Warehouse Prediction and Planning Based on K-MEANS Algorithm and GA-LSTM

Junyang Xu [1, †, *], Yixin Wang [2, †], Xiya Liu [1, †], Yuxin Liu [1, †], Yinxing Zhao [1, †], Tong Han [3], Dan Wei [3], Qianyue Luo [4], Runqi Zhong [5], Fujia Guo [4]

[1] College of Materials Science and Engineering, Xi'an Shiyou University, Xi'an, China

[2] School of Electronic Engineering, Xi'an Shiyou University, Xi'an, China

[3] School of Economics and Management, Xi'an Shiyou University, Xi'an, China

[4] College of Petroleum Engineering, Xi'an Shiyou University, Xi'an, China

[5] College of New Energy, Xi'an Shiyou University, Xi'an, China

[†] These authors also contributed equally to this work

* Corresponding Author Email: 202212050514@stumail.xsyu.edu.cn

Abstract. The purpose of this paper is to study warehouse prediction and planning based on K-means algorithm and GA-LSTM through data analysis and deep learning methods. Firstly, this paper utilizes the inventory and daily sales data and uses K-means clustering analysis to classify the data and verifies the validity of the clustering results by evaluating the indicators. Then, the daily sales volume is predicted based on the LSTM model, and the GA-LSTM is further used for multiple fitting, which yields more stable and accurate prediction results. The dual-objective function model for minimizing the daily rental cost and maximizing the upper capacity limit is established by introducing piecewise constraints and correlation constraints through optimization algorithms (SSA, AROA). The results show that the daily rental cost of warehouse capacity and the capacity cap fluctuate within a reasonable interval in the selected optimization solution, respectively, which verifies the effectiveness and superiority of the proposed method in warehouse planning. The research in this paper provides a new idea of warehousing prediction and planning for e-commerce enterprises, which can effectively improve the accuracy and efficiency of warehousing management and provides a theoretical basis and practical guidance for subsequent inventory optimization and supply chain management.

Keywords: K-mean clustering; GA-LSTM; SSA (Sparrow Search Algorithm); AROA (Attraction-Repulsion Optimization Algorithm).

1. Introduction

In the context of rapid development of e-commerce, warehouse management is faced with the challenge of the proliferation of commodity types and quantities [1]. In order to improve management efficiency and reduce costs, this paper integrates a variety of advanced algorithms and models, including K-mean clustering [2], GA-LSTM [3] (Genetic Algorithm Combined with Long and Short-Term Memory Networks), SSA [4] (Sparrow Search Algorithm), and AROA [5] (Attractive-Repulsive Optimization Algorithm), for optimal allocation of warehousing resources. First, for the prediction of inventory and sales, K-mean clustering is used to preprocess the data to better analyze the characteristics of different categories. Then, the GA-LSTM model is combined to accurately predict the daily sales volume. In the optimization of the binning scheme, SSA and AROA algorithms are introduced in this paper, aiming to find the optimal binning scheme by comprehensively considering several indexes such as the total bin rental cost, bin utilization rate and capacity utilization rate. Through multi-objective optimization, it ensures the minimization of warehousing cost under the premise of meeting business constraints. In summary, this paper improves the efficiency and accuracy of warehouse management through the comprehensive use of a variety of computer algorithms, which provides strong support for the development of e-commerce enterprises [6].

2. Forecast of Warehouse Cargo Volume

2.1. Data Preprocessing

2.1.1 Preprocessing of Categories, Dates, and Inventory Quantities

In the paper, we provide the months and corresponding inventory quantities of 350 categories in the storage network, after sorting the data (category and date sorting), we process the data with outliers and missing values, after observing and counting the data, there are no outliers and missing values, because the given data does not include the time data such as "Double 11" and "Double 12", so there is no need to eliminate additional outliers, and we only perform descriptive statistics on the inventory quantities. Descriptive statistics are sufficient.

