Computer Science and Information Systems 20(2):701–722 https://doi.org/10.2298/CSIS220703019Z

Using Neural Network to Automatic Manufacture Product Label in Enterprise under Iot Environments

Kai Zhang[1] and Chongjie Dong[2]

[1] School of information, Guangdong Communication Polytechnic,
Guangzhou, 510650, China
gdcpzk@qq.com
[2] Department of Computer Engineering, Dongguan Polytechnic,
Dongguan, 523808, China
dchj2020@foxmail.com

Abstract. When the manufacturing industry is dealing with information technology, it has to face a large number of parameters and frequent adjustments. This study proposed artificial intelligence models to find out the hidden rules behind a large number of customized labels, through data processing and model building. Model and parameter experiments are used to improve the effectiveness of artificial intelligence models, and the method of cyclic testing is adopted to increase the diversity of the test set. The results of this paper, we integrate each stage and an auxiliary decision-making is established. The contributions of this paper, can improve the problem with reducing production line shutdown and improve factory productivity. The accuracy rate of the artificial intelligence model can be increased to 95%. The number of stoppages is reduced from 4 times to 1 time per month. Under full capacity, this assist the decision-making system can reduce loss cost.

Keywords: Artificial intelligence (AI); Machine learning; Random forest; Neural network; Automatic product label.

1. Introduction

With the development of science and technology, in order to save manpower, reduce costs, increase production capacity, and improve quality, companies have successively introduced new technologies and new concepts, from the early information and automation to today’s smart factories, Industry 4.0, artificial intelligence as the goal of intelligence [25]. To implement intelligence, automation must be carried out first. Based on automation, intelligence can make decisions based on the actual conditions of the site, and deploy the automation work sequence to optimize efficiency and productivity. To implement automation, information must be carried out first, based on information only by automation can judge the correct position, use the best path, make appropriate actions, replace labor and improve efficiency. Only in this way can maximum value of existing resources, difficulties can be reduced and benefits can be improved. However, when the manufacturing industry deals with information, it is not only necessary to import all the parameters into the system. The factory has a variety of products, machines, manufacturing processes, post-station data, and the production line is divided into production and assembly, testing, packaging, some parameters have a reciprocal and complex relationship with each other. In addition, it is necessary to deal with the problem of frequent adjustment and change of parameters [31, 25].

It is possible to design card control check logic for products with higher severity, more value and longer cycle are including data setting limit card control and post-setting check comparison. In the case of frequent changes to specifications and revisions, the information department is time-consuming and laborious, and cannot meet the overall needs. In response to this problem, this research attempts to use artificial intelligence models to establish an auxiliary decision-making system. When the user sets an error, a warning will be displayed to reduce the error rate and improve the problem [15]. In the face of a difficult to control environment with diversified customers, diversified products, variable specifications, and short production cycles in the manufacturing industry, the only thing that remains constant is always changing. Under such circumstances, the setting of the production line system is very complicated. Under the premise of sharing the settings as much as possible, there are still many customized special rules. It is easy for the setting personnel to confuse, and the setting is wrong due to accidental setting. Once the setting is wrong, it will cause loss. Defective products require maintenance, heavy industry, or even scrapping. If there is an error, the product will be scrapped in the slightest, and the line will be stopped in the serious, resulting decrease factory productivity, and even delayed shipments. In the case of full capacity, the line is stopped due to the customized label problem, which takes about half an hour to solve each time. The loss is nearly every year ten million dollars. Decrease in factory productivity will affect quality and delivery, which will cause customers to reduce orders and revenue will lead to unhappy shareholders and the board of directors; unhappy and negative attitudes at the bottom of the company will lead to a decline in factory productivity [43, 24].

This study uses artificial intelligence models to try to predict the setting value in the changing environment to reduce the setting error rate and maintain the stable productivity of the factory [1, 5]. On the technical side: When facing a single customer and a single product, if the specifications can be clarified, the final answer should be limited to a few possibilities based on conditions. When the answer falls outside these possible ranges, the probability of error is extremely high. However, it is difficult to be familiar with all the rules due to human experience, and it is difficult to pass on the experience of senior staff to new recruits through education and training. However, the information department of general enterprises has limited resources, and it is difficult to develop all specifications with limited time and manpower, not to mention clarification of specifications, meetings, and communication time cost. In order to be simple and fast, the program structure of the stacked bed frame also makes the enterprise virtually impossible with less technical debt. The data is based on the factory label, all specifications, and stages of raw materials, semi-finished products, finished products and packaging. It will be assigned numbers and labeled for identification. Common label contents include: serial number, part number, model number, date, quantity and other text and barcode. The date range of this research is from the fourth quarter of 2018 to the second quarter of 2020, the data is established by the company in recent years by sign-off process into the information-based transfer process, but the variable field has been revised. It is confirmed that there are more than 30,000 effective documents that have been applied to production line. The data retains historical version data, since changes and errors are the norm in the factory industry, after evaluation, these data are retained but not excluded, but need added to facilitate identification and influence the weight of the data through the characteristics, thereby obtaining correct results. The contributions of this paper can improve the problem with reducing production line shutdown and improve factory productivity. The accuracy rate of the artificial intelligence model can be increased to 95%. The number of stoppages is reduced from 4 times to 1 time per month. Under full capacity, this assist the decision-making system can reduce loss cost.

The remainder of the paper is as following: Section 2 provides a review of the Relative work; Section 3 outlines the research method, describes the exploration problems in this article and proposes the research process and overall structure; Section 4 is the experimental results and analysis; Section 5 is the conclusion future work.

2. Relative Work

There are three common applications of artificial intelligence in the manufacturing industry: scheduling optimization, numerical monitoring, and image recognition. The purpose of scheduling optimization is mainly to balance the load of the production line, shorten the production cycle, and reduce costs; the purpose of numerical monitoring is to reduce defective products, control yield and reduce costs; image recognition is often used in optical inspection stations for the purpose of reducing defective products, control yield, and reducing costs. There are many literatures on the application of artificial intelligence models in the manufacturing industry to study how to control the yield, but most of the artificial intelligence-related research focuses on the production environment [18, 19,20,21]. Even if problems are found through the artificial intelligence model, most cases still need to be stopped for adjustment, resulting in loss of productivity. If errors can be found when setting parameters at an earlier stage, the number of stoppages can be reduced. Fig. 1 shows the flow of the research data approval form.

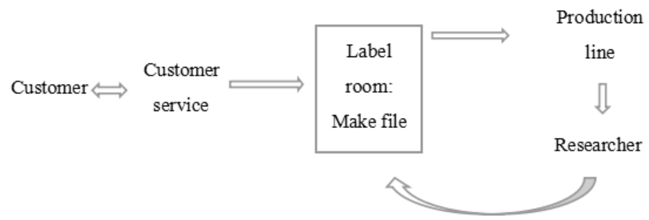

Fig. 1 Data flow

1. The customer provides the reference image file, style, printed content and other information of the label.
2. The Product Control Coordinator (PCC) obtains information through letters and

phone calls and fills in the application form.

