CIT. Journal of Computing and Information Technology, Vol. 33, No. 1, March 2025, 57–71
doi: 10.20532/cit.2025.1005877

Automated Machine Learning and Data-Driven Decision Support System for Strategy Management in Organizational Activities

Meiling Lu

School of Business Administration, Guangzhou Civil Aviation College, Guangzhou, China

In modern organizational activities, the increasing complexity and dynamism of strategy management have rendered traditional static analysis and experience-based decision-making methods inadequate for meeting the rapidly changing market demands and intricate internal processes. To address these challenges, this paper proposes an automated machine learning-based data-driven decision support system. The system incorporates a flexible and scalable model that integrates a strategy management automation algorithm, combining Long Short-Term Memory (LSTM) networks and Deep Q-Network (DQN) algorithms, to enhance the scientific and accurate nature of decision-making. The integrated algorithm shows a significantly higher probability of successful decision-making in organizational environments of different scales compared to traditional DQN and random strategies, demonstrating its superiority in complex decision-making scenarios. Key data indicate that the algorithm exhibits strong stability and robustness in terms of error function curves, algorithm performance, and the number of successful decisions, further validating its effectiveness under various interference conditions. While existing research has attempted to apply machine learning to strategy management to some extent, common issues include inadequate handling of time series data, suboptimal strategy optimization, and lack of flexibility in system models. Experimental results show that the algorithm's decision success rate is significantly higher than that of traditional DQN and random strategies across various organizational scales, demonstrating its efficiency and stability in complex decision-making environments. This study not only provides innovative technical means for strategy management but also offers theoretical and practical references for the future development of intelligent decision support systems.

ACM CCS (2012) Classification: Computing methodologies → Machine learning → Machine learning algorithms

Keywords: strategy management, automated decision-making, machine learning, Long Short-Term Memory (LSTM), Deep Q-Network (DQN), data-driven decision support system, organizational activities

1. Introduction

In modern organizational activities, the complexity and dynamism of strategy management are increasing day by day [1–4]. Whether it is resource allocation, market strategy, project management, or operational optimization, the decision-making process needs to face a large amount of uncertainty and a changing external environment. Traditional strategy management methods often rely on experience and static analysis, which are difficult to adapt to the rapidly changing market demands and complex internal processes [5–7]. Therefore, building a data-driven decision support system based on machine learning to address these challenges has become an important research direction in the field of organizational management.

Many current machine learning algorithms require high-quality and large quantities of data, however, in practical organizational settings, data may contain noise, can be incomplete, or exhibit inconsistencies, which can impact the accuracy of these algorithms. Additionally, when dealing with high-dimensional, diverse, and unstructured data, machine learning models may face challenges related to computational complexity and excessive resource consumption. Moreover, machine learning models are often "black boxes," making it difficult to explain their decision-making processes and outcomes. This lack of transparency can lead to a lack of trust among decision-makers in strategic management contexts. Researching the application of machine learning in strategy management can not only improve the scientific and accurate nature of decision-making but also significantly enhance the overall efficiency of the organization [8, 9]. Through automated data analysis and decision support, organizations can respond more agilely to market changes, optimize resource allocation, and enhance competitiveness [10–14]. In addition, the continuous development of machine learning algorithms provides a technical foundation for building more intelligent and efficient decision support systems. The significance of this research direction lies in introducing advanced technical means into the traditional management field, promoting innovation and improvement of management models [15, 16].

Although some studies have attempted to apply machine learning to strategy management, these methods usually have many deficiencies. Firstly, many methods are limited in handling time series data and capturing complex time-varying features, making it difficult to accurately predict future states [17]. Secondly, traditional strategy optimization algorithms often struggle to find optimal solutions in complex and changing environments, leading to unsatisfactory decision-making effects [18–21]. Furthermore, most existing system models lack flexibility and scalability, making it difficult to meet the personalized needs of different organizations. These deficiencies limit the widespread application and practical effects of machine learning in strategy management.

The research problem addressed in this paper is how to design and construct a flexible and scalable decision support system for strategy management in organizational activities, and to propose an automated decision-making algorithm capable of handling complex organizational data and diverse management needs. The research objectives include developing an effective decision support framework and designing an automated decision-making algorithm that integrates LSTM networks and DQN algorithms to enhance the accuracy and efficiency of strategy management. This study comprises two main components. The first is the development of a flexible and scalable decision support system model tailored to manage complex organizational data and diverse management needs. The second component introduces a strategy management automation algorithm that combines Long Short-Term Memory (LSTM) networks and Deep Q-Network (DQN) algorithms. This integration leverages LSTM to extract significant features from time series data and utilizes DQN for strategic optimization, enhancing decision accuracy and efficiency. Collectively, this research not only advances technical solutions for strategy management but also provides valuable insights for the evolution of intelligent decision support systems.