2.1.2 Cluster Analysis (Inventory Quantity)

Because the inventory quantity varies greatly, the maximum value is 4676058, the minimum value is 1, so the inventory quantity is clustered to analyze the data more clearly. In this paper, k-means (k-means) algorithm is used to categorize the data, and the basic principles of clustering index are as follows.

(1) Clustering Criteria

For the continuity features, the similarity between samples is measured by the following distance, as shown in table 1.

Table 1. Distance Formula

Distance	Determinant	Clarification
Absolute distance	$d_{ij}(1) = \sum_{k=1}^{p} \lvert x_{ik} - x_{jk} \rvert$	distance operations in one dimension
Euclidean distance	$d_{ij}(2) = \sqrt{\sum_{k=1}^{p} (x_{ik} - x_{jk})^2}$	distance operations in two dimensions
Minkowski distance	$d_{ij}(q) = [\sum_{k=1}^{p} (x_{ik} - x_{jk})^q]^{\frac{1}{q}}, q > 0$	distance operations in q-dimensional space
Chebyshev distance	$d_{ij}(\infty) = \max_{1 \le k \le p} \lvert x_{ik} - x_{jk} \rvert$	q Minkowski distance when taken to infinity
Lance distance	$d_{ij}(L) = \sum_{k=1}^{p} \frac{\lvert x_{ik} - x_{jk} \rvert}{x_{ik} - x_{jk}})$	Ability to reduce the impact of extremes

For discrete features, the VDM distance is used to measure the similarity between samples:

Let $\mathrm{m_{u,a}}$ denote the number of samples that take the value a on attribute u, $\mathrm{m_{u,a,i}}$ denote the number of samples that take the value a on attribute u in the ith sample cluster, and k is the number of sample clusters, then the VDM distance between two discrete values a and b on attribute u is:

$$\mathrm{VDM_P(a,b)} = \sum_{\mathrm{i=1}}^{\mathrm{k}} \left| \frac{\mathrm{m_{u,a,i} - m_{u,b,i}}}{\mathrm{m_{u,a} - m_{u,b}}} \right|^{\mathrm{p}} \quad (1)$$

For mixed features, the combination of distances described above is used to measure the similarity between samples:

Mixed attributes can be handled by combining the Minkowski distance with the VDM, assuming that there is $\mathrm{n_c}$ ordered attributes and $\mathrm{n - n_c}$ unordered attributes, without loss of generality, such that the ordered attributes are ranked before the unordered attributes, then:

$$\mathrm{MinkovDM}M_P(x_i, x_j) = \left(\sum_{u=1}^{n_c} \left| x_{iu} - x_{ju} \right|^p + \sum_{u=n_{c+1}}^{n} VDM_P(x_i, x_j) \right)^{\frac{1}{p}} \quad (2)$$

(2) Clustering Metrics

Since there is no real label, the contour coefficient is used here as the algorithm performance evaluation index, which contains two factors, cohesion degree and separation degree. The degree of

cohesion can be understood as reflecting the degree of closeness of a sample point to the elements within the class. Separation degree can be understood as reflecting the closeness of a sample point to the elements outside the class.

The formula for the contour coefficient is:

$$S(i) = \frac{b((i))-a((i)}{\max\{a(i\},b(i)} \tag{3}$$

Where $a(i)$ denotes the degree of cohesion of the sample point, calculated as:

$$a(i) = \frac{1}{n}\sum_{j\neq i}^{n} d_{ij} \tag{4}$$

Where: $b(i)$ is calculated in a similar way as $a(i)$. Except that it is necessary to traverse the other class clusters to get multiple values, from which the smallest value is selected as the result.

(3) K-mean Clustering Model Solving

Based on the elbow rule, as the number of clusters increases, the inertia of the clustering result will decrease, but the decrease will be slower. When the number of clusters increases to a certain degree, the reduction of inertia will slow down dramatically, forming an inflection point like the elbow. The number of clusters corresponding to this inflection point is the optimal number of clusters. Therefore, the number of clusters in 3 or 4 form an inflection point, the number of clusters used in this paper is 3 can be clearer to the data clustering.

