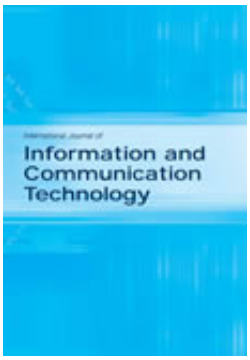


International Journal of Information and Communication Technology

ISSN online: 1741-8070 - ISSN print: 1466-6642
https://www.inderscience.com/ijict

Application of visual data analysis system based on artificial intelligence

Xinyun Cheng, Shijie Zhang, Pengfei Wang, Zhikang Wang, Lincheng Qi

DOI: 10.1504/IJICT.2025.10074751

Article History:

Received:	18 July 2025
Last revised:	01 September 2025
Accepted:	09 September 2025
Published online:	12 December 2025

Application of visual data analysis system based on artificial intelligence

Xinyun Cheng

Information Transportation Inspection Center,
State Grid Jiangsu Electric Power Company,
Information and Telecommunication Branch,
Nanjing, 210024, Jiangsu, China
Email: yukicccy@163.com

Shijie Zhang*

Jiangsu Electric Power Information Technology Co., Ltd.,
Nanjing, 210024, Jiangsu, China
Email: muyachuju@126.com
*Corresponding author

Pengfei Wang and Zhikang Wang

Information Transportation Inspection Center,
State Grid Jiangsu Electric Power Company,
Information and Telecommunication Branch,
Nanjing, 210024, Jiangsu, China
Email: gaosudongfang@163.com
Email: nwpuwangzhikang@163.com

Lincheng Qi

Jiangsu Electric Power Information Technology Co., Ltd.,
Nanjing, 210024, Jiangsu, China
Email: miaomiaoguoke@gmail.com

Abstract: For the insufficiency of traditional systems in automated data processing and predictive analysis capability, this study explored a visual data analysis system based on advanced artificial intelligence technology, integrating the three core functions of automated data preparation, intelligent recommendation and predictive analysis. Data cleaning was carried out by weighted k-nearest neighbours imputation and isolation forest algorithm. The unstructured data was handled utilising bidirectional encoder representations from transformers (BERT) models, and key patterns, trends and anomalies were discovered by means of association rule learning techniques. Relying on the autoregressive integrated moving average (ARIMA) model, the time series data was precisely forecasted. Distributed deployment supports the hardware and solves the system layout problem. The evaluation outcomes demonstrated that the ARIMA model performed the best in data prediction with an average prediction time of only 1.075 seconds, the lowest RMSE (7.19) and MAE

(4.70), and the highest prediction accuracy (96.00%). This paper provides efficient and intelligent data support and solutions to help decision-making and strategic planning in various industries.

Keywords: visual data analysis system; artificial intelligence technology; distributed deployment; predictive analysis; ARIMA model; data automation processing.

Reference to this paper should be made as follows: Cheng, X., Zhang, S., Wang, P., Wang, Z. and Qi, L. (2025) 'Application of visual data analysis system based on artificial intelligence', *Int. J. Information and Communication Technology*, Vol. 26, No. 43, pp.1–19.

Biographical notes: Xinyun Cheng received her Master's degree from Southeast University, China. Currently, she works in State Grid Jiangsu Electric Power Company Information and Telecommunication Branch. Her research interests include cloud security, chaos encryption and information security.

Shijie Zhang received his Master's degree from Nanjing University. Currently, he works in Jiangsu Electric Power Information Technology Co., Ltd. His research interests include electricity informatisation and power system database.

Pengfei Wang received his Master's degree from Nanjing University of Science and Technology, China. Currently, he works in State Grid Jiangsu Electric Power Company Information and Telecommunication Branch. His research interests include cloud security, chaos encryption and information security.

Zhikang Wang received his Bachelor's degree from Beihang University, China. Currently, he works in State Grid Jiangsu Electric Power Company Information and Telecommunication Branch. His research interests include cloud-native ops, cloud security and machine learning.

Lincheng Qi received his Bachelor's degree from Hohai University. Currently, he works in Jiangsu Electric Power Information Technology Co., Ltd. His research interests include electricity informatisation and power system database.

1 Introduction

As the big data era (Zhenpeng, 2024; Luo, 2023) approaches, the amount of data generated by various industries is rapidly increasing, including structured data (Xu et al., 2024), semi-structured data (Leshcheva and Begler, 2022) and unstructured data (Puja et al., 2024). Traditional data analysis systems (Bai, 2022; Bansal et al., 2022) require a lot of manual intervention, especially in the data cleaning and preprocessing stages, which not only consume human resources, but also tend to bring in errors, affecting the quality of data and the accuracy of analysis results. The lack of effective tools and techniques for handling unstructured data in traditional systems has led to insufficient exploration of its potential value. Besides, traditional predictive analytics methods (Cao et al., 2024; Chien et al., 2020) rely on statistical models, thus making it difficult to deal

with complex data relationships and multidimensional features, and limiting predictive accuracy and usefulness, especially in dynamic environments.

To address the above issues, this paper presents a visual data analysis system based on advanced artificial intelligence techniques (Fotouhi et al., 2021; Peres et al., 2021), which is capable of handling unstructured data and has strong predictive analysis capabilities. Combined with machine learning (Mohammed et al., 2020; Schlosser et al., 2019) and natural language processing (NLP) techniques, automated data processing has been realised to improve the efficiency and quality of data processing. To enhance the capability of predictive analysis, the system utilises an ARIMA model (Gui et al., 2021; Gu et al., 2022) to capture complex relationships and trends in the data to improve the accuracy and usefulness of the predictions.

2 Related work

Recent research has indicated that data analysis techniques are becoming increasingly popular for optimising decision making and management in various fields. Li (2024a) constructed a data analysis and decision support system for animal husbandry operations on a cloud computing platform, which improved the efficiency of operations and the accuracy of decision making. Zhao (2024) emphasised the key role of data analysis in new energy investment decision, covering market analysis, risk assessment and decision optimisation. Li's (2024b) research proposed solutions to the problem of hospital archive management, effectively improving the application value of archive data. Yan (2024a) explored the innovative application of data analysis in audit work, which helped improve audit efficiency and the ability to discover financial data problems. Yan's (2024b) research focused on the core application of data analysis in procurement cost control in manufacturing enterprises. By optimising supplier evaluation, procurement volume prediction, and inventory strategies, procurement costs were effectively reduced. These studies collectively demonstrate the widespread application and importance of data analysis techniques in multiple fields.