2. Strategy Management Decision Support System Model in Organizational Activities

The scenarios of automated strategy management in organizational activities cover multiple fields. In supply chain management, automated decision systems can analyze market demand, inventory levels, and supplier delivery conditions in real-time, and automatically adjust procurement and inventory strategies to ensure the efficient operation and cost control of the supply chain. In marketing, automated strategies can formulate and adjust marketing plans by analyzing consumer behavior data, market trends, and competitive dynamics, optimizing advertising placement and promotional activities, improving market response speed and marketing effectiveness. In human resource management, automated decision systems can analyze employee performance data, workload, and career development plans, automatically performing employee allocation, training needs prediction, and performance evaluation, enhancing the precision and efficiency of human resource management. In project management, automated systems can monitor project progress, resource usage, and risk factors in real-time, automatically adjusting project plans and resource allocation to ensure the project is completed on time and within budget. In financial management, automated decision systems can analyze the company's financial data and market economic indicators, automatically conducting budget preparation, cost control, and investment decisions, improving the scientific and accurate nature of financial management. In these scenarios, automated decision-making can not only improve management efficiency but also significantly reduce human errors and subjective biases in the decision-making process, enabling organizations to respond more flexibly and accurately to market changes and internal management needs.

In response to the above scenarios, this paper proposes a strategy management automation decision algorithm that integrates LSTM and DQN algorithms, fully considering two aspects during the implementation process: coordination among organizational members and avoidance of interference in strategy decisions. Regarding coordination among organizational members, the LSTM network, by capturing long-term and short-term dependencies in time series, can foresee the dynamic changes in collaboration between members and departments, providing a more comprehensive perspective for decision-making. In terms of avoiding interference in strategy decisions, the DQN algorithm, through the reinforcement learning mechanism, can autonomously learn the optimal strategy in complex and changing environments. Its automated decision-making process can reduce human subjective factors' interference and quickly adjust strategies to adapt to new situations when facing uncertainty and random interference. The algorithm also needs to consider the issues of internal power structure and interest distribution within the organization to ensure the fairness and transparency of decisions, gaining the trust and support of organizational members.

The goal of the algorithm is to maximize the effectiveness of organizational activities, ensuring that the organization maintains superior competitiveness and operational efficiency in various interference scenarios while achieving efficient decision-making. To achieve this goal, the algorithm needs the ability to quickly converge to the optimal strategy in a complex dynamic environment. Additionally, the algorithm design must consider various interference factors in practical application scenarios, including changes in internal and external environments, dynamic resource allocation, and coordination issues among members.

In the constructed automated decision model for strategy management in organizational activities, each organizational member forms an organizational network within a certain area, possessing the ability to perceive the strategic environment, *i.e.*, to obtain and analyze strategy-related information in real-time to make decisions. Each organizational member can perform four main tasks within a decision cycle: strategy execution, strategy perception, strategy selection, and learning decision. It is assumed that in each decision cycle, all members of the organization have the task of strategy execution, but they can choose not to execute a specific strategy in a certain cycle but to engage in strategy perception or learning decision instead. In the model, the set of strategies that organizational members can choose from is the same as the set of strategies that external interference factors, such as market competitors' strategies, might adopt. This means that when selecting strategies, organizational members need to consider the impact of external interference factors and formulate strategies with interference resistance. Figure 1 shows a schematic diagram of the strategy management decision cycle in organizational activities.

Specifically, in the model, organizational members can be viewed as agents. Each organizational member, within a decision cycle, perceives the current state T_s from the strategic environment, where T represents the state space. This state information can include market dynamics, competitor behavior, internal resource conditions, and other factors affecting strategy formulation and execution. Based on the current state t_s, organizational members need to choose an action x_s from the action space X, which is to choose a strategy or decision to execute. This choice process needs to consider the current environmental conditions and possible evolutions. Based on the input state t_s and the chosen action x_s, the organizational member will receive a reward e_s, which could be economic gain, market share growth, or other forms of performance indicators. In the next decision cycle $s + 1$, the environmental state t_s will change to a new state $t_s + 1$ based on the organizational member's action x_s and changes in the external environment. At the end of each decision cycle, organizational members update their strategies based on the reward e_s they received. This process is similar to policy updates in reinforcement learning, optimizing strategy selection through continuous trial and error and feedback. The goal of organizational members is to maximize the cumulative reward obtained throughout the decision process through this series of interactions. This means they need to continually pursue the optimal decision outcome in the process of perceiving the environment, selecting strategies, executing strategies, and updating strategies.