3. The label room staff set the variable parameters corresponding to each item on the label.
4. The label room staff needs to ask information staff to assist in setting new variables.
5. The label room staff creates proofing label files based on information in the sign-off form.
6. After the approval is completed and effective, the label file is used to print the label during production on the production line.

2.1. Application of artificial intelligence in manufacturing

The relevant research on similar issues in the manufacturing industry is as follows: In 2020, Germany authors used KNN, random forest, and neural network model to solve the manpower problem required for the automatic optical inspection (AOI) result verification during the production of surface mount technology in the manufacturing industry. In order to solve the problem of the high cost of setting up inspection stations in the manufacturing industry, to replace the defect recognition system established by traditional computer vision with a deep learning model. Especially for the recognition of small objects, proposed a two-stage object detection algorithm [4, 13, 30, 32, 44]. To reduce the over-fitting situation, and finally use this automatic defect detection system to assist manual detection to reduce the missed detection rate and labor costs. Try to solve the problem of balancing the workload of the production line, reducing the number of workstations to reduce enterprise costs, and maximizing the work efficiency of the production line, using genetic algorithms, immune algorithms, and particle swarm algorithms, and proposed a new the coding method is discussed and compared, and the results show that the solution speed of the particle swarm algorithm is better than the other two algorithms, and the solution quality of the immune algorithm is better than the other two algorithms. In order to solve the problem of the deviation of the control parameters in semiconductor process production machine, which led to the deviation of the process, and caused the wafer yield to decrease, and even scrapped, A neural network combined with a failure detection and classification system to establish an early warning mechanism to ensure process yield and guaranteed production capacity [26, 31].

2.2. Artificial intelligence algorithm

The intelligence expressed by artificially manufactured machines can be called artificial intelligence, which is usually realized by computer programs and algorithms. It systematically learns from data and uses the learned knowledge to make predictions, thereby establishing expert systems and decision-making.

KNN. Nearest neighbor method, the classification of a target is determined by the classification of its K neighbors. It is explained by the spatial distribution. The concept is the same data higher the similarity, and closer the distribution. It is easy to understand in machine learning algorithms. The input of the model can be classification or regression data, and the output is the category of the target. K is the number of nearest neighbors. If it is set too small, it will reduce the classification accuracy. If it is set too large, it may increase noise and affect the results. The model can be based on the neighbor distance gives different weights to optimize the results [7, 8, 9, 22, 39, 40]. When new data is available in this model, it can be added directly without retraining, and it is not sensitive to outliers; its disadvantage is needs to be re-calculated each time for the classification requires a large amount of memory, and it is sensitive to the local structure of the data. When the data is distributed unbalanced, the forecast is prone to deviation. Commonly used in text classification, pattern recognition, cluster analysis, multi-classification fields [42].

Decision tree. It is a tree model, where each node represents a feature, and each branch represents the possible attributes of the feature. Along the tree structure, the path is determined according to each feature attribute, and finally reaching the leaf node is the answer. The decision tree model can build classification trees, regression trees, and even both classification and regression trees that can be used. Basically, the decision tree has only a single output. When building a decision tree, it is necessary to determine the order of using features to create nodes based on the amount of variation as the division criterion. There are two common processing methods: use entropy to calculate the information gain, and subtract the information disorder before the division from the information after the division. If the decision tree model loses control and builds too many branches, it will be easy to overfit, so that all data has its own path, so it is necessary to restrict growth or even pruning. This model is easy to understand and implement, has a high degree of interpretability, does not need to do too much data pre-processing, and can process data and category data at the same time. Its disadvantage is that it is easy to overfit, and small changes in data are possible. When the data is unbalanced, the growth tends to have more numerical features, resulting in poor performance and neglecting the correlation between the attributes in the data [6].

Random forest. Random forest is composed of multiple decision trees. Because a single decision tree is easy overfitting, it has the characteristics of low deviation but high variance. Random forest uses random sampling and random selection of features to build multiple decision trees to solve this problem. Its output is determined by voting on the answers of all decision trees. The input of the model can be classification or regression data, and the output is the target category. The random sampling method of random forest is to use the method of retrieval and replacement. The same sample may be selected multiple times, or none of them may be selected at random. This model can evaluate the importance of features and has a fast training time. It can process high-dimensional data. If there is missing data, it can still maintain its accuracy. If the data is unbalanced, it has balance errors; its disadvantage is that the interpretation is poor. When there is a problem, it is impossible to make a prediction beyond the data range. If the data is noisy, it may still be over-fitting. It is often used in various classification and regression problems, or used in outlier detection, and also used in unsupervised learning classification problems [10, 12, 16, 28].

Neural Network. The concept of neural network is to simulate the function and structure of biological neural network through algorithms. Each neuron is composed of its input, excitation function, and output. The entire network is composed of input layer, hidden layer, and output layer. The input layer simulates numerous neurons receive huge non-linear input information. The hidden layer simulates the synaptic connection of neurons and is responsible for transmitting the information to the corresponding position. The more complex non-linear relationships, the more it can lead to over-fitting [34, 36]. The output layer: The simulation information is processed through neural connections, and the model can learn by optimizing the weights of neurons in each layer and the connections between neurons. The input of the model can be classified or regression data, but the categorical data needs to be converted into numerical values through one-hot encoding. In the entire network, each neuron is connected to all the neurons in the next layer, and the neurons in the same layer. The elements are not connected to each other. The number of layers in the network and the number of neurons in each layer are determined by the complexity of the problem. It can be optimized and controlled during the parameter tuning stage. Usually, when the number of neurons is the same is a deep neural network. The performance is better than that of shallow neural networks. In addition to the development of deep learning research, it also extends more advanced methods such as recurrent neural networks and convolutional neural networks [27].

This model has self-learning ability, can learn the rules and patterns behind it from the data, and the learned knowledge is scattered throughout the neural network, so it has a certain fault tolerance, and a small part of the damage will not cause too much to the whole It can also adjust itself based on the learned results, and combined with the newly provided data; its disadvantage is that the learning speed is slow, and it is not explanatory. When the neural network is deeper and more complex, the huge amount of parameters requires a larger amount of data assists in training, otherwise it is easy to overfit. Because neural networks have infinite possibilities, it is difficult to find the best solution. In the process of tuning parameters, only a lot of attempts can be used to obtain better parameters. Commonly used in speech recognition, image recognition, recommend systems [33, 35].

3. Research Methods

The data used in the customized labeling and endorsement form of electronic manufacturer. The fields filled in the applicant are the input data of the artificial intelligence model. The variable setting parameters actually set in the label room are the artificial intelligence model. This experiment will build hardware specifications: processor CPU Xeon-E5-2650, memory DDR4 64 GB, operating system Windows 10, software based on open source software Python, and use the python library provided by OpenCV, and use Keras, Tensorflow, etc. as neural network development tools. Implemented in Windows and Linux systems.