With the development of modern technology, it has become an obvious trend that artificial intelligence technology is widely used in various fields. Taking the Python-Flask framework-based safety information analysis and decision-making assistance system for train operation depots proposed by Liu et al. (2022) as an example, this system not only improved decision-making efficiency but also significantly enhanced the ability to manage safety risks. Yang et al. (2021) combined visualisation technology and machine learning to successfully develop a dynamic environment management system for data centres based on digital twins, achieving real-time monitoring, anomaly analysis, and predictive inference functions. He et al. (2024) pointed out in their review of the field of dental medicine that the application of big data and artificial intelligence has greatly improved the diagnostic precision and treatment effect of dental medicine, providing important support for the development of intelligent dental medicine. In the field of energy, the study of Zhang and Lv (2024) explored the operation data analysis and optimisation methods of thermal power plants based on machine learning, demonstrating the great potential of artificial intelligence technology in optimising energy production efficiency. In summary, the visual data analysis system based on artificial intelligence has a wide range of application prospects. This paper integrates data

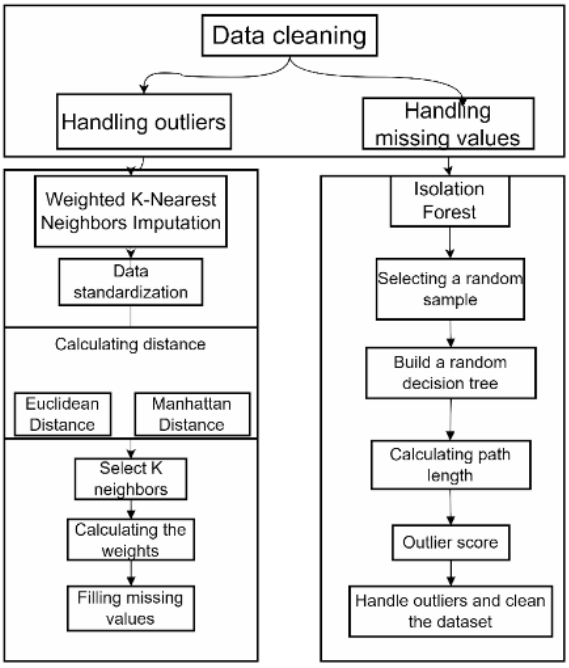
analysis, machine learning and visualisation techniques to provide intelligent decision support and management optimisation tools for various industries.

3 System design and implementation

3.1 Data cleaning

Data cleaning is one of the crucial steps in the data analysis process, aimed at handling missing and outliers in the data, ensuring the quality and accuracy of the data, and providing a reliable foundation for subsequent analysis and modelling. The artificial intelligence-based visual data analysis system introduced in this paper adopts two methods, weighted K-nearest neighbours imputation (Zheng et al., 2021) and isolation forest algorithm (Wei et al., 2024), to address data quality issues during the data cleaning stage, as shown in Figure 1.

Figure 1 Data cleaning process diagram



3.1.1 Weighted K-nearest neighbours imputation

In a visual data analysis system based on artificial intelligence, weighted K-nearest neighbours imputation automatically fills in missing values by considering the distance between adjacent data points. Unlike ordinary K-nearest neighbours imputation, weighted K-nearest neighbours imputation assigns higher weights to neighbours who are closer, thereby improving the precision of filling. The specific implementation process is as follows: the system selects an appropriate K value based on the size and characteristics of the dataset, that is, the number of adjacent data points, as shown in Table 1. Through

cross validation, the system selects the optimal K value that balances computational efficiency and filling accuracy. When facing spatial coordinate data, image data, and numerical data, the system selects Euclidean distance; when facing text data, network data, urban planning, and path planning systems, Manhattan distance is chosen to measure the distance between data points (Temple, 2023; Pambudi et al., 2018). For each data point containing missing values, the system calculates its distance from all other data points. According to the calculated distance, the nearest K neighbour data points are found for each missing value point. For each neighbour data point, the weight is calculated based on the distance between it and the missing value point, and the weight is the inverse of the distance.

Table 1 K-value selection criteria

Dataset type	*Sample size*	*Feature count*	*Recommended K range*
Small-scale dataset	Less than 1000	Less than 10	3 to 5
Small-scale dataset	Less than 1000	10 to 100	3 to 5
Medium-scale dataset	More than 10,000	Less than 10	5 to 10
Medium-scale dataset	1,000 to 10,000	10 to 100	5 to 10
Large-scale dataset	More than 10,000	More than 100	10 to 20

The weight formula:

$$Wi = \frac{1}{di + \varepsilon} \tag{1}$$

3.1.2 Isolation forest algorithm

The isolation forest algorithm identifies outliers in the data by constructing multiple isolated trees. The algorithm randomly selects subsamples to construct multiple isolated trees, calculates the average path length of each data point in the tree, and generates a normalised anomaly score. Data points with an anomaly score above a threshold are marked as anomalous and removed from the dataset.

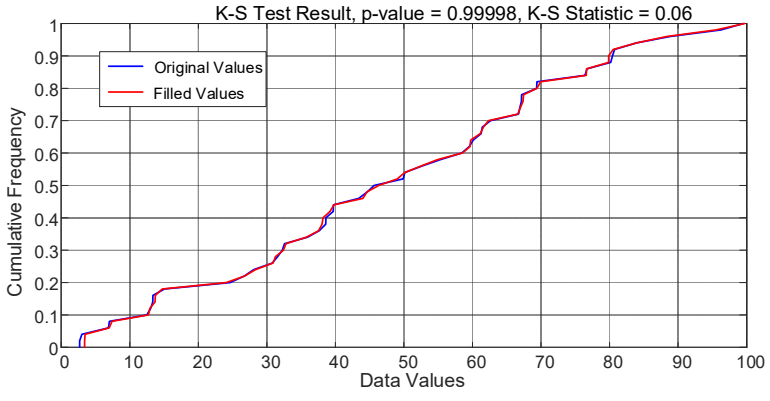
Abnormal score calculation formula:

$$s(x, n) = 2^{-\frac{E(h(x))}{c(n)}} \tag{2}$$

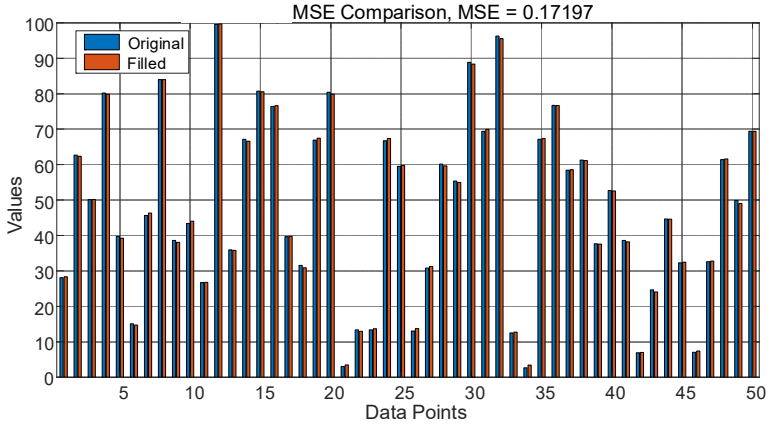
3.1.3 Evaluation of data cleaning effectiveness