In the model, let the set of organizational members be $V = \{1, 2, ..., V\}$, representing all members involved in strategy formulation and execution. The strategy space $L = \{1, 2, ..., L\}$ represents all possible strategies or actions, where $V < L$ means each member has multiple strategy choices. Each state in the state space T can be represented by a vector, where t_u represents the state of the u-th factor. The state of each factor can be either discrete or continuous. The condition for member j to successfully make a decision is that the chosen strategy is not occupied by other members or affected by external interference factors. In this case, the system generates a confirmation signal, denoted as $G_v(s)$:

$$G_v(s) = \begin{cases} 1 & \text{The decision is successful} \\ 0 & \text{The decision is unsuccessful} \end{cases}. \quad (1)$$

Define the action space for each organizational member as $X_v = \{0, 1, 2, ..., L\}$, *i.e.*, the set of strategies each member can choose from. The joint action space of all members is $X = \otimes X_v = (1, 2, ..., V)$, which represents the Cartesian product of all members' strategy choices, indicating all possible strategy combinations. Further define the reward function as $e_v(T_s, x_v, s)$. After each member chooses a strategy in time slot s, they receive an immediate reward e_v based on the current state T_s and the chosen strategy x_v. The reward function can be defined based on the effect of strategy execution, such as profit growth, market share increase, or resource utilization efficiency. Suppose the action chosen by member v is denoted as x_v, the action of the interferer is denoted as x_k, and the action chosen by any user other than v in the set of organizational members is denoted as x_m, then:

$$e_v(t_s, x_v, s) = \begin{cases} 1 & x_v \neq x_k \text{ and } x_v \neq x_m \left(m \neq v, m \in V\right) \\ 0 & \textit{Others} \end{cases}. \quad (2)$$

The goal of each member over a period is to maximize their cumulative discounted reward, typically represented by the following formula, where ε is the discount factor and S is the time span. Cumulative discounted reward considers long-term benefits, encouraging members to focus not only on immediate benefits but also on long-term impacts:

$$E_v = \sum_{s=1}^{S} \varepsilon^s e_v(T_s, x_v, s). \quad (3)$$

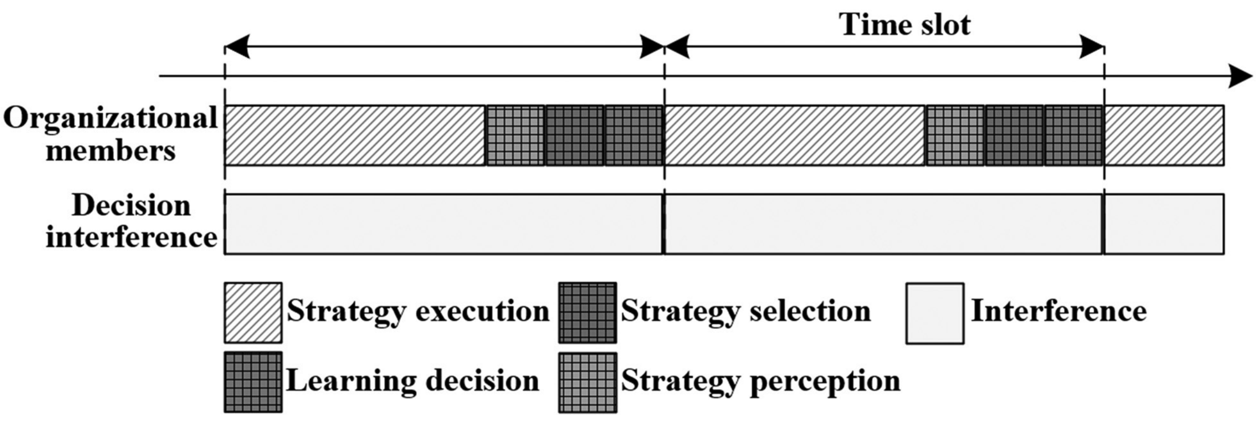

Figure 1. Schematic diagram of the strategy management decision cycle in organizational activities.