The data is processed by k-mean model. From Fig. 1 can be obtained from the k-mean model classification results, category 1 category samples are much higher than category 2 and category 3, but the inventory of category 3 is much higher than category 2 and category 1. And it can be seen from the change of inventory over time that the inventory is higher from January to June and lower from July to September.

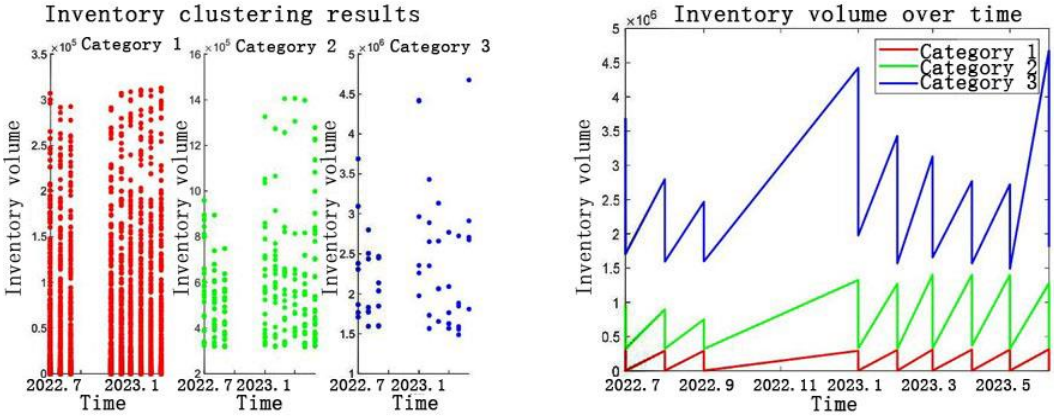

Fig. 1 Results of k-means model clustering and stock levels over time

2.2. Cluster analysis (daily sales)

The number of clusters used for daily sales is also 3, so the clustering results and evaluation indicators for daily sales are shown in table 2.

Table 2. Cluster Class Center and Evaluation Index

Type of clustering	Center value sales
1	484.4248967
2	34314.93624
3	8264.215733

In the final analysis, the contour coefficient is 0.837, DBI is 0.448, CH is 121298.083, and the daily sales volume is clustered better.

2.3. Evaluation Indexes of Time Series Forecasting Model

(1) RMSE (Root Mean Square Error): It is one of the common indicators to evaluate the accuracy of forecasting models. It measures the degree of deviation between the predicted value and the real value and is calculated as the square root of the sum of the squares of the differences between the predicted value and the real value. The value of RMSE ranges from 0 to infinity, and the smaller the value is, the smaller the prediction error of the model is, and the better the prediction ability of the model is.

$$RMSE = \sqrt{\frac{1}{n}\sum_{i=1}^{n} (y_i - \dot{y}_i)^2} \tag{5}$$

Where: n is the sample value; y_i is the true value; $\dot{y}_i$ is the predicted value.

(2) MAE (Mean Absolute Error): It is a commonly used assessment index. It calculates the average of the absolute value of the prediction error of each sample, the advantage of MAE is that it can intuitively see the size of the gap between the predicted value of the model and the true value, it does not take into account the positive or negative of the predicted value, and also pays more attention to the size of the absolute error, which makes it more reflective of the degree of deviation from the predicted value than the mean square error. The MAE is equal to 0 when the predicted value matches the true value, i.e. the perfect model, the larger the error, the larger the value.