The mean square error (MSE) and the similarity of data distribution before and after imputation are used to evaluate the imputation effect. The mean square error is used to measure the difference between the imputed data and the true value, with smaller MSE indicating better imputation. In this paper, some of the data in dataset A are randomly deleted and the missing values are filled in using the weighted K-nearest neighbour imputation algorithm. Then, the MSE between the filled values and the original values is calculated. In addition, to determine whether the imputation process preserves the original distributional characteristics of the data, the Kolmogorov-Smirnov test (K-S test) (Yamaguchi and Saito, 2022) compares the data distributions before and after the imputation (see Figure 2).

Figure 2 Missing value filling evaluation, (a) K-S test chart (b) MSE comparison chart (see online version for colours)



(a)



(b)

As shown in Figure 2(b), the filled values on the data points are very close to the original values, with an MSE value of 0.17197, indicating that the numerical difference between the filled values and the original values is small and the filling effect is good. This indicates that the weighted K-nearest neighbours imputation algorithm can accurately recover missing values in most cases. The K-S test in Figure 2(a) shows that the distribution of the original data and the filled data is very close. The p-value of the K-S test result is 0.99998, and the K-S statistic is 0.06, further confirming that there is no significant difference in the distribution of the two groups of data. Overall, the weighted K-nearest neighbours imputation algorithm performs well in handling missing values, not only being very close to the original data in numerical terms, but also maintaining consistency in data distribution characteristics. It can effectively preserve the integrity and accuracy of data and is suitable for practical data analysis and processing tasks.

In terms of the effectiveness of outlier detection, accuracy, precision, and recall are used as evaluation indicators. The detection accuracy measures the proportion of detected outliers that are actually outliers. The experimental method involves randomly replacing some normal values in the dataset with outliers, resulting in five different datasets. The

isolation forest algorithm is used to identify outliers and calculate accuracy, precision, recall, and F1-score. Table 2 shows the results of the calculations. Overall, the detection performance of the isolation forest algorithm is very stable and excellent on different datasets. The accuracy is generally higher than 0.95, and both the precision and recall are above 0.90, indicating that the algorithm can correctly distinguish between outliers and normal values in anomaly detection tasks.

Table 2 Evaluation table for abnormal value detection

Dataset	*Normal values*	*Anomalous values*	*Detected anomalous values*	*Accuracy*	*Precision*	*Recall*	*F1-score*
Dataset A	800	200	190	0.97	0.95	0.9	0.924
Dataset B	900	100	95	0.985	0.947	0.9	0.923
Dataset C	700	300	288	0.952	0.938	0.9	0.919
Dataset D	880	120	117	0.983	0.94	0.917	0.928
Dataset E	600	400	395	0.965	0.962	0.95	0.956

Accuracy measures the proportion of correctly predicted samples to the total number of predicted samples.

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

Precision represents the proportion of detected outliers that are actually outliers.

$$\text{Precision} = \frac{TP}{TP + FP} \tag{4}$$

Recall represents the proportion of actual outliers correctly detected as anomalies.

$$\text{Recall} = \frac{TP}{TP + FN} \tag{5}$$

The F1-value is the harmonic mean of precision and recall:

$$F1 = 2 \cdot \frac{\text{Precision} \cdot \text{Recall}}{\text{Precision} + \text{Recall}} \tag{6}$$

3.2 Data conversion

In the modern data-driven decision-making environment, a large amount of data is constantly emerging, including a large amount of unstructured natural language data, such as social media content, user comments, and product descriptions. Although these data contain rich information, traditional data analysis and modelling methods often cannot directly process and utilise them due to their unstructured nature. The main goal of data transformation is to transform these unstructured natural language data into structured data form for further data analysis, pattern recognition, and predictive modelling. This study uses advanced natural language processing techniques and utilises the BERT model (Zhao et al., 2024; Wang et al., 2024) as the core tool for data

conversion. The BERT model can more accurately capture the complex relationships of vocabulary and the semantics of context in language. Through the application of the BERT model, this system can effectively convert unstructured data contained in text or speech into computer-processed structured data, providing a solid foundation for subsequent data analysis and model construction.

In order to achieve the data conversion function of the BERT model in practical applications, this system adopts the preprocessed BERT model provided by Hugging Face Company, and uses its provided dataset for model training and optimisation. The dataset is divided into 80% training set and 20% validation set. Before training the model, it is necessary to preprocess the dataset text data using tokeniser, including word segmentation, encoding, and generating the model input format. The trainer class in the transformers library provides a way to simplify the training process. Firstly, it is necessary to define training parameters, then create a trainer object and start training. The training parameters specify important parameters such as output directory, evaluation strategy, batch size, number of training cycles, and weight decay. Through the trainer class, the system can efficiently train models under specified parameters. At the end of each training cycle, the results returned by the evaluation function are analysed. These results include the precision, recall, and F1-score of the model on the validation set. According to the evaluation results, the model is optimised to improve its performance by adjusting the learning rate. After tuning the model, it is retrained and evaluated again. Through multiple iterations, the performance of the model is gradually optimised. Each evaluation result should be recorded and compared with the previous one to verify the effectiveness of the tuning.

Figure 3 Model learning rate, precision, recall, F1-score (see online version for colours)