3.1. Research Structure

As shown in Fig. 2, the research steps are mainly divided into three blocks: data processing, construct modeling, and optimized experiment. The purpose of data processing in the first block is to prepare the input data required for construct modeling, including data collection, sorting and screening. Data pre-processing that the model can operate normally and correctly. Feature construction and selection to clarify the importance of features. The purpose of construct modeling, in order to find a model that is more suitable for the data type, seven models that are well-known in the performance of categorical data are selected and compared with each other. Optimized experiment based on the results of the previous block, after finding a better model, try to optimize further and obtain better results. This part is divided into three types of experiments, feature processing experiments, circuit training experiments, and parameters experiment. Finally, the stage results of each step can be used as fixed parameter for other steps, and optimize other steps.

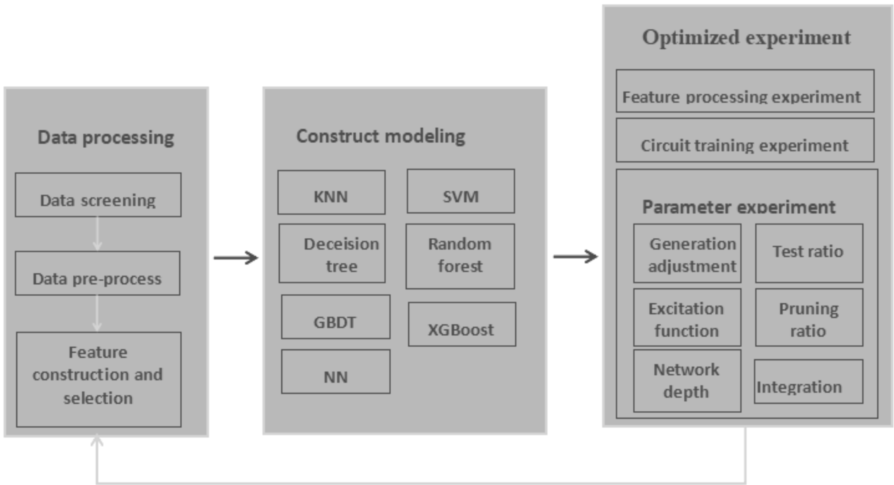

Fig. 2. Research architecture

3.2. Data Collection, Sorting and Screening

The data of this research is stored in the company's internal database. According to the approval form process, three types of data sheets are designed to store. One is the approval form data, which records the contents filled in by each unit in the transmission and can be based on the site authority of the approval process modification; Two is the historical version data that has been validated after the approval is completed, and it is provided for confirmation by the customer and the inspection unit, which is also the basis for the next revision; the third is the data actually used by the production line after it takes effect, which is extremely sensitive data. Once modified, it will be directly modified affect the production line. The data used in this study falls into the second category: historical version data that has been validated and entered into force. As the information in subscribe, there will be many blank fields that have not been filled in, and there may be errors that have not been finalized. On the other hand, the data actually used by the production line is only the latest version of the data, the data is not continuous, and there is no past data, resulting in too little data volume, it is difficult to find out the setting rules. The historical version data is only recorded when the production line takes effect, not only can ensure the accuracy of the data, but also can be traced back to the past. Using this data to train an artificial intelligence model, it is expected that the model can learn the rules behind each customized label over time and version evolution. The data are stored independently in five factories under the same structure. Different factories may face different customers, fill in forms and set up personnel, and the proportion of data varies greatly. Compared with the scale of data 3 to 26 times, so we finally decided to use only the largest plant area data for research.

The data comes from 4400 signed forms, but because only the second type of signed forms are used, the number of valid forms is actually only 3800, which is about the total number of forms. The source of these forms comes from 2800 different material numbers, but only 2500 material numbers have gone through the sign-off process. Each form contains one or more different label types, and all the used label types are 24 types, only 23 types are actually applied to the production line. In practice, there are 6 types that are more commonly used: 1. CB_SN, semi-finished product customer label, mostly a two-dimensional code, used to record the serial number of the semi-finished product stage; 2. SN, production line input the serial number label, mostly one-dimensional code, to record the production serial number in the factory; 3. FCC, the product label on the host, mostly one-dimensional code, record the product serial number; 4. BOX, the label on the color box, mostly one-dimensional code, record the serial number of the color box; 5. CARTON, the label on the outer box, mostly composite bar code, including the outer box serial number, product specifications, quantity, content serial number list; 6. PALLET, pallet label, mostly composite bar code, including pallet serial number, product specification, quantity, content serial number list. The other types are usually an extra label at the same position to display additional content or special specifications. Because the amount of data is small or too special, this research will skip these data and not adopt it, and use six types is the main research data.

A label type contains one or more different variables. Among 3800 completed forms contains 41000 variable data. However, the variable settings have been revised. The old variable names have been discarded and no longer used. There are only 24500 data on the new system variables. There are only 22,600 items left. The field that may be incorrectly set by the setting personnel is the variable parameter field. It will list as the final prediction target of the artificial intelligence model. Under all label types, there are 580 answers in the variable field. Focusing on six types of data, there are 460 answers after screening. Among all the types, the answers to some variables are the same. After taking the intersection, there are 270 types of answers, while the six types of intersections are left with 260 types of answers. The unpredictable unknown is regarded as one answer added, namely the largest output layer dimension of the artificial intelligence model. It use the 580 variable types, the number of variable types and the amount of data of each type are not even. Sort is according to the number of variable types: FCC> CARTON> BOX> SN> PALLET> CB_SN; sort is according to the amount of data: CARTON > BOX> FCC> PALLET> CB_SN> SN. The more types of variables, the more complicated the rules behind, the higher the error rate may be included; if the amount of data is less, the more difficult it is to accumulate experience and learn the rules, and the error rate will also increase. Looking at these two elements together, the ratio of the amount of available data divided by the number indicates how much data can be used in training for each answer. Sort is according to this ratio: CB_SN> CARTON> BOX> PALLET> the sum of the six categories> All> SN> FCC, it can be expected that CB_SN has the most learning materials and is the easiest to learn; FCC has the least learning materials and is the least easy to learn.

All types of data, the utilization rate of variable types is extremely uneven. The usage rate of serial number, quantity and date is particularly high. Among the 583 variable types, the top two together account for 20%, the top four together account for 30%, and the top six together account for 40%. The top eight together accounted for 50% of the ratio. For the overall accuracy rate, it uses the sum of the proportions of several commonly used variables as one of the control groups, and then compares it with the trained artificial intelligence model.

3.3 Data pre-processing

In the process information, it is inevitable that some data is incomplete. It may be that the data table structure design is not perfect at the beginning, resulting in some data not being collected at the initial stage; or the personnel are not rigorously operated, causing the data be misplanted; or the data is improperly maintained and other modifications are made. Data is accidentally moved by mistake, causing data abnormalities; it may also cause abnormal system instability and data damage. The situation and processing methods of incomplete data are as follows: When there are missing values or missing data, the common processing method is to discard or supplement the value. Because the amount of data is not large, the available data should not be discarded as much as possible.

In some models, the category field needs to be converted by one hot encoding before it can be used. Most of the data fields are category data, and one category field needs to be converted into multiple values using one hot encoding. The fields of 0 or 1 are then passed as input to the model for learning. The text field is usually processed after cutting keywords or conversion vectors. However, some field data may contain valuable hidden data. After combining professional knowledge and experience, the hidden information of the data can be integrated and extended. The data implies some information that cannot be seen directly on the surface or item number. The first two codes of the data represent the production stage of the factory for the item number that extract into a new field may help the model.