To optimize the overall strategy management of the organization, the model also needs to define the global reward for a certain time slot s, which is the total or weighted average of the immediate rewards for all members in that time slot:

$$E_h = \sum_{v=1}^{V} e_v\left(t_s, x_v, s\right). \tag{4}$$

3. Automated Strategy Management Decision Algorithm Based on LSTM and DQN

3.1. DQN Algorithm

Strategy management in organizational activities involves many complex decision processes that can be optimized through the interaction between agents and the environment. In this context, the goal of the agent is to continually adjust its strategy to maximize the rewards obtained from the environment, thereby finding the optimal strategy management solution.

The DQN algorithm is a model-free reinforcement learning algorithm that combines the traditional Q-learning algorithm with deep neural networks to approximate the state-action value function $Q(t, x)$. In the strategy management of organizational activities, the state s can represent the current organizational state or resource configuration, and the action x represents different strategy choices or decisions. By using neural networks to approximate $Q(t, x)$, we can handle high-dimensional state and action spaces, which is particularly important in complex organizational environments. Figure 2 shows the structure of the DQN value network. Assuming the learning factor is denoted by $\beta(0 < \beta < 1)$, and the state value function is represented by $N^*(t_s + 1)$, the update formula for $Q(t, x)$ is as follows:

$$Q\left(t_s, x_s\right) \leftarrow (1-\beta) Q\left(t_s, x_s\right) + \beta\left[e_s + \varepsilon N^*\left(t_{s+1}\right)\right] \tag{5}$$

$$N^*\left(t_{s+1}\right) = \underset{x \in X}{MAX}\, Q\left(t_{s+1}, x\right) \tag{6}$$

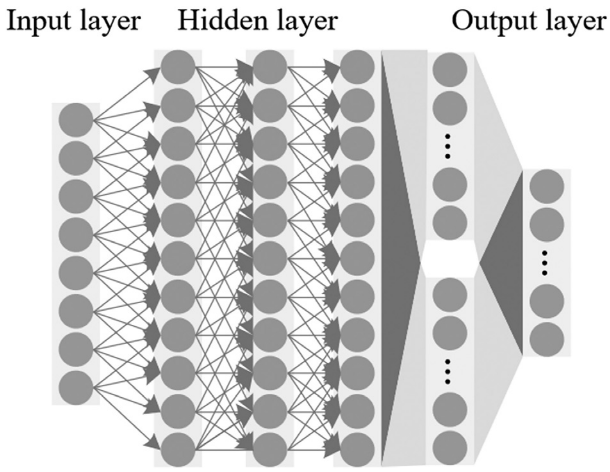

Figure 2. DQN value network structure.

In DQN, the weight parameters ϕ_u of the neural network are trained by minimizing the error between the predicted Q-value $q(t, x; \phi_u)$ and the target Q-value b. The target Q-value b is usually determined by the current reward and the maximum Q-value of the next state. The predicted $w(t, x; \phi_u)$ value can be iteratively updated by the neural network.

$$q\left(t_s, x_s; \varphi_u\right) \leftarrow (1-\beta) q\left(t_s, x_s; \varphi_u\right) + \beta\left[e_s + \varepsilon \underset{x_{s+1}}{MAX}\, q\left(t_{s+1}, x_{s+1}; \varphi_u\right)\right] \tag{7}$$

The error function can be computed as follows.

$$LOSS\left(\varphi_u\right) = R\left[b_u - q\left(t_s, x_s; \varphi_u\right)\right]^2 \tag{8}$$

The w-value b_u can be expressed as:

$$b_u = R\left[e_s + \varepsilon MAX q\left(t_s, x_s; \varphi_u\right)\right] \tag{9}$$

To prevent the agent from falling into local optima during decision-making, we use the ε-greedy policy for action selection. This policy randomly selects an action with a probability of ε and chooses the current optimal action with a probability of $(1 - \varepsilon)$. As training progresses, the value of ε gradually decreases, thereby increasing the probability of exploiting the current optimal policy. This method encourages exploration in the initial stage, ensuring that the agent can fully explore the state-action space, and in the later stage, it exploits the best-learned strategies. Assuming a random number between

0 and 1 is represented by o_r, and the exploration probability is represented by $\gamma(0 < \gamma < 1)$, the ε-greedy policy expression defined as follows.

$$\tau^{\varepsilon} = \begin{cases} \underset{x \in X}{\operatorname{argmax}}\, q^{\tau}\left(t_s, x_s; \varphi_u\right) & if \quad o_r > \gamma \\ \dfrac{1}{|X|} \quad if \quad o_r \le \gamma \end{cases} \tag{10}$$