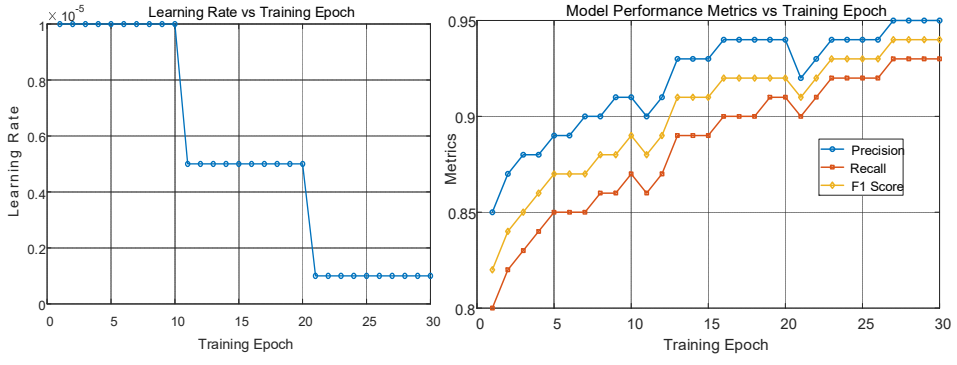


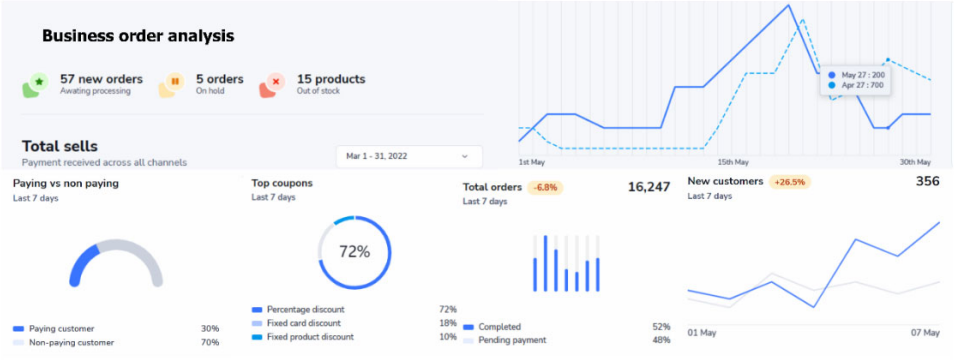
Figure 3 shows the changes in learning rate and the performance of the model on various evaluation metrics during different training cycles. In the first 10 cycles, the learning rate is high and the model parameters are updated frequently, resulting in a rapid increase in precision, recall, and F1-score. The precision rate steadily increases from 0.85 to 0.91; the recall rate increases from 0.80 to 0.87; the F1-score increases from 0.82 to 0.89. However, in order to avoid overfitting and ensure better generalisation of the model, the learning rate gradually decreases after the tenth cycle, making the model more stable and convergent while ensuring higher performance. Finally, after the learning rate decreases to $1 * 10^{-6}$, the overall precision remains between 0.92–0.95; the recall rate remains between 0.90–0.93; the F1-score remains between 0.91–0.94. The various evaluation

indicators of the model have reached the optimal level with small fluctuations, indicating that the model has undergone sufficient training and achieved a high level of performance.

3.3 Chart recommendations

The rule-based recommendation system infers the most suitable type of data chart based on the properties of the data and the analysis goal through a predefined series of rules. The system thoroughly analyses the input data in the detailed data attribute identification phase, including data type, data distribution, and data dimensions. The rule base consists of a series of if-then rules that recommend the corresponding chart type based on specific conditions: line or area charts for time series data, and bar or pie charts for categorical data. In the rule matching phase, the system matches the recognised data attributes with the conditions in the rule base, and checks the conditions in the rule base one by one to determine whether the input data satisfy these conditions through the condition detection, multi-condition processing and conflict resolution mechanisms. When the input data satisfies several rules at the same time, the system selects the most appropriate chart type through the preset priority ranking or comprehensive scoring mechanism, and finally recommends one or more optimal chart options to the user, as shown in Figure 4.

Figure 4 Rendering (see online version for colours)

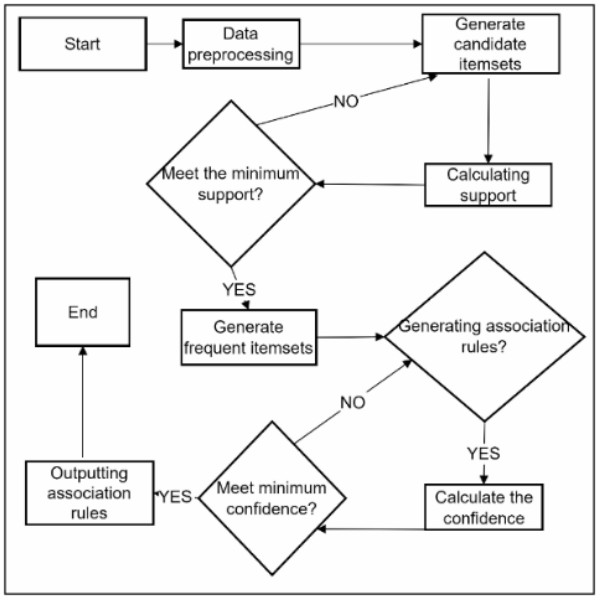


3.4 Insight discovery

In visual data analysis systems, insight discovery provides deep insights for decision makers by mining patterns, trends, and correlations in the data. Apriori algorithm (Song and He, 2023) is a classical frequent itemset and association rule mining algorithm, and its operation process is shown in Figure 5. After preprocessing the raw data to ensure consistent data quality and format, the algorithm iteratively generates candidate itemsets (C1) for all individual items, calculates the support of each itemset, and filters out frequent itemsets (L1) based on the minimum support threshold. Then, the frequent itemsets are self-connected to generate larger candidate itemsets, and support and pruning are continued to be calculated until new frequent itemsets cannot be generated. After finding all frequent itemsets, the algorithm generates association rules and calculates confidence. By setting a confidence threshold, high-quality association rules are filtered

out, and further metrics such as improvement are calculated to evaluate the usefulness of the rules.

Figure 5 Apriori algorithm flowchart



Support is used to measure the frequency of an itemset appearing in the database. Among them, Count(A) is the number of times itemset A appears in the dataset, and N is the total number of transactions.

$$\text{Support}(A) = \frac{\text{Count}(A)}{N} \tag{7}$$

Confidence is used to measure the probability of itemset B occurring simultaneously under the premise of itemset A. Support ($A \cup B$) is the support for the simultaneous occurrence of itemsets A and B.

$$\text{Confidence}(A \to B) = \frac{\text{Support}(A \cup B)}{\text{Support}(A)} \tag{8}$$

Lift is used to measure the improvement effect of the association rules between itemset A and itemset B compared to the individual occurrence of itemset B. For association rules, their degree of improvement is defined as:

$$\text{Lift}(A \to B) = \frac{\text{Confidence}(A \to B)}{\text{Support}(B)} \tag{9}$$

3.5 Distributed deployment

The virtual NE generation process is deployed, and the deployment module analyses the topology description file, and then generates it according to the information in it,

command to deploy virtual NE into a separate shell file, where the algorithm is generated the docker run command is used to deploy all virtual NEs, and the corresponding image in section 3.1.2 of this article is called to generate container instances.

