1861, Houston, TX, USA (1997). <https://doi.org/10.1109/icnn.1997.614181>
6. Pang, S.L., Gong, J.Z.: C.50 classification algorithm and its application on individual credit score for banks. Syst. Eng. Theor. Pract. **39**(12), 94–104 (2009)
7. Ho, T.K.: The random subspace method for constructing decision forests. IEEE Trans. Pattern Anal. Mach. Intell. **20**(8), 832–844 (1998). <https://doi.org/10.1109/34.709601>
8. Lin, C.-F., Wang, S.-D.: Fuzzy support vector machines. In: Learning and Soft Computing: Support Vector Machines, Neural Networks, and Fuzzy Logic Models, MITP (2001)
9. Ishikawa, M., Moriyama, T.: Prediction of time series by a structural learning of neural networks. Fuzzy Sets Syst. **85**(2), 167–176 (1996)
10. Shao, Y., Taff, G.N., Walsh, S.J.: Comparison of early stopping criteria for neural-network-based subpixel classification. In: IEEE Geoscience and Remote Sensing Letters, January 2011, vol. 8, no. 1, pp. 113–117. <https://doi.org/10.1109/lgrs.2010.2052782>