3.3. Feature Construction and Selection

The original data contains multiple features, but not all features are necessary. Too many features may interfere with the learning of the model, and there may even be doubts about dimensional explosion. Reducing unnecessary features can reduce the calculation of the model. In practice, it is interpretable to convince the boss or customer. Many reasons show the necessity of analyzing the importance of features. After analyzing the importance of data features, it can be used to verify the correctness of the model. If the prediction is inaccurate, this judgment basis can more quickly clarify the problem and improve the model. For important features, in addition to optimizing the existing model, it also collecting follow-up data, it can enhance the user's proper use of important features. At this stage, it tries to clarify the importance of each feature, find out the key features in different feature fields, and confirm whether each feature is as expected to help the model predict the correct answer. In addition to preliminary screening based on experience and professional knowledge, it uses the random forest model to explore the importance of features. Random forest judges the importance of features by calculating the contribution of each feature on each tree, and then take the average value and draw a graph.

3.4. Common Model Modeling and Comparison

The data field is mainly based on categories. When you are not sure which model is more suitable for this data type, try and compare several common models that perform better in classification. The data that has been pre-processed before is modeled, and the effectiveness is evaluated. As shown in Table 1, the advantages and disadvantages of the artificial intelligence model are sorted out.

Table 1 Comparison of advantages and disadvantages of artificial intelligence models

Model	Advantages	Disadvantages
KNN	Information can be added directly without retraining. It is fast and insensitive to outliers.	Each classification needs recalculate. The memory demand is large. When data distribution is unbalanced, the forecast is easy to be biased.
SVM	Strong generalization ability and is easy to understand. Part of the data can be used to make hyperplane decisions. It works well for processing high-dimensional data.	Poor performance and sensitive to missing data. There is no universal solution to nonlinear problems. The explanation of the kernel function is not high.
Decision tree	Easy to understand and implement, and has high degree interpretability. No need to do much data pre-processing, can process data and category information at the same time	It is easy to overfit, the result of data changes is unstable, and the performance is poor when the data is unbalanced. Ignore the correlation between the attributes in the data.
Random forest	Not easy to overfit, training speed is fast, The importance of features can be assessed with high accuracy. Can deal with missing data, and unbalanced data.	The interpretation is poor, when dealing with regression problems. It is impossible to make predictions beyond the scope of the data. If the data is noisy, it may still overfit.
GBDT	Can prevent overfitting and does not require complex features, characteristic processing, non-linear transformation, can process linear or non-linear data.	The computational complexity is high. Failure to parallelize is time-consuming. Not suitable for sparse high-dimensional data (Liu et al., 2018).
XGBoost	Prevents over-fitting, nodes at the same level can be parallel processing, can handle sparse data.	When a node splits, it is necessary to traverse the data set. Use twice as much memory (Jidong et al., 2018).
NN	Flexible, it has fault tolerance and self-adjustment ability.	It is not explanatory and computationally intensive. A lot of information need tuning and requires more trial and error.

3.5. Optimization model

Feature processing. Through more complete data pre-processing, the model learning effect can be better, mainly for the processing of non-required fields and fuzzy data. Use all the models to try to learn the modeling for the data before and after processing, and then evaluate and compare benefit. Complementary value method: SN_NAME is required, but may be an alias. For fuzzy processing method, there is no mandatory and standardized way to fill in these two fields, and there is no restriction on symbols and capitalization. There may be several types of filling for the same object. Therefore, after removing the symbols in the two fields, only letters, numbers and then the letters are converted to uppercase, so that the fuzzy approximate data can be integrated into consistent data.

Loop experiment. After testing the neural network, it was found that the data was too inconsistent with the actual application. After analysis, it was found that the data has time characteristics. If the training set and the test set are randomly selected, there may be future data of the test data in the training set. The answer before verification may cause the model to get out of control, overfitting, and failing to learn correctly. Therefore, the experiment was designed to clarify the actual situation based on the test set segmentation method. During the analysis and evaluation, it was found that the material numbers in the data were unevenly distributed in the entry action. Due to the time characteristics, the test data could not be randomly selected. In this case, the data diversity in the test set was low. As shown in Fig. 3, in order to increase the diversity of the test set, partial data is used section by section through loops, so that more diverse data can have the opportunity to act as test data.

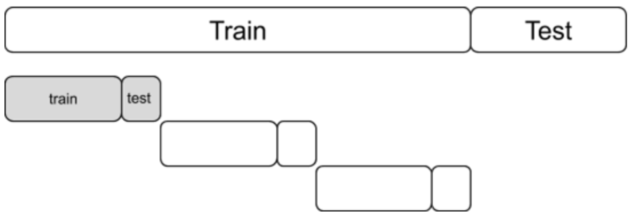

Fig. 3. Schematic diagram of loop training

As shown in Fig. 4, in order to increase diversity, different methods of using partial data are designed so that more data can have the opportunity to act as test data. In order to avoid the use of partial data and too little data, the models are all memorizing answers, this experiment uses all types together.

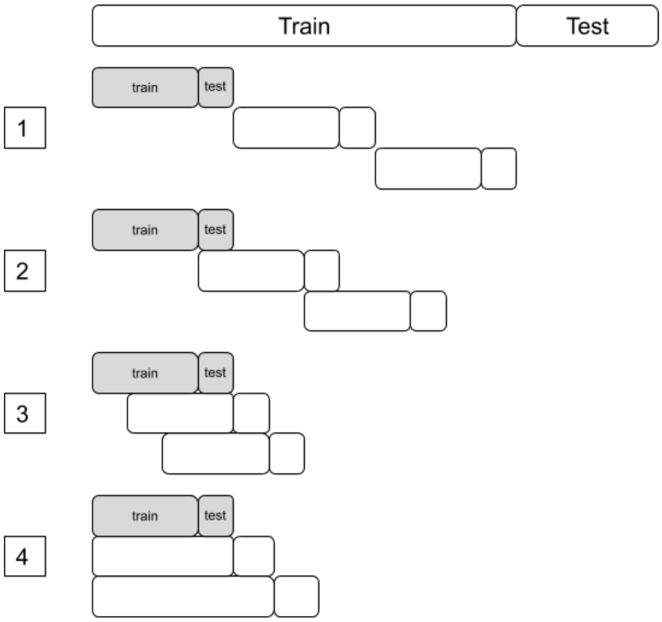

Fig. 4. Schematic diagram of partial data usage of loop training

Method one is that all data is used only once regardless of training or testing; method two is that all data will be put into the training set once; method three is that all data will be used once in the test set; method four is that all data will be used in the test set once, the used data as the next training set, and then take a certain percentage of the new data as the test set.