Virtual link generation adopts signal deployment to generate links, and the algorithm selects the method of generating links based on the link information in the topology profile. There are two ways to generate virtual links in the deployment module, one is to use the Veth-pair interface connection between virtual network elements, use the transformer bidirectional encoder representation (BERT) model to process unstructured data, and discover key patterns, trends, and anomalies through associative rule learning technology. Realise the internal communication between virtual network elements on the same host, and assist the cross-host module in the network element algorithm to realise cross-host communication between virtual network blocks, so as to realise the connectivity of the entire virtual network after distributed deployment to the host. Weighted K-neighbour interpolation and isolated forest algorithm are used for data cleaning. Based on the autoregressive comprehensive moving average (ARIMA) model, the time series data are accurately predicted.

Configure the virtual network element after the deployment of the virtual network element and link to the host machine, all network element nodes need to be configured according to the information carried by the network element, including the IP address, subnet mask, gateway, etc. of the network element.

3.6 Time series prediction

Time series prediction is one of the important functions of this system. By analysing the trends and patterns of historical data, it provides predictions for future development and provides strong reference basis for decision makers. This system mainly uses ARIMA model for time series data prediction, and the dataset used is from the order data of a gift wholesale e-commerce platform in the Tianchi dataset within one year. Firstly, the system uses a data cleaning module to preprocess time series data, ensuring data integrity and consistency. After the data preprocessing is completed, the system performs stationarity detection and differential processing on the time series data. The augmented Dickey-Fuller (ADF) test is used to determine if the data is stationary.

The ADF test statistic formula is:

$$\mathrm{ADF} = \frac{\hat{\alpha}}{\mathrm{SE}(\hat{\alpha})} \tag{10}$$

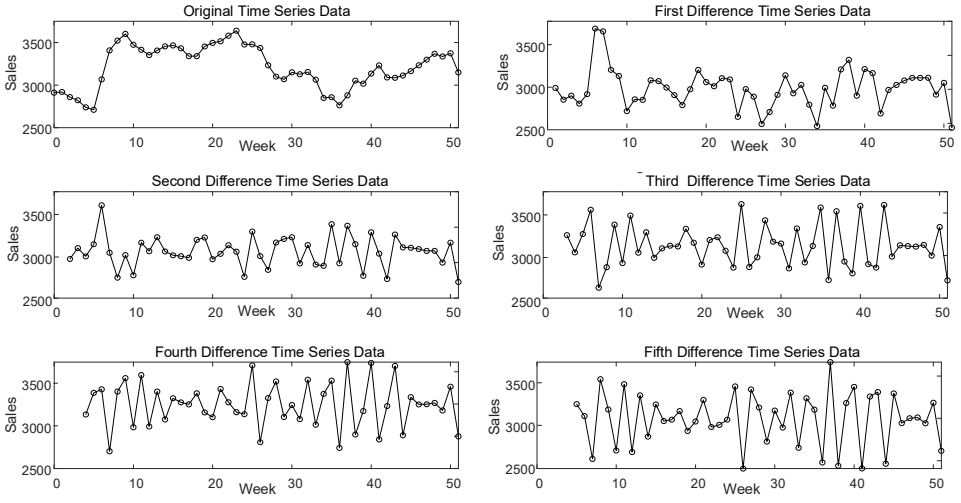
Among them, $\hat{\alpha}$ is the estimated value of the autoregressive coefficient in the ADF regression model, and $\mathrm{SE}(\hat{\alpha})$ is its standard error. If the data is non-stationary, the system performs differential processing until the data is stable. Differential processing eliminates trends and seasonal components by performing one or more differential operations on data. The formula for a differential operation is:

$$Y_t = Y_t - Y_{t-1} \tag{11}$$

For d differential operations, the formula is:

$$\Delta^d Y_t = \Delta^{d-1} Y_t - \Delta^{d-1} Y_{t-1} \tag{12}$$

Figure 6 Time series data and its difference plot



By analysing the original time series data graph in Figure 6 and the time series data graph after one to five differencing, the impact of different differencing levels on the data can be observed. The original time series data shows the overall trend and random fluctuations of sales over weeks. A differential processing eliminates long-term trends while retaining the characteristics of short-term fluctuations. The quadratic difference further removes linear trends and highlights the seasonal and periodic components of the data. As the number of differencing increases, such as three to five differencing, the fluctuation of the data becomes more severe; the noise component significantly increases; the identification of useful information becomes difficult. Compared to the third to fifth difference, the second difference not only stabilises the data but also preserves sufficient information.

Next, the system identifies the order (p, d, q) of the ARIMA model through the autocorrelation function (ACF) and partial autocorrelation function (PACF). Among them, p is the autoregressive order; d is the order of difference; q is the moving average order. Due to the best performance of quadratic differentiation, which not only stabilises the data but also preserves sufficient information, the order of differentiation d should be 2. By observing the truncation and trailing features of ACF and PACF graphs, appropriate parameter values p and q are determined. The standard formula for the ARIMA model is:

$$Y_t = c + \phi_1 Y_{t-1} + \phi_2 Y_{t-2} + \cdots + \phi_p Y_{t-p} + \theta_1 \varepsilon_{t-1} + \theta_2 \varepsilon_{t-2} + \cdots + \theta_q \varepsilon_{t-q} + \varepsilon_t \tag{13}$$

Among them, Y_t is the actual value of time t; c is a constant term; ϕ is an autoregressive parameter; θ is the moving average parameter; ε_t is the white noise error term; p is the autoregressive order; q is the moving average order.

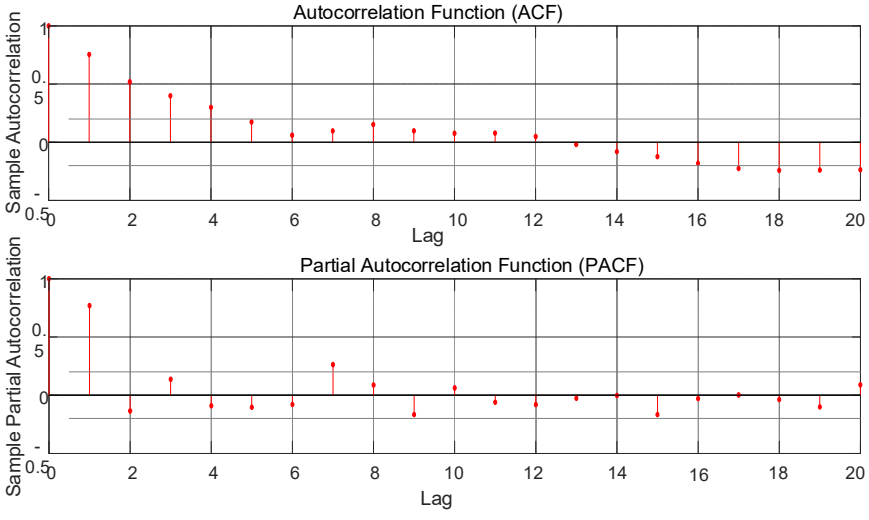
ACF is used to describe the correlation between observations in a time series. The ACF function is defined as the correlation coefficient between the time series Y_t and its own lagged k-period:

$$\mathrm{ACF}(k) = \frac{\mathrm{Cov}\left(Y_t, Y_{t-k}\right)}{\sigma_Y^2} \tag{14}$$