$$MAE = \frac{1}{n}\sum_{i=1}^{n} |y_i - \dot{y}_i| \tag{6}$$

(3) LOSS FUNCTION: MSE (Loss of Mean Squared Error): is one of the most used loss functions in machine learning, deep learning regression tasks. Intuitively understanding the loss of mean square error, this loss function has a minimum value of 0 (when the prediction is equal to the true value) and a maximum value of infinity. MSE is the calculation of the Euclidean distance between the predicted value and the true value. The closer the predicted value is to the true value, the smaller the mean squared error of the two, and the mean squared error function is often used in linear regression, time series forecasting, and so on.

$$\mathrm{MSE} = \left(\frac{1}{N}\right)(y_i - \dot{y}_i)^2 \tag{7}$$

2.4. GA-LSTM Modeling

Three gating units are introduced: input gate, forgetting gate and output gate. These gates control the flow of information and forgetting. The so-called "gate" structure is used to remove or add the ability of information to the cell state. Here the cell state is the core, which belongs to a hidden layer, like a conveyor belt, that runs along the entire chain. The details of the LSTM are described as follows (the current cell is called time step t).

Forgetting the gate decides which information needs to be forgotten. The output C_t is a value between 0 and 1, indicating how much past information should be forgotten. When C_t is close to 1, past information is completely retained; when C_t is close to 0, past information is completely forgotten.

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \tag{8}$$

Input gate decides which information should be retained and updates the cell state. The output t is a value between 0 and 1, indicating which new inputs should be retained. When 4 is close to 1, all new inputs are completely retained; when "is close to 0, all new inputs are completely ignored. Next, the candidate cell state $C_t t$ is calculated, which indicates how much the new inputs at the current time step can affect the cell state. The role of the input gate is to control the weight of the new inputs at the current time step t. The input gate is a function of the cell state.

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \tag{9}$$

$$C_t = \tanh(W_C \cdot [h_{t-1}, x_t] + b_C) \tag{10}$$

Output gate: controls which information should be output. Output O_t is a value between 0 and 1, indicating which information should be output. When O_t is close to 1, all information is completely preserved; when O_t is close to 0, all information is completely blocked. The LSTM processes the cellular state C_t through a $\tan h$ function to obtain the hidden state h_t for the current time step.

$$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \tag{11}$$

$$h_t = o_t \tan h(C_t) \tag{12}$$

Cell state: the cell state can be seen as the core of the entire LSTM network, which stores and transmits information, as well as controlling the flow of information and updates. The cell state of the LSTM is updated and passed on to the next time step $t + 1$.

$$C_t = f_t * C_{t-1} + i_t * C_t \tag{13}$$

2.5. Solving for Sales (LSTM)

Since each category has its corresponding data such as sales, date, etc., the category and date are considered as inputs and the daily sales quantity of the item is considered as an output, so that the LSTM neural network can be utilized to train the data of the item data, and finally, the trained model will be used for predicting the sales quantity of the item.

Four sets of data are randomly selected, namely, category 61, 121, 181, 271; during the iteration process of the model, the RSME of the LSTM model is finally stabilized at about 0.4, and in the evaluation of neural network performance indexes, it is usually considered that the neural network has better performance when the RSME is in the range of 0.1-1, and therefore it can be assumed that this neural network has a better fitting effect and the model obtained is more accurate. Therefore, it can be assumed that this neural network has a better fitting effect, and the model obtained is more accurate.

Subsequently, the time series graphs of the real, fitted, and predicted values are plotted. From Fig. 2, the fitting effect is very good, so the error of predicting the sales result using LSTM is small.