Parameter experiment. The model needs to adjust the parameters according to the data type to find a better parameter set to perform more ideally, so that the model has a better performance. Based on the results of modeling comparison, this part is for neural network. First, for the generation parameters, observe the learning curve of the data to find out the generation parameters that are suitable for the data type; then adjust the test set proportions and observe the changes in accuracy to determine the better test set proportion parameters; then adjust the excitation function, the time factor is also taken into consideration to determine the better excitation function parameters; in order to prevent over-fitting, observe the changes in the accuracy rate to find the better pruning ratio parameters; to ensure neural network model has enough neuron connections to learn the data rules completely, do experiments to confirm the best network depth; finally, integrate the parameters obtained at each stage to verify whether the neural network model progresses as expected.

4. Results and Analysis

The research results and experimental results are presented and analyzed in the following. Although the results of some experiments are not very significant, as long as each link can be optimized, it will eventually bring great improvement.

4.1. Feature Construction and Selection

Using the random forest model, it is found that the features are not the original data field names, and the importance proportion is also very low. Most of the researched data are categorical fields, which have been converted into multiple fields after one hot encoding. The importance of each original feature can be added the importance value of each derived field based on the original field. That SN_NAME has the highest importance, which is higher than SN_TITLE. The importance of ACTION_TYPE is the lowest.

4.2. Common Model Modeling and Comparison

The control group in this experiment uses practical statistical data in company. Each time the form is entered, the average change content is 20%. If the entire form is sent directly without modification, the accuracy rate is 80%.

Table 2. Experimental results of different artificial intelligence model

	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
Blind guess	0.11	0.17	0.25	0.21	0.15	0.18	0.22	0.01
Base				0.80				
KNN	0.85	0.91	0.85	0.92	0.76	0.84	0.72	0.87
SVM	0.26	0.38	0.40	0.83	0.28	0.58	0.28	0.44
Decision tree	0.19	0.29	0.39	0.74	0.25	0.57	0.45	0.47
Random forest	0.87	0.93	0.88	0.94	0.85	0.87	0.89	0.89
GBDT	0.764	0.86	0.80	0.92	0.73	0.75	0.69	0.78
XGBoost	0.84	0.92	0.84	0.95	0.78	0.85	0.78	0.84
NN	0.88	0.89	0.88	0.93	0.84	0.89	0.80	0.86

As shown in Table 2, the horizontal axis of this table is of various types: All means all types; others is the remaining data after excluding the six types. The vertical axis is the control group and various models. Looking at the vertical axis, that regardless of the model, the accuracy of CB_SN is very high. It can be found that the target field of CB_SN has the least type, but in addition to looking at the target field type, attention should also be paid to the amount of data. Divide the number of target fields of each type by the ratio of the amount of data to sort to get: CB_SN > CARTON> BOX> PALLET> All > SN> FCC. It founds that the higher the data ratio, the more data available for training, the higher the accuracy rate, because there is still a problem of sparseness between the amount of data and the target field that needs to be considered.

The CARTON data ratio is the 2nd, and the BOX data ratio is the 3rd. However, the performance of these two types in SVM and decision tree is worse than that of the 4th PALLET in the data ratio. Looking at the horizontal axis, the accuracy of KNN, random forest, XGBoost, NN and other models exceeds the control group, reaching the reference standard, and the performance is good.

4.3. Evaluation of the Effectiveness of Data Pre-processing

In the previous step to clarify the importance of features, it shows that SN_TITLE and SN_NAME are the more important fields, and these two fields have room for further optimization. After compensation and processing of fuzzy data, each artificial intelligence model is retrained and observed accuracy. Complementary value method: SN_NAME is required, but may be an alias; SN_TITLE is not required and may be empty. Filling in the empty SN_TITLE into the data to the SN_NAME field. Fuzzy processing method: There is no mandatory and standardized way to fill in these two fields, as long as users of each station can understand it, and there is no restriction on symbols and capitalization. Depending on the user's habits, there may be several types of filling for the same object. Therefore, after removing the symbols in the two fields, only letters, numbers, and Chinese are left, and then converted to uppercase, so that the fuzzy approximate data can be integrated into consistent data.

As shown in Table 3, the horizontal axis of this table is various types, the vertical axis is the control group and various models, and the data column is the accuracy rate change value. It shows that SVM and decision tree have increased significantly, and the accuracy rate has increased; as for KNN, random forest, GBDT, XGBoost and NN, the impact is less significant. Some models may have already learned the upper limit on the existing data and features, so there is not so much for optimization, or some models such as random forest can deal with the problem of missing values or fuzzy data.

4.4. Evaluation of the Effectiveness of Loop Training

The experiment tries to optimize the neural network model such as the best overall performance, explains the purpose of the loop test, analyzes and discusses the experimental results.

Comparison of effectiveness evaluation of loop training. 80% of the data is used for training, and only 20% is used for verification. Because the data field information is mostly categorical data, and the interval time of each material number is inconsistent, resulting in uneven data distribution. Due to time characteristics, the test data cannot be randomly selected. In this case, the data diversity in the test set is low. In order to solve this problem, this experiment was designed in order to increase the diversity of the test set, partial data is used section by section through a loop, so that more data can have the opportunity to act as the test data.

Table 3. Experimental results after data processing

	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
KNN	+0.01	-0.01	+0.01	+0.01	+0.01	+0.02	+0.02	+0.01
SVM	+0.21	+0.18	+0.14	+0.09	+0.14	+0.02	+0.09	+0.05
Decision tree	+0.09	+0.09	+0.06	+0.11	+0.18	-0.01	+0.01	+0.02
Random forest	+0.00	-0.03	-0.01	+0.00	+0.02	+0.02	+0.01	+0.03
GBDT	+0.00	+0.05	+0.02	+0.01	+0.02	+0.01	+0.05	-0.01
XGBoost	+0.03	+0.01	+0.01	-0.02	-0.03	-0.00	-0.01	-0.01
NN	+0.00	+0.01	-0.01	+0.01	+0.00	+0.00	+0.03	+0.01

As shown in Table 4, the horizontal axis of this table is various types, and the vertical axis is the number of cycles. In this experiment, the total data volume of the test set is the same as that of the control group. It can be proved that only the test set is dispersed and increased diversity. It can indeed improve results. Only the results of the SN type are lower than the control group, which may be related to the fact that most of the data types are special cases. In table 4 that the best result has no significant relationship with the number of cycles, and it is speculated that it should be more related to the diversity of the test set distribution.

Table 4. Results of the loop training experiment

	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
NN Base	0.89	0.90	0.86	0.94	0.84	0.89	0.84	0.87
Roll 2	0.91	0.93	0.91	0.92	0.83	0.90	0.79	0.86
Roll 3	0.92	0.95	0.93	0.94	0.82	0.94	0.78	0.84
Roll 4	0.91	0.97	0.92	0.94	0.83	0.90	0.77	0.89
Roll 5	0.90	0.97	0.93	0.94	0.80	0.88	0.71	0.87
Roll 6	0.90	0.95	0.96	0.93	0.87	0.86	0.71	0.84
Roll 7	0.89	0.94	0.94	0.92	0.89	0.84	0.83	0.85
Roll 8	0.91	0.95	0.95	0.88	0.83	0.87	0.78	0.80
Roll 9	0.92	0.93	0.91	0.84	0.91	0.80	0.78	0.78
Roll 10	0.91	0.93	0.94	0.88	0.84	0.86	0.70	0.76

Comparison of effectiveness evaluation of loop training. In order to increase diversity, design different methods of using partial data, so that more data can have the opportunity to act as test data. In order to avoid the use of partial data, the amount of data is too small, resulting in the model is memorizing answers, and therefor experiment only uses all types to carry out. The control group is the basic neural network model and method one of the previous experiment. Method one is that all data is used only once regardless of training or testing; method two is that all data will be put into the training set once; method three is that all data will be used once in the test set; method four is that all data will be used in the test set once, the data used each time is used as the next training set, and a certain percentage of the new data is taken as the test set.