PACF measures the pure correlation between observed values in a time series, excluding other lag effects. Its definition is the k-order partial autocorrelation coefficient of time series Y_t:

$$\phi_k = \mathrm{Corr}\left(Y_t, Y_{t-k} \middle| Y_{t-1}, Y_{t-2}, \ldots, Y_{t-k+1}\right) \tag{15}$$

Figure 7 ACF and PACF diagrams (see online version for colours)



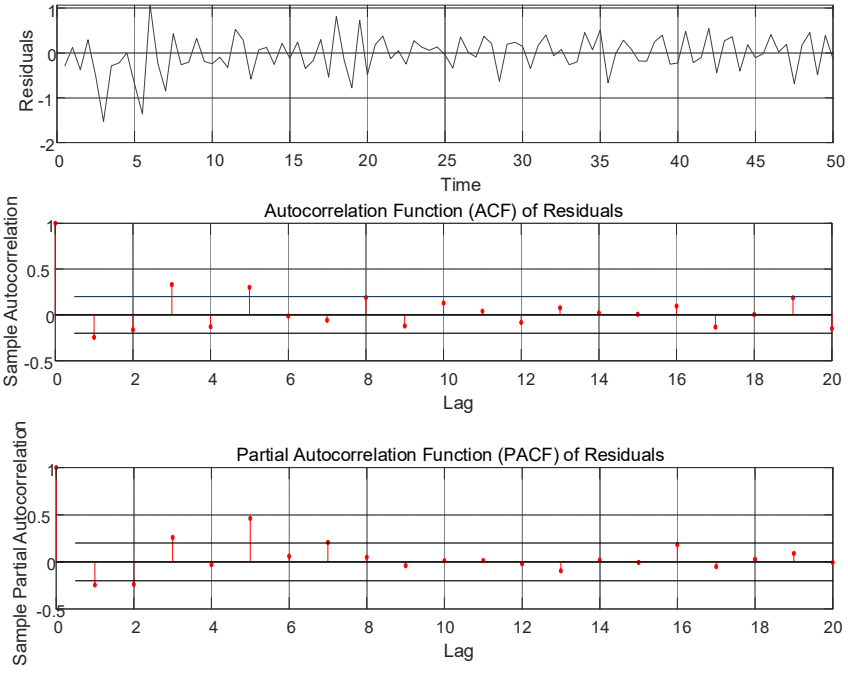
The obtained ACF and PACF plots are shown in Figure 7. The ACF plot shows an autocorrelation coefficient of approximately 0.7 for lag 1, indicating a significant positive correlation between the current observation and the previous observation. As the lag period increases, the autocorrelation coefficient gradually decreases and begins to truncate at lag periods of 3 or 4. The order p of the autoregressive model may be 3 or 4. The PACF plot shows a partial autocorrelation coefficient of approximately 0.7 for hysteresis period 1, with a moving average order q of 1.

After the parameter estimation is completed, the system verifies the model through residual analysis and Ljung-Box test. Residual analysis verifies the fitting effect of the model. Ideally, the residual should appear as white noise, meaning that the residual sequence has no autocorrelation and the mean is zero. The system checks whether the residuals meet the white noise characteristics by calculating the ACF and PACF plots of the residuals. In addition, the system uses Ljung-Box test to further verify the white noise properties of the residuals. If the test results show that the residual is white noise, it indicates that the model fits well; otherwise, the system needs to adjust the model parameters or re-identify the model.

Based on the residual analysis in Figure 8, the autocorrelation coefficient $\hat{\rho}_k$ of the residual sequence at different lag periods k is calculated, and then the Ljung-Box statistic Q and p-values are calculated. If the p-value is greater than 0.05, the residual can be

considered as white noise. The calculated p-value is 0.18284, which is greater than 0.05. The test results show that the residual is white noise, indicating a good fit of the model.

Figure 8 Residual analysis (see online version for colours)



Autocorrelation coefficient $\hat{\rho}_k$:

$$\rho_k = \frac{\sum_{t=k+1}^{T}(x_t - \bar{x})(x_{t-k} - \bar{x})}{\sum_{t=1}^{T}(x_t - \bar{x})^2} \tag{16}$$

The calculation formulas for Ljung-Box statistic are:

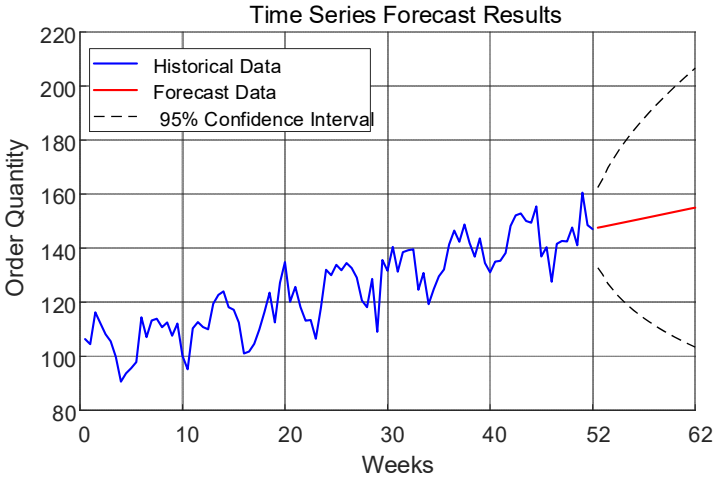
$$Q = T(T+2)\sum_{k=1}^{h}\frac{\hat{\rho}_k^2}{T-k} \tag{17}$$

$$p = 1 - \chi^2_{c,dof}(Q) \tag{18}$$

After the model validation is passed, the system uses the ARIMA model to predict future time points. The system generates predicted values and confidence intervals for future time points based on historical data and model parameters. In order to facilitate user understanding and analysis of the prediction results, the system visualises the prediction results. Figure 9 shows the historical order data for one year (blue line) and the forecast results for the next 10 weeks (red line), as well as the 95% confidence interval of the forecast results (black dashed line). From Figure 9, it can be visually seen the trend of historical data and the predicted values and uncertainty range of future data. By

predicting future sales trends, enterprises can adjust their supply chain and market strategies reasonably, and improve operational efficiency and decision-making quality.