Assuming the maximum and minimum values that γ can take are represented by γ_{MAX} and γ_{MIN}, respectively, the decay factor is represented by ς, and the current iteration count is represented by π, then we have:

$$\gamma = \gamma_{MIN} + \left(\gamma_{MAX} - \gamma_{MIN}\right) e^{-\varsigma s}. \tag{11}$$

3.2. LSTM Model

In organizational activities, strategy management involves a large amount of time-series data, such as resource allocation records, project progress status, and employee performance data. This data not only contains current state information but also implicitly holds rich historical trends and potential future directions. In the automated decision-making system for strategy management in organizational activities, the LSTM model is used to process and predict the aforementioned time-series-based strategy data. Specifically, in automated strategy management decision-making, the LSTM model can analyze past strategy execution, identify key factors affecting strategy effectiveness, and predict future possible trends. In resource allocation strategies, the LSTM model can use past resource usage records to predict future resource demands, helping the organization to allocate resources more reasonably, avoiding waste or shortages. In project management strategies, the LSTM model can analyze historical project progress data to predict future project completion times and possible risk points, providing more accurate project planning and risk management advice.

The LSTM structure is a variant of the recurrent neural network (RNN), which addresses the issues of long-term dependencies and vanishing gradients in traditional RNNs by introducing three gating mechanisms: input gate, forget gate, and output gate. These mechanisms allow more effective use of historical information. The forget gate determines which information from the previous memory cell state should be retained or discarded, preventing excessive accumulation of information. The input gate controls the extent to which current input information updates the memory cell state, ensuring the effective storage of new information. The output gate determines which parts of the memory cell will be output, influencing the final prediction results. In the constructed model, the functions of these gates enable the LSTM to capture and retain key features in time series data, optimizing time dependencies in the decision-making process and thereby enhancing the accuracy and stability of the algorithm. Assuming the input weights and biases of the three gates are represented by $Q_{u,z,p}$ and $y_{u,z,p}$, a_s is the input at the current time s, the output of the LSTM cell at time $s-1$ is represented by g_{s-1}, the forget gate output is represented by d_s, and the cell state and candidate value are represented by Z_s and $\tilde{Z}_s$, respectively. The following expressions define the three gating mechanisms:

$$d_s = \text{sigmoid}\left(Q_d \times \left[g_{s-1}, a_s\right] + y_d\right) \tag{12}$$

$$u_s = \text{sigmoid}\left(Q_u \times \left[g_{s-1}, a_s\right] + y_u\right) \tag{13}$$

$$\tilde{Z}_s = \tanh\left(Q_z \times \left[g_{s-1}, a_s\right] + y_z\right) \tag{14}$$

$$Z_s = d_s Z_{s-1} + u_s \tilde{Z}_s \tag{15}$$

$$g_s = \text{sigmoid}\left(Q_0 \times \left[g_{s-1}, a_s\right] + y_o\right) \times \tanh\left(Z_s\right) \tag{16}$$

In dynamic and complex organizational environments, various unforeseen disturbance factors may affect the effectiveness of strategy execution. By analyzing anomalies and disturbance patterns in historical data, the LSTM model can identify potential disturbance factors in advance and provide corresponding adjustment suggestions to ensure effective strategy execution. In market strategy management, the LSTM model can analyze past market reaction data, predict future market trends and potential competitor behaviors, and help the organization formulate more forward-looking marketing strategies.

3.3 Automated Decision-making Algorithm for Organizational Activity Strategy Management

LSTM is used to capture important features in time series data to identify long-term dependencies, while DQN is employed for strategy optimization by utilizing a reinforcement learning framework to enhance decision-making strategies. The integration of the two allows DQN to perform more precise strategy learning and optimization based on the time series features provided by LSTM. The algorithm first utilizes the LSTM network to process and analyze time series data to extract important time-dependent features. These features are then fed into the DQN for strategy optimization. The DQN adjusts the strategy through reinforcement learning to improve decision accuracy. In different environments and under varying interference conditions, the algorithm demonstrates exceptional robustness and stability by optimizing decision strategies. Specifically, the output of the LSTM network serves as the state input for the DQN, enabling DQN to make more accurate decisions based on historical information from the time series during the decision-making process. This integration allows the decision-making process to not only consider the current state but also effectively leverage past experiences, thereby enhancing decision accuracy and robustness. In the automated decision-making model for organizational activity strategy management shown in Figure 3, the integration of LSTM and DQN algorithms provides robust support for complex and dynamic decision environments. In the model, assume the LSTM network input vector $A_v(s)$ represents the state information and action results of user v at time s, which includes detailed information about multiple strategy executions within organizational activities. In project resource management, vector $A_v(s)$ may contain details such as which resources are currently used or are idle, which resource was last allocated to which project, market demand changes, and project progress. On the other hand, DQN estimates the Q-value function through deep neural networks, linking the current state to future rewards. In strategy management, DQN can help an organization make optimal decisions. Regarding marketing strategies, DQN can analyze the current market environment and historical marketing data to select the marketing channels and strategic actions most likely to bring high returns. Specifically, the DQN algorithm continuously adjusts strategies to maximize the expected returns such as market share, sales, or customer satisfaction, thereby optimizing the organization's strategy execution.