As shown in Table 5, the horizontal axis of this table is the amount of data for each training of methods one to three, the control group and various methods, and the vertical axis is the proportion of the amount of data used for each cycle training.

Table 5. Advanced experimental results of loop training

Data volume	Training set	Test set	NN Base	Method 1 Base	Method 2	Method3	Method4
1/2	9796	2450		0.91	0.91	0.92	0.89
1/3	6531	1633		0.92	0.93	0.92	0.84
1/4	4898	1225		0.91	0.94	0.94	0.86
1/5	3918	980		0.90	0.93	0.92	0.86
1/6	3265	817		0.90	0.95	0.95	0.89
1/7	2798	700		0.89	0.89	0.94	0.76
1/8	2448	613	0.89	0.91	0.94	0.92	0.79
1/9	2176	545		0.92	0.91	0.91	0.79
1/10	1959	490		0.91	0.92	0.92	0.79

From table 5 shows that the results of method two and method three are better than those of the control group, but only looking at the data, there is no obvious advantage or disadvantage between method two and method three. The result is generally lower than that of the control group. After analysis, it is found that method 2 is relatively normal, and method 3 is suspected of overfitting. Therefore, method 2 is selected, and the amount of data is 1/6 which is the better parameter for the experiment.

4.5. Evaluation of the effectiveness of model optimization

The experiments are aimed at optimizing neural network models such as the best overall performance, and discuss and compare the results after sorting out the experimental results.

Comparison of effectiveness evaluation of adjusting the test set proportion. This experiment adjusts the test set ratio and observes the changes in accuracy. As shown in Table 6, due to the numerous combinations of loop training methods and data volume, this experiment only takes the methods and data volume parameter sets that perform well in the previous loop training experiments for further attempts. The horizontal axis is the control group and cyclic training parameters, the first parameter is the method, and the second parameter is the data volume denominator; the vertical axis is the test set ratio, under NN Base, 0.1> 0.2> 0.15> 0.3> 0.25; Under Roll(1,9), 0.1> 0.15> 0.2> 0.25> 0.3; under Roll(2,6), 0.1> 0.2> 0.15> 0.3> 0.25; under Roll(3,7), 0.2> 0.25> 0.15 = 0.3> 0.1. The overall effect is better with a result of 0.2. At first glance, the smaller the test ratio, the higher the accuracy rate. However, after analysis, it is obvious that the high accuracy rate is illusion when the test ratio is small. As the test ratio is less, the diversity of test data is lower. The more difficult it is to verify the correctness of the data; if the test ratio is too high, it will in turn lead to too low diversity of training data, resulting in a sharp drop in accuracy, so 0.2 is the better test ratio in NN based method.

Table 6. Test proportions in the loop training experiment result

Test ratio	NN Base	Roll(1,9)	Roll(2,6)	Roll(3,7)
0.1	0.91	0.96	0.97	0.83
0.15	0.89	0.92	0.93	0.92
0.2	0.89	0.92	0.95	0.96
0.25	0.80	0.89	0.93	0.94
0.3	0.81	0.87	0.93	0.92

Comparison of effectiveness evaluation of adjusting trigger function. This experiment adjusts the excitation function and observes the changes in accuracy.

Table 7. Excitation function experiment results

Parameter	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
softmax	0.12	0.12	0.23	0.24	0.12	0.17	0.18	0.14
sigmoid	0.88	0.88	0.87	0.92	0.79	0.87	0.82	0.83
elu	0.88	0.89	0.87	0.94	0.83	0.89	0.83	0.85
relu	0.89	0.90	0.86	0.94	0.84	0.89	0.84	0.87
selu	0.89	0.91	0.88	0.95	0.85	0.89	0.82	0.87

As shown in Table 7, the horizontal axis of this table is each type, and the vertical axis is the excitation function parameter. Under the all type, relu> selu> elu> sigmoid> softmax; under the BOX type, selu> relu> elu> sigmoid> softmax; CARTON Under the type, selu> elu> sigmoid> relu> softmax; under the CB_SN type, selu> elu> relu> sigmoid> softmax; under the FCC type, selu> relu> elu> sigmoid> softmax; under the PALLET type, relu> selu = elu > sigmoid> softmax; for SN type, relu> elu> sigmoid> selu> softmax; for Others type, selu> relu> elu> sigmoid> softmax. On the whole, relu ≒ selu> elu> sigmoid> softmax, but the difference between relu, selu, elu, and sigmoid is not big, the difference is less than 1% for the All types; the difference is about 4% for the BOX type; the difference is about 4% for the CARTON type. the difference is about 2.5% for the CB_SN type; the difference is about 6% for the FCC type; the difference is about 2% for the PALLET type; the difference is about 1.5% for the SN type; the difference is about 4% for the others type. Since the execution time of relu is faster, the time difference is about 20%, so relu is the better setting.

Adjusting the depth of the network effectiveness evaluation comparison. The data in this study has been transformed or the input layer has a dimension of about 1,000, and the output layer has a dimension of about 256. This experiment adjusts the number of hidden layers to observe the changes in accuracy. Due to the limited amount of data in this study, if the depth of the network is too deep, the parameters will be too large and over-fitting. Therefore, this experiment only tested three hidden layers at most.

Table 8. Network depth experiment results table

layer	All	BOX	CARTON	CB_SN	FCC	PALLET	SN	Others
1	0.88	0.88	0.88	0.92	0.79	0.89	0.83	0.84
2	0.89	0.90	0.87	0.94	0.84	0.89	0.84	0.87
3	0.89	0.90	0.87	0.93	0.83	0.89	0.82	0.85