Figure 9 ARIMA model prediction (see online version for colours)



4 Performance evaluation

This section compares the time performance and accuracy analysis of the ARIMA model with five mainstream prediction models: exponential smoothing (ETS), temporal convolutional network (TCN), Prophet, long short-term memory (LSTM), and gradient boosting machine (GBM), to evaluate the advantages and disadvantages of ARIMA model in prediction. The dataset used is from the order data of a gift wholesale e-commerce platform in the Tianchi dataset within a year, which includes the weekly sales quantity and has obvious seasonal and trend characteristics, making it suitable for comparative analysis of time series prediction models. This paper conducts cross validation on data, calculates the time and accuracy indicators of each model, and conducts comprehensive comparisons.

According to the data in Table 3, the ARIMA model performs well in terms of time performance compared to the ETS model, TCN model, Prophet model, LSTM model, and GBM model, with an average training time of 96.25 seconds and an average prediction time of only 1.075 seconds, far lower than other comparative models. From 2019 to 2022, the training time of the ARIMA model decreased from 100 to 92, a decrease of 8%, and the prediction time increased from 1 to 1.2, an increase of 20%; from 2019 to 2022, the ETS model decreased from 150 to 142, a decrease of 5.3%, and the prediction time increased from 2 to 2.2, an increase of 10%; from 2019 to 2022, the training time decreased from 140 to 132, a decrease of 5.7%, and the prediction time increased from 1.8 to 2, an increase of 11.1%; from 2019 to 2022, the training time decreased from 130 to 122, a decrease of 6.1%, and the prediction time increased from 1.5 to 1.7, an increase of 13.3%; from 2019 to 2022, the training time of the LSTM model decreased by 5% from 160 to 152, and the prediction time increased from 2.5 to 2.7, an increase of 8.1%;

from 2019 to 2022, the GBM model saw a decrease of 4.8% in training time from 145 to 138, and a decrease of 13.3% in prediction time from 1.5 to 1.7.

Table 3 Model time performance evaluation (seconds)

Model		*ARIMA*	*ETS*	*TCN*	*Prophet*	*LSTM*	*GBM*
2019	Training time	100	150	140	130	160	145
	Prediction time	1	2	1.8	1.5	2.5	1.5
2020	Training time	95	145	135	125	155	140
	Prediction time	1.1	2.1	1.9	1.6	2.6	1.6
2021	Training time	98	148	138	128	158	143
	Prediction time	1	2	1.8	1.5	2.5	1.5
2022	Training time	92	142	132	122	152	138
	Prediction time	1.2	2.2	2	1.7	2.7	1.7
Average training time		96.25	146.25	136.25	126.25	156.25	141.5
Average prediction time		1.075	2.075	1.875	1.575	2.575	1.575

The evaluation of prediction accuracy is mainly based on indicators such as root mean square error (RMSE), mean absolute error (MAE), mean absolute percentage error (MAPE), and prediction accuracy.

The root mean square error measures the standard deviation of the error between the predicted and actual observed values:

$$\mathrm{RMSE} = \sqrt{\frac{1}{n}\sum_{i=1}^{n}\left(y_i - \hat{y}_i\right)^2} \tag{19}$$

The mean absolute error is the average absolute error between the predicted value and the actual observed value:

$$\mathrm{MAE} = \frac{1}{n}\sum_{i=1}^{n}\left|y_i - \hat{y}_i\right| \tag{20}$$

The mean absolute percentage error is the average percentage error relative to the actual observed value:

$$\mathrm{MAPE} = \frac{1}{n}\sum_{i=1}^{n}\left(\frac{\left|y_i - \hat{y}_i\right|}{y_i}\right)\times 100\% \tag{21}$$

Table 4 Accuracy evaluation

Model	*RMSE*	*MAE*	*MAPE*	*Prediction accuracy*
ARIMA	7.19	4.70	2.10%	96.00%
ETS	8.92	5.97	2.99%	91.07%
TCN	8.78	5.78	2.73%	92.06%
Prophet	8.14	5.47	2.65%	93.02%
LSTM	10.96	7.47	3.74%	88.04%
GBM	9.45	6.45	3.11%	90.03%

According to the data in Table 4, the ARIMA model demonstrates excellent prediction accuracy, with the lowest root mean square error and mean absolute error of 7.19 and 4.70, respectively, while exhibiting a mean absolute percentage error of 2.10% and a prediction accuracy of up to 96.00%. Although the Prophet model is slightly inferior to ARIMA, it still demonstrates good MAPE (2.65%) and prediction accuracy (93.02%). In contrast, the LSTM model has a larger prediction error (RMSE exceeding 10.0). In summary, the ARIMA model performs well in handling order data with significant seasonality, making it a stable and effective choice.

5 Conclusions

This study introduces a visual data analysis system based on artificial intelligence, aiming to enhance the capabilities of traditional systems in data automation processing and predictive analysis. The system integrates three major modules: automated data preparation, intelligent recommendation, and predictive analysis. The weighted K-nearest neighbours imputation and isolation forest algorithms are used for data cleaning, and the unstructured data is converted into structured data through the BERT model. Rule-based recommendation algorithms automatically select the most suitable chart type, while association rule learning algorithms mine relationships and patterns in data. The ARIMA model is adopted for predicting time series data, achieving precise trend prediction through systematic preprocessing, model construction, validation, and prediction, providing data support for decision making and strategic planning. Although the system has improved data processing efficiency and analysis accuracy, there are still problems such as high computational resource consumption and insufficient real-time performance when processing large-scale data and complex models. Distributed deployment is a method of dispersing data and applications to be stored and run on multiple independent machines. It provides strong scalability and fault tolerance for software and support, can maximise the use of hardware resources, avoid the situation that some machines are overloaded and other machines are idle, and distributed deployment makes management and control more decentralised, is more suitable for distributed management and control. Future research can further optimise algorithm performance, enhance real-time processing capabilities of the system, expand application scenarios, and integrate more advanced machine learning and deep learning technologies, providing intelligent data analysis support for more industries.

Declarations

All authors declare that they have no conflicts of interest.

References

Bai, J. (2022) 'Design and implementation of data analysis system of social network', *International Journal of Frontiers in Sociology*, Vol. 4, No. 2.

Bansal, M.A., Sharma, D.R. and Kathuria, D.M. (2022) 'A systematic review on data scarcity problem in deep learning: solution and applications', *ACM Computing Surveys (CSUR)*, Vol. 54, No. 10s, pp.1–29.

Cao, H., Tan, C., Gao, Z., Xu, Y., Chen, G., Heng, P.A. and Li, S.Z. (2024) 'A survey on generative diffusion models', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 36, No. 7, pp.2814–2830.