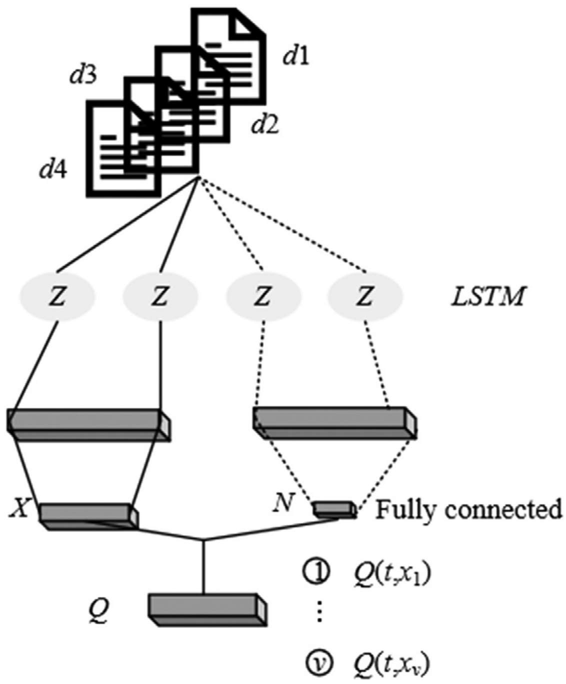

Figure 3. Structure diagram of the automated decision-making algorithm for organizational activity strategy management.

The application of the model in strategy management can be summarized by the following steps:

1. Data input and processing: Collect and construct historical data of various strategy executions in organizational activities to form the input vector $A_v(s)$.

2. Feature extraction: Use the LSTM network to process the input data and extract important information and features from time series. These features reflect the time dependencies between various variables during strategy execution, providing a foundation for prediction and decision-making.

3. Strategy optimization: Through the DQN algorithm, based on the features extracted by LSTM, evaluate the expected returns of various strategic actions. DQN continuously learns and updates, selecting the optimal strategic actions to maximize long-term returns.

4. Decision execution and feedback: Implement the selected strategic actions and adjust the model based on execution results and feedback to further optimize future strategic decisions.

Figure 4 presents the training process of the automated decision-making algorithm for organizational activity strategy management.

4. Experimental Results and Analysis

The primary aim of the experiments in this study is to evaluate the performance of the proposed integrated LSTM and DQN algorithm in automated decision-making for strategy management. To achieve this, the experiments utilized datasets from multiple simulated organizational environments, encompassing different scales of organizational members and various decision-making scenarios. The evaluation metrics included decision success probability, algorithm runtime, error function curves, and algorithm performance under internal and external interference. The baseline models selected for comparison were the traditional DQN algorithm and random strategies, to ensure the superiority of the proposed algorithm in complex decision-making environments. By analyzing the data in Figure 5, it can be observed that the error