As shown in Table 8, the horizontal axis of this table is each type, and the vertical axis is the number of hidden layers. Under the All type, 2> 3> 1; under the BOX type, 3> 2> 1; under the CARTON type, 1> 3> 2; Under CB_SN type, 3> 2> 1; Under FCC type, 2> 3> 1; Under PALLET type, 3> 2> 1; Under SN type, 2> 1> 3; Under Others type, 2> 3> 1. On the whole, the difference between the results of each network depth is not big, the difference is less than 1% under the All type; the difference is about 2% under the BOX type; the difference is approximately 2% under the CARTON type; the difference is approximately 3.5% under the CB_SN type; The difference is about 5.5% for FCC type; about 1% for PALLET type; about 2% for SN type; and about 3% for others type. In order to solve the problem of over complexity, the deeper the neural network is the better, but if it is too deep, it may cause overfitting. From the table that under different types, the best and the worst are not much different. The neural network model can learn well under the basic structure, so it is impossible to optimize too much through this parameter. The overall effect is the result of second hidden layers is better. Due to the numerous combinations of loop training methods and data volume, this experiment only takes the methods and data volume parameter sets that perform well in the previous loop training experiments for further attempts. As shown in Table 9, the horizontal axis of this table is the control group and loop training parameters. The first parameter is the method and the second parameter is the denominator of the data volume; the vertical axis is the number of hidden layers, under NN Base, 2> 3> 1; under Roll (1, 9), 3> 2> 1; under Roll (2, 6), 2> 1> 3; under Roll (3, 7), 1> 2> 3; the overall effect is better with second hidden layers, but the difference is very small. The difference between the results of each network depth is not big, the difference is about 0.5% under Roll (1, 9); the difference is about 0.5% under Roll (2, 6); the difference is about 0.5% under Roll (3, 7). In order to solve overly complex problems, the deeper the neural network depth is the better, but too deep may cause over-fitting. Under different types, the difference between the best and the worst is only 0.5% are no obviousdifferent. The neural network model can learn well under the infrastructure even after it is trained in a loop. Therefore, it is impossible to optimize too much through this parameter, and the overall effect is better. The results of second hidden layers is better.

Table 9. Network depth in the loop training experiment results table

Layer	NN Base	Roll(1,9)	Roll(2,6)	Roll(3,7)
1	0.88	0.90	0.94	0.96
2	0.90	0.91	0.95	0.95
3	0.89	0.92	0.93	0.94

5. Conclusion

This research collects the fill-in content of the approval form of customized labels in the manufacturing industry to predict production line labels. In the study, such as random forests and neural networks were used for training and prediction. The experiments results of this research show that the random forest and the neural network method is better performance in AI method. The auxiliary decision-making system established by the better trained model can reduce the error rate in practice and maintain the productivity of the factory. Through the results of this research, it shows that among the data features, the label alias is the most important and has the greatest impact. The contributions of this paper:

1. According to statistics, the production line before tuning, the accuracy rate of new recruits is only about 80%, after use the best of artificial intelligence models, can reach to 95%.
2. The number of stoppages is reduced from 4 times to 1 times per month. In the case of full capacity, this research can assists decision making system to reduce loss cost.
3. The production line labels decision making was established, can enhance the operating experience and improve work efficiency, reduce error rate and increase factory productivity.

This artificial intelligence module is based on the factory approval form and parameter settings. There are many systems in the company that involve verification and parameter setting. If this research can be extended to other application scenarios in the factory, it will be optimize and overall improvement. Future work, if the filling standard can be promoted, additional valuable implicit information can be obtained from this study when conducting research in the future. It also can assist artificial intelligence model for training and learning too. Limitation is setting errors may cause abnormalities during production and reducing factory productivity and bring losses to the enterprise.

References

1. Bai Y, Li C, Sun Z, Chen H, (2017) Deep neural network for manufacturing quality prediction," 2017 Prognostics and System Health Management Conference. Harbin, 2017, pp. 1-5, doi: 10.1109/PHM.2017.8079165.
2. Cheng, H., Wu, L., Li, R., et al. (2021) Data recovery in wireless sensor networks based on attribute correlation and extremely randomized trees[J]. Journal of Ambient Intelligence and Humanized Computing, 12(1): 245-259.
3. Cheng, Y., Jiang, H., Wang, F., et al. (2019) Using High-Bandwidth Networks Efficiently for Fast Graph Computation[J]. IEEE Transactions on Parallel and Distributed Systems, 30(5): 1170-1183.
4. Dai, Y., Wang, S., Chen, X., et al. (2020) Generative adversarial networks based on Wasserstein distance for knowledge graph embeddings [J]. Knowledge-Based Systems, 190: 105165.
5. Ding Z, Peng W, Yan Q, Lin F (2019) Research on Intelligent Manufacturing System of Sustainable Development," 2019 2nd World Conference on Mechanical Engineering and Intelligent Manufacturing (WCMEIM), Shanghai, China, 2019, pp. 657-660, doi: 10.1109/WCMEIM48965.2019.00139.
6. Elaidi H, Elhaddar Y, Benabbou Z, Abbar H (2018) An idea of a clustering algorithm using support vector machines based on binary decision tree," 2018 International Conference on Intelligent Systems and Computer Vision (ISCV), Fez, 2018, pp. 1-5, doi: 10.1109/ISACV.2018.8354024.
7. Fu, Y. G., Huang, H. Y., Guan. Y., et al. (2021b) EBRB cascade classifier for imbalanced data via rule weight updating [J]. Knowledge-Based Systems, 223: 107010.
8. Fu, Y. G., Ye, J. F., Yin, Z. F., et al. (2021a) Construction of EBRB classifier for imbalanced data based on Fuzzy C-Means clustering[J]. Knowledge-Based Systems, 234: 107590.
9. Fu, Y. G., Zhuang, J. H., Chen. Y, P., et al. (2020) A framework for optimizing extended belief rule base systems with improved Ball trees[J]. Knowledge-Based Systems, 210:

106484.
10. Ghosh S, Banerjee C (2020) A Predictive Analysis Model of Customer Purchase Behavior using Modified Random Forest Algorithm in Cloud Environment. 2020 IEEE 1st International Conference for Convergence in Engineering (ICCE).
11. Jidong L, Ran Z (2018) Dynamic Weighting Multi Factor Stock Selection Strategy Based on XGboost Machine Learning Algorithm," 2018 IEEE International Conference of Safety Produce Informatization (IICSPI), Chongqing, China, 2018, pp. 868-872, doi: 10.1109/IICSPI.2018.8690416.
12. Kou L, Liu C, Cai G, Zhang Z, Li X, Yuan Q (2020) Fault Diagnosis for Power Converters based on Random Forests and Feature Transformation. 2020 IEEE 9th International Power Electronics and Motion Control Conference.
13. Li, X. Y., Lin, W., Liu, X., et al. (2022) Completely Independent Spanning Trees on BCCC Data Center Networks with an Application to Fault-Tolerant Routing[J]. IEEE TRANSACTIONS ON PARALLEL AND DISTRIBUTED SYSTEMS, 33(8): 1939-1952.
14. Liu A, Liu X, Yu H, Zhang C, Liu Q, Tao D (2021) Training Robust Deep Neural Networks via Adversarial Noise Propagation. IEEE Transactions on Image Processing. 30.
15. Liu S, Cui Y, Ma Y, Liu P (2018) Short-term Load Forecasting Based on GBDT Combinatorial Optimization," 2018 2nd IEEE Conference on Energy Internet and Energy System Integration (EI2), Beijing, 2018, pp. 1-5, doi: 10.1109/EI2.2018.8582108.
16. Liu Y, Liu L, Gao Y, Yang L (2019) An Improved Random Forest Algorithm Based on Attribute Compatibility," 2019 IEEE 3rd Information Technology, Networking, Electronic and Automation Control Conference (ITNEC), Chengdu, China, 2019, pp. 2558-2561, doi: 10.1109/ITNEC.2019.8729146.
17. Liu Y, Zhao Y, Tao L, Zhao K, Li K (2018b) The Application of Digital Flexible Intelligent Manufacturing System in Machine Manufacturing Industry," 2018 IEEE 8th Annual International Conference on CYBER Technology in Automation, Control, and Intelligent Systems (CYBER), Tianjin, China, 2018, pp. 664-668, doi: 10.1109/CYBER.2018.8688303.
18. Liu, G., Chen, X., Zhou, R., et al. (2021) Social learning discrete Particle Swarm Optimization based two-stage X-routing for IC design under Intelligent Edge Computing architecture[J]. Applied Soft Computing. 10, 107215.
19. Liu, G., Chen, Z., Zhuan,g Z., et al. (2020a) A unified algorithm based on HTS and self-adapting PSO for the construction of octagonal and rectilinear SMT[J]. Soft Computing, 24(6): 3943-3961.
20. Liu, G., Zhang, X., Guo, W., et al. (2022a) Timing-Aware Layer Assignment for Advanced Process Technologies Considering Via Pillars. IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems, 41(6): 1957-1970.
21. Liu, G., Zhu, Y., Xu, S., et al. (2022b) PSO-Based Power-Driven X-Routing Algorithm in Semiconductor Design for Predictive Intelligence of IoT Applications. Applied Soft Computing, 114: 108114.
22. Liu, N., Pan, J. Sun, C., et al. (2020c) An efficient surrogate-assisted quasi-affine transformation evolutionary algorithm for expensive optimization problems[J]. Knowledge-Based Systems, 209: 106418.
23. Lu, Z., Liu, G., and Wang, S. (2020) Sparse neighbor constrained co-clustering via category consistency learning[J]. Knowledge-Based Systems, 201, 105987.
24. Muhammad Y, Cong P, Lu H, Fan Y (2010) MES development and significant applications in manufacturing -A review," 2010 2nd International Conference on Education Technology and Computer, Shanghai, 2010, pp.97-101, doi: 10.1109/ICETC.2010.5530040.
25. Ostrowski D (2018) Artificial Intelligence with Big Data," 2018 First International Conference on Artificial Intelligence for Industries (AI4I), Laguna Hills, CA, USA, 2018, pp.125-126, doi: 10.1109/AI4I.2018.8665678.
26. Ozdemir R, Koc M (2019) A Quality Control Application on a Smart Factory Prototype

Using Deep Learning Methods," 2019 IEEE 14th International Conference on Computer Sciences and Information Technologies (CSIT), Lviv, Ukraine, 2019, pp. 46-49, doi: 10.1109/STC-CSIT.2019.8929734.

27. Pauli P, Koch A, Berberich J, Kohler P, Allgöwer F (2022) Training Robust Neural Networks Using Lipschitz Bounds. IEEE Control Systems Letters. Volume: 6.
28. Prihatno A, Nurcahyanto H, Jang Y. M (2021) Predictive Maintenance of Relative Humidity Using Random Forest Method. 2021 International Conference on Artificial Intelligence in Information and Communication (ICAIIC).
29. Shen, S., Yang, Y., Liu, X. (2021) Toward data privacy preservation with ciphertext update and key rotation for IoT[J]. Concurrency and Computation: Practice and Experience, e6729.
30. Song H, Huang G, Chauvel F, Xiong Y, Hu Z, Sun Y, Mei H (2011) Supporting runtime software architecture: A bidirectional-transformation-based approach. Journal of Systems and Software, 84(5): 711-723.
31. Thielen N, Werner D, Schmidt K, Seidel R(2020) Reinhardt and J. Franke, "A Machine Learning Based Approach to Detect False Calls in SMT Manufacturing," 2020 43rd International Spring Seminar on Electronics Technology (ISSE), Demanovska Valley, Slovakia, 2020, pp. 1-6, doi: 10.1109/ISSE49702.2020.9121044.
32. Wang, S., Wang, Z., Lim, K. L., et al. (2021) Seeded random walk for multi-view semi-supervised classification [J]. Knowledge-Based Systems, 222: 107016.
33. Yang Y, Guan Z, Wan Z, et al. (2021) PriScore: blockchain-based self-tallying election system supporting score voting[J]. IEEE Transactions on Information Forensics and Security, 16: 4705-4720.
34. Yang Y, Liu X, Deng R. (2018a) Expressive Query over Outsourced Encrypted Data. Information Sciences, 442: 33-53.
35. Yang Y, Liu X, Guo W, et al. (2020) Multimedia access control with secure provenance in fog-cloud computing networks. Multimedia Tools and Applications, 79(15): 10701-10716.
36. Yang Y, Zheng X, Chang V, et al. (2018b) Lattice assumption based fuzzy information retrieval scheme support multi-user for secure multimedia cloud. Multimedia Tools and Applications, 77(8): 9927-9941.
37. Yu Z, Zheng X, Huang F, et al. (2021) A framework based on sparse representation model for time series prediction in smart city[J]. Frontiers of Computer Science, 15(1): 1-13.
38. Zhang Y, Huang G, Liu X, Zhang W, Mei H, Yang S (2012) Refactoring android Java code for on-demand computation offloading. ACM SIGPLAN Conference on Object-Oriented Programming, Systems, Languages, and Applications.
39. Zhang, H., Li, J. L., Liu, X. M., et al. (2021a) Multi-dimensional feature fusion and stacking ensemble mechanism for network intrusion detection[J]. Future Generation Computer Systems, 122: 130-143.
40. Zhang, Y., Lu, Z., and Wang, S. (2021b) Unsupervised feature selection via transformed auto-encoder[J]. Knowledge-Based Systems, 215: 106748.
41. Zheng, X., Rong, C, et al. (2019) Foreword to the special issue of green cloud computing: Methodology and practice[J]. Concurrency and Computation: Practice and Experience. 31(23): e5425.
42. Zhou L, Wang L, Ge X, Shi Q (2010) A clustering-Based KNN improved algorithm CLKNN for text classification," 2010 2nd International Asia Conference on Informatics in Control, Automation and Robotics (CAR 2010), Wuhan, 2010, pp. 212-215, doi: 10.1109/CAR.2010.5456668.
43. Zhou Y, Liu X, Li M (2011) Discussion about Problems and Corresponding Countermeasures in Enterprise Informatization Construction," 2011 International Conference on Business Computing and Global Informatization, Shanghai, 2011, pp. 243-246, doi: 10.1109/BCGIn.2011.69.
44. Zou, W., Guo, L., Huang, P., et al. (2022) Linear time algorithm for computing min-max movement of sink-based mobile sensors for line barrier coverage[J]. Concurrency and

Computation: Practice and Experience, 34(2): e6175.

Dong Chongjie, Master graduated from Guangdong University of Technology in 2010.He is working in Dongguan Polytechnic now. He is an associate professor, and his main research direction is artificial intelligence technology application and database technology.

Zhang Kai received his master's degree from South China University of Technology in 2010. At present, he is a teacher of software technology in Guangdong Communication Polytechnic. His main research interests include artificial intelligence applications, mobile application development, and the application of big data technology.

Received: July 14, 2022; Accepted: December 18, 2022.