Chien, J.C., Wu, M.T. and Lee, J.D. (2020) 'Inspection and classification of semiconductor wafer surface defects using CNN deep learning networks', *Applied Sciences*, Vol. 10, No. 15, p.5340.

Fotouhi, S., Pashmforoush, F., Bodaghi, M. and Fotouhi, M. (2021) 'Autonomous damage recognition in visual inspection of laminated composite structures using deep learning', *Composite Structures*, Vol. 268, p.113960.

Gu, R., Shi, J., Chen, X. et al. (2022) 'Octopus-DF: Unified DataFrame-based cross-platform data analytic system', *Parallel Computing*, Vol. 110, p.102879.

Gui, J., Sun, Z.,Wen, Y., Tao, D. and Ye, J. (2021) 'A review on generative adversarial networks: algorithms, theory, and applications', *IEEE Transactions on Knowledge and Data Engineering*, Vol. 35, No. 4, pp.3313–3332.

He, Z., Zhang, Q., Liu, Q. et al. (2024) 'Intelligent analysis of big data in dentistry', *Chinese Journal of Medicine and Clinical Sciences*, Vol. 24, No. 4, pp.264–269.

Leshcheva, I. and Begler, A. (2022) 'A method of semi-automated ontology population from multiple semi-structured data sources', *Journal of Information Science*, Vol. 48, No. 2, pp.223–236.

Li, C. (2024a) 'A data analysis and decision support system for animal husbandry management based on cloud computing platform', *Technology Innovation and Application*, Vol. 14, No. 16, pp.24–27, DOI: 10.19981/j.CN23-1581/G3.2024.16.006.

Li, L. (2024b) 'Research on the digital transformation and information security of hospital archives management', *Lantai Neiwai*, Vol. 2024, No. 14, pp.34–36.

Liu, Z., Liao, J. and Zhong, X. (2022) 'Research on safety information analysis and auxiliary decision system for train depot', *Railway Transport and Economy*, Vol. 44, No. 2, pp.45–51, DOI: 10.16668/j.cnki.issn.1003-1421.2022.02.07.

Luo, L. (2023) 'Research on artificial intelligence applications based on data mining algorithms in the era of big data', *Journal of Artificial Intelligence Practice*, Vol. 6, No. 1.

Mohammed, R., Rawashdeh, J. and Abdullah, M. (2020) 'Machine learning with oversampling and undersampling techniques: overview study and experimental results', *2020 11th International Conference on Information and Communication Systems (ICICS)*, IEEE, pp.243–248.

Pambudi, E.A., Andono, P.N. and Pramunendar, R.A. (2018) 'Image segmentation analysis based on K-means PSO by using three distance measures', *ICTACT Journal on Image and Video Processing*, Vol. 9, No. 1, pp.1821–1826.

Peres, R.S., Guedes, M., Miranda, F. and Barata, J. (2021) 'Simulationbased data augmentation for the quality inspection of structural adhesive with deep learning', *IEEE Access*, Vol. 9, pp.76532–76541.

Puja, A.R., Jewel, R.M., Chowdhury, M.S. et al. (2024) 'A comprehensive exploration of outlier detection in unstructured data for enhanced business intelligence using machine learning', *Journal of Business and Management Studies*, Vol. 6, No. 1, pp.238–245.

Schlosser, T., Beuth, F., Friedrich, M. and Kowerko, D. (2019) 'A novel visual fault detection and classification system for semiconductor manufacturing using stacked hybrid convolutional neural networks', *2019 24th IEEE International Conference on Emerging Technologies and Factory Automation (ETFA)*, IEEE, pp.1511–1514.

Song, Y. and He, Y. (2023) 'Toward an intelligent tourism recommendation system based on artificial intelligence and IoT using apriori algorithm', *Soft Computing*, Vol. 27, No. 24, pp.19159–19177.

Temple, J.T. (2023) 'Characteristics of distance matrices based on Euclidean, Manhattan and Hausdorff coefficients', *Journal of Classification*, Vol. 40, No. 2, pp.214–232.

Wang, Y., Zhang, J., Yang, Z. et al. (2024) 'Improving extractive summarization with semantic enhancement through topic-injection based BERT model', *Information Processing & Management*, Vol. 61, No. 3, p.103677.

Wei, T., He, S., Hu, Z. et al. (2024) 'Wind turbine abnormal data cleaning based on improved isolation forest algorithm', *Science Technology and Engineering*, Vol. 24, No. 9, pp.3691–3699.

Xu, R., Mayer, W., Chu, H. et al. (2024) 'Automatic semantic modeling of structured data sources with cross-modal retrieval', *Pattern Recognition Letters*, Vol. 177, pp.7–14.

Yamaguchi, A. and Saito, A. (2022) 'Second-level randomness test based on the Kolmogorov-Smirnov test', *JSIAM Letters*, Vol. 14, pp.73–76.

Yan, H. (2024a) 'The innovative application of data analysis techniques in audit work', *Finance and Economics*, Vol. 2024, No. 10, pp.165–167, DOI: 10.19887/j.cnki.cn11-4098/f.2024.10.002.

Yan, X. (2024b) 'A procurement cost control method for manufacturing enterprises based on data analysis', *Finance and Economics*, Vol. 2024, No. 10, pp.66–68, DOI: 10.19887/j.cnki.cn11-4098/f.2024.10.047.

Yang, Y., Liu, M., Ye, F. et al. (2021) 'A dynamic environment management system for data center data centers based on digital twins', *Electrical Technology*, Vol. 2021, No. 20, pp.35–37, DOI: 10.19768/j.cnki.dgjs.2021.20.012.

Zhang, B. and Lv, X. (2024) 'Research on machine learning based operational data analysis and optimization of thermal power plants', *Electrical Technology and Economics*, Vol. 2024, No. 5, pp.68–70.

Zhao, J. (2024) 'The application of data analysis in new energy investment decision-making', *China Collective Economy*, Vol. 2024, No. 12, pp.65–68.

Zhao, L., Gao, W. and Fang, J. (2024) 'Optimizing large language models on multi-core CPUs: a case study of the BERT model', *Applied Sciences*, Vol. 14, No. 6, p.2364.

Zheng, Z., Wang, M. and Tian, W. (2021) 'Research on missing data filling based on weighted K-nearest neighbor algorithm', *Intelligent Computers and Applications*, Vol. 11, No. 11, pp.31–33, 42.

Zhenpeng, Y. (2024) 'Application of artificial intelligence in computer network technology in the age of big data', *Journal of Artificial Intelligence Practice*, Vol. 7, No. 1, pp.11–16.