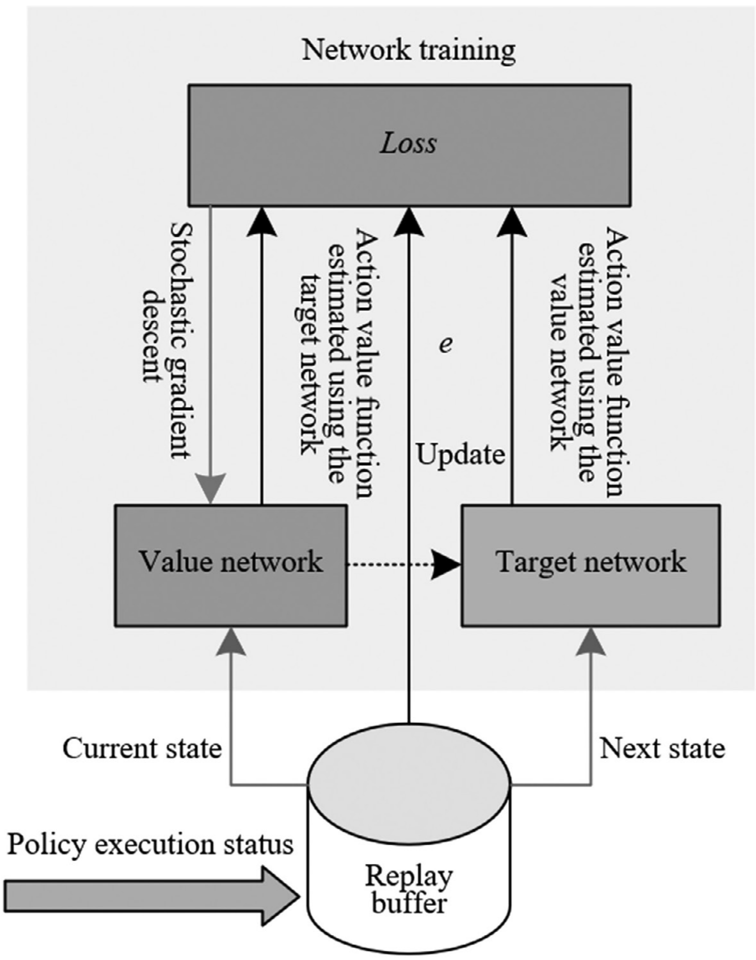

Figure 4. Flowchart of the training process of the automated decision-making algorithm for organizational activity strategy management.

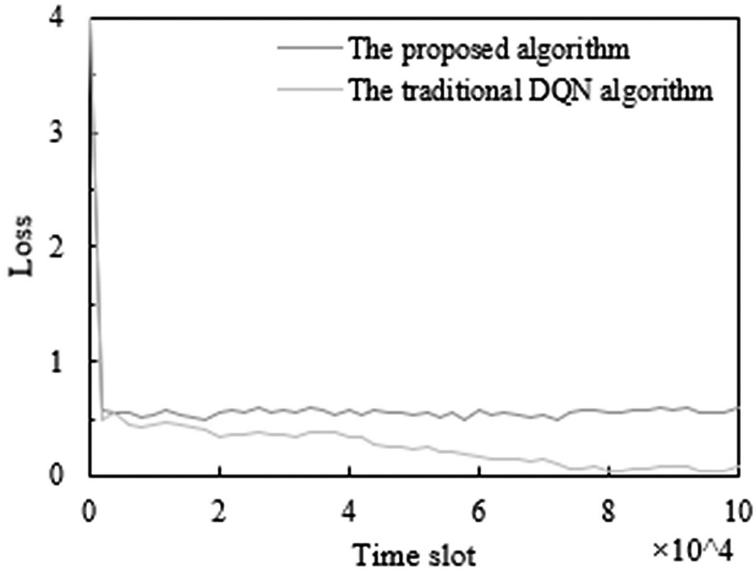

Figure 5. Comparison of error function curves of different decision algorithms.

function performance of the proposed algorithm and the traditional DQN algorithm at different time slots differs. The error function of the proposed algorithm starts at 0.58 at time slot 0, with small overall fluctuations, consistently remaining between 0.52 and 0.59, demonstrating high stability and consistency. In contrast, the error function of the traditional DQN algorithm starts at 0.48 at time slot 0, showing some fluctuations in the initial slots (*e.g.*, time slots 2 and 4), but then sharply declines after time slot 10, eventually dropping to 0.09 at time slot 50. The error of the traditional DQN algorithm gradually increases, especially in the later stages, with the error value significantly lower than that of the proposed algorithm, exhibiting greater volatility and instability.

In strategy management within organizational activities, the interference factors are roughly divided into two categories: internal interference and external interference. Internal interference mainly stems from internal factors within the organization, including poor communication, resource conflicts, interest conflicts, as well as organizational culture and structure issues. On the other hand, external interference comes from changes and uncertainties in the external environment of the organization. These interferences include market changes, changes in policies and regulations, economic environment changes, technological transformations, and natural disasters and emergencies.

From the data in Figure 6, it can be seen that under internal interference conditions, the proposed strategy management automated decision algorithm integrating LSTM and DQN exhibits significant performance advantages. Specifically, the proposed algorithm's initial score at time slot 0 is 0.2, and it shows a continuous and significant upward trend in subsequent slots, reaching 0.64, 0.72, 0.75, 0.79, and 0.82 at time slots 2, 4, 6, 8, and 10, respectively. In contrast, the traditional DQN algorithm's initial score is 0.14, and although it also shows improvement in the subsequent slots, the magnitude is smaller, scoring 0.58, 0.63, 0.64, 0.65, and 0.66 at time slots 2, 4, 6, 8, and 10, respectively. The random strategy, however, remains unchanged in all slots, consistently scoring 0.5, indicating its inability to cope with internal interference.