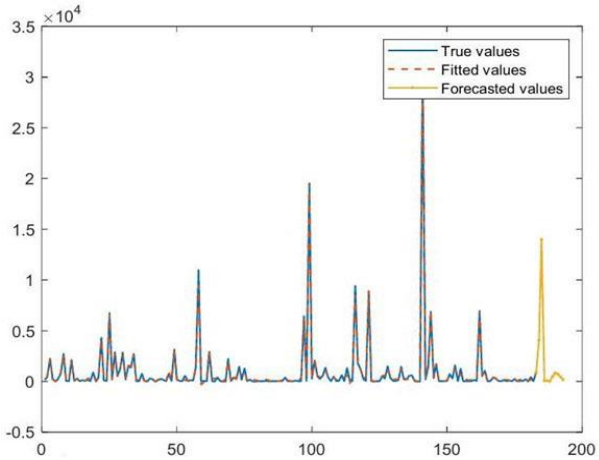

Fig. 2 Time series plot of true, fitted, and predicted values

3. Analysis of Categorized Warehousing Programs

3.1. Construction of the Planning Model

In this section, the planning model is used to solve the problem, and the optimization algorithms include SSA, AROA, etc. The constraints are as follows.

(1) The number of warehouses for the same category cannot exceed 3, and its objective function is:

$$\text{Minimize} Z = f(x) + M \sum_{i=1}^{n} \max\left(0, \sum_{j=1}^{m} x_{ij} - 3\right) \tag{14}$$

Where: f(x) is the original objective function which reflects the main objective desired to be optimized. M is a penalty factor large enough to ensure that any violation of the constraints during the optimization process will result in a significant increase in the objective function value; $\max\left(0, \sum_{j=1}^{m} x_{ij} - 3\right)$ is the penalty term which will be greater than 0 when the number of items of category i stored in all warehouses exceeds 3, thus introducing a penalty in the objective function.

(2) The same piece type and the same high-level categories are placed in one warehouse as much as possible, and its objective function is:

$$\text{Minimize} Z = f(x) + M_1 \sum_{i=1}^{n} \sum_{j=1}^{p} \left(\sum_{k=1}^{m} y_{ijk} - 1\right)^2 \tag{15}$$

Where: M_1 is a large penalty factor used to ensure that the penalty term plays a sufficient role in the objective function. is the penalty term, which measures whether the same piece-type, same high-level category is placed in different warehouses. If piece type j of each category i is stored in only one warehouse, then $\sum_{k=1}^{m} y_{ijk} - 1$ will be equal to 1 and the penalty term will be zero. If it is stored in more than one warehouse, the penalty term will increase, thus increasing the value of the objective function.

(3) The objective function for the same category distributed among different warehouses with the same proportion of stocked quantity and the same proportion of outgoing quantity is.

$$\text{Minimize } Z = f(x) + M_2 \sum_{i=1}^{n} \left(\sum_{k=1}^{m} \left(\frac{S_{ik}}{T_i} - \alpha\right)^2 + \sum_{k=1}^{m} \left(\frac{D_{ik}}{R_i} - \beta\right)^2\right) \tag{16}$$

Where: M_2 is a large penalty factor used to ensure that the penalty term plays a sufficient role in the objective function. The first penalty term $\sum_{k=1}^{m} \left(\frac{S_{ik}}{T_i} - \alpha\right)^2$ measures the difference between the inventory stock ratio and the target ratio between different warehouses for the category i. The second penalty term $\sum_{k=1}^{m} \left(\frac{D_{ik}}{R_i} - \beta\right)^2$ measures the difference between the ratio of stock-out quantity and the target ratio between different warehouses for category i.

3.2. SSA (Sparrow Search Algorithm)

Sparrows in nature have two main behaviors: subject to feeding and anti-feeding.

Tolerant feeding: Sparrows are categorized into explorers and followers; the sparrows that can search for better food (higher fitness function) are explorers and the rest of the sparrows are followers influenced by the direction of explorers.

Anti-feeder: a certain percentage of sparrows in the sparrow population will scout and behave accordingly when they find a feeder coming.

(1) Initialization of Group Location

$$x = lb + rand * (ub - lb) \tag{17}$$

where, ub, lb represent the upper and lower position boundaries of the population respectively

(2) Explorer Position Updating

$$x_i^{t+1} = \begin{cases} x_i^t * \exp\left(\frac{-i}{\alpha * iter_{max}}\right), R_2 < ST \\ x_i^t + Q, R_2 \geq ST \end{cases} \tag{18}$$

Where R_2, ST are the safety and warning values, respectively. When $R_2 < ST$, the sparrow conducts a global search to be fed; when $R_2 \geq ST$, the sparrow wanders randomly with a normal distribution.

(3) Follower Position Update

$$x_i^{t+1} = \begin{cases} Q * \exp\left(\frac{x_{worst}^t - x_i^t}{i^2}\right), i > \frac{n}{2} \\ x_p^{t+1} + \left|x_i^t - x_p^{t+1}\right| * A^+, \text{otherwise} \end{cases} \tag{19}$$

When $i > n/2$, the less adapted sparrow does not acquire food, and therefore needs to acquire more food; other cases illustrate that the sparrow searches for a random location near the current optimal location.

(4) Scouting and Warning

$$x_i^{t+1} = \begin{cases} xb_i^t * \beta(x_i^t - xb_i^t), f_i \neq f_g \\ x_i^t + K\left(\frac{x_i^t - xw_i^t}{|f_i - f_w| + \varepsilon}, f_i = f_g\right. \end{cases} \tag{20}$$

When $f_i \neq f_g$, the sparrow is at the edge of the population, easy to be attacked by the predator, at this time towards the optimal sparrow flight; when $f_i = f_g$, the sparrow is in a dangerous position, away from the worst sparrow position.

3.3. AROA (Attraction-Repulsion Optimization Algorithm)

AROA mimics the equilibrium associated with the phenomenon of attraction-repulsion that occurs in nature, where candidate solutions move through the search space according to the quality of solutions in their neighborhoods and the best candidate solution.

(1) Attraction and Repulsion

The positions of the candidate solutions are updated according to the information about fitness achieved by the other members of the group, depending on the distance between the group members.

$$d^2(x_i, x_j) = \sum_{k=1}^{\dim} \left(x_i^k - x_j^k\right)^2 \tag{21}$$

(2) The Vector n_i is Added to the Positions of the Candidate Solutions

$$n_i = \frac{1}{n}\sum_{i=1}^{k} c(x_j - x_i) I(d_{i,j}, d_{i,\max}) s(f_i, f_j) \tag{22}$$

(3) Evaluate the Solution Impact Function

$$I(d_{i,j}, d_{i,\max}) = 1 - \frac{d_{i,j}}{d_{i,\max}} \tag{23}$$

(4) Direction of Change

$$s(f_i, f_j) = \begin{cases} 1 & f_i > f_j \\ 0 & f_i = f_j \\ -1 & f_i < f_j \end{cases} \tag{24}$$

(5) Number of Neighbors k

$$\mathrm{k} = \left|\left(1 - \frac{\mathrm{t}}{\mathrm{t_{max}}}\right)\mathrm{n}\right| + 1 \tag{25}$$

(6) $t_{\max}$ Maximum Number of Iterations

$$t_{\max} = \left|\frac{fes_{\max} - n}{2n}\right| \tag{26}$$

3.4. Planning Model Solving

Through the addition of two major constraints and through the two major constraints and the above constraints want to integrate and the objective function, used to solve this paper a product of three warehouses program, the use of multi-objective planning, will be the total cost of the daily rental, the piece type ABC, as well as the correlation between the types of four matrices, brought into the establishment of the two major objective function for the solution, and ultimately get a product of three warehouses solution program, part of the following Fig. 3 shows.