From the data in Figure 7, it can be seen that under external interference conditions, the proposed strategy management automated decision algorithm integrating LSTM and DQN also exhibits significant performance advantages. Specifically, the proposed algorithm's initial score at time slot 0 is 0.3, and it continuously improves in subsequent slots, reaching 0.36, 0.37, 0.39, 0.418, and 0.465 at time slots 2, 4, 6, 8, and 10, respectively, showing a stable upward trend. In contrast, the traditional DQN algorithm's initial score is 0.2, but in the subsequent slots, the score fluctuates slightly with almost no substantial improvement, scoring 0.18, 0.18, 0.18, 0.18, and 0.185 at time slots 2, 4, 6, 8, and 10, respectively. The random strategy's initial score is 0.4, and in time slots 2, 4, 6, 8, and 10, its scores are 0.405, 0.41, 0.41, 0.41, and 0.41, respectively, indicating a relatively stable but flat growth trend.

From the data in Figure 8, it can be seen that under external interference conditions, the decision success count of the proposed algorithm integrating LSTM and DQN is significantly better than that of the traditional DQN algorithm and the random strategy. Specifically, the proposed algorithm's initial success count at time slot 0 is 0, and it continuously increases in subsequent slots, reaching 2000, 4000, 6000, 8000, and 10000 times at time slots 2, 4, 6, 8,

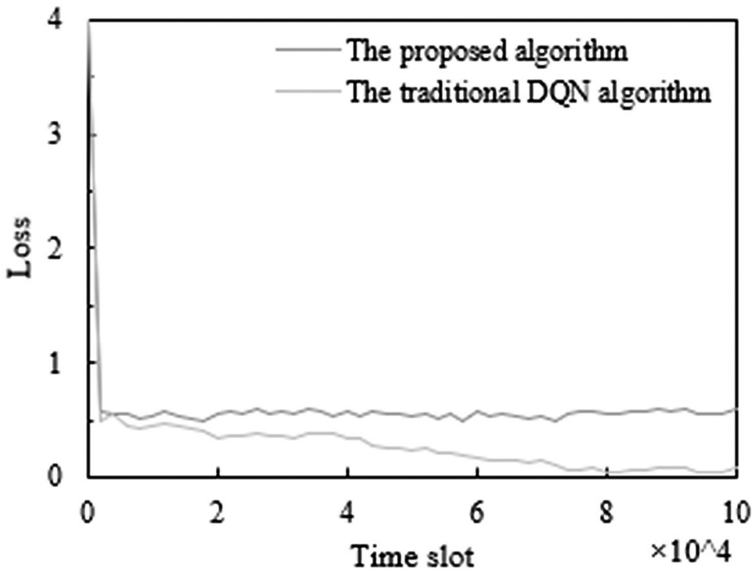

Figure 6. Performance comparison of different decision algorithms under internal interference.

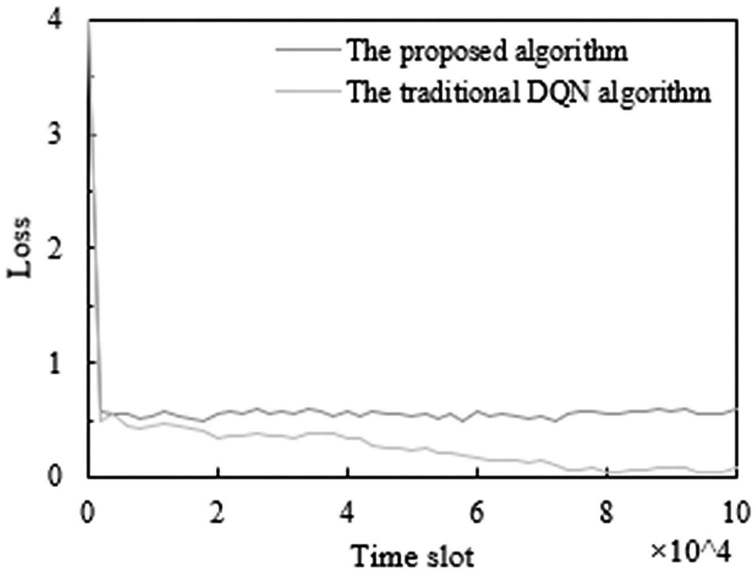

Figure 7. Performance comparison of different decision algorithms under external interference.