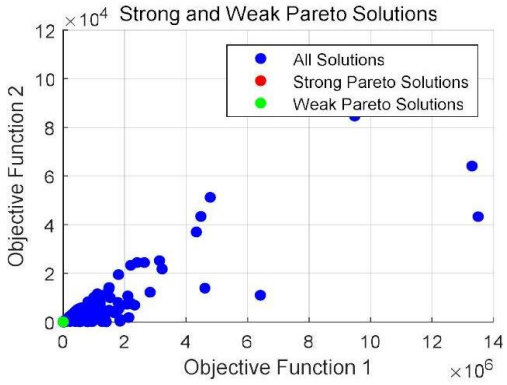

Fig. 3 multi-objective function solution for one product and three bins

The coordinates of the points in the obtained result are, from left to right, $(-83.17, 27.92)$, $(-50.57, 26.81)$ and $(-49.73, 26.20)$, respectively. Referring to the Strong Pareto Solution and Weak Pareto Solution distributions, it can be seen that $(-83.17, 27.92)$ and $(-49.73, 26.20)$ are like the uppermost and lowermost ends of the Strong Pareto Solution curves, and thus the optimal solution may be one of the two of them. Comparing the ranges of the horizontal and vertical coordinates, with $f2$ increasing by only 1.52, $f1$ decreases by 33.44, a change of 2786.67% of $f2$. Therefore, the point in the upper left corner is chosen as the optimal solution. Here, the minimum parameter of the daily rental cost of storage capacity for this optimal solution has 17 points, the daily rental cost is in the range $[0.32,4.31]$ with the mean value of 3.29, and the upper limit of the capacity is in the range $[1192.32,1964.71]$ with the mean value of 1666.05.

4. Conclusion

In this paper, a comprehensive solution based on K-mean algorithm, GA-LSTM model and optimization algorithm is proposed for the situation of inventory quantity prediction and binning planning faced by e-commerce enterprises in warehouse management. By accurately predicting the inventory volume and daily sales volume and further optimizing the binning scheme, the utilization rate of storage resources is effectively improved. First, in terms of inventory quantity prediction, through K-mean clustering analysis of historical data, the data were successfully divided into different categories, revealing the inventory and sales characteristics of different categories. Based on this, the LSTM model was used to predict the daily sales. Under the multiple fitting of GA-LSTM, the RSEM and LOSS values of the model are stabilized in a low range (0.41992 and 0.32923), which indicates that the prediction effect is good and meets the demand for accurate prediction in warehouse management. Secondly, in the optimization of warehouse planning, this paper adopts SSA (Sparrow Search Algorithm) and AROA (Attraction-Repulsion Optimization Algorithm) to find the optimal binning scheme. By introducing piece-type constraints and category relevance constraints, the warehouse layout is optimized, which significantly reduces the daily rental cost, and at the same time improves the capacity utilization of the warehouse. In the Pareto frontier solution obtained from the solution, both the warehouse daily rental cost and the capacity utilization rate reach the optimal balance, demonstrating the win-win result of cost and efficiency.

References

[1] Liu Ying, Yu Ruobing. Research on modern warehouse management strategy based on big data analysis[J]. China Aviation Weekly, 2024, (48): 68-70.

[2] Hu Huaqiang, Wang Xi. Research on reconnaissance data sorting methods based on improved K-mean clustering algorithm [J]. Software, 2024, 45(09): 4-6.

[3] Fei Shanshan, Zhang Zhonglin. Improved genetic algorithm to optimize temporal prediction model for GA-LSTM networks [J/OL]. Computer Engineering and Applications, 1-9 [2024-12-19]. http: // kns. cnki. net/kcms/detail/11.2127.TP.20200528.1414.008.html.

[4] Chu Zheyu, Liu Dan, Zhang Rongbo, et al. Application of sparrow search algorithm and 0-1 planning in fresh storage site selection problem [J]. Automation Applications, 2021, (09): 56-59. DOI: 10. 19769/ j. zdhy.2021.09.015.

[5] ZhaoO Pengjun, Liu Sanyang, LI Chao. Particle swarm optimization algorithm based on attraction-repulsion mechanism [J]. Computer Applications, 2009, 29(02): 542-544+557.

[6] Xie Jing. Research on innovation strategy of marketing path of e-commerce enterprises under the background of knowledge economy [J]. Journal of Taiyuan City Vocational and Technical College, 2024, (11): 20-22. DOI: 10.16227/j.cnki.tycs.2024.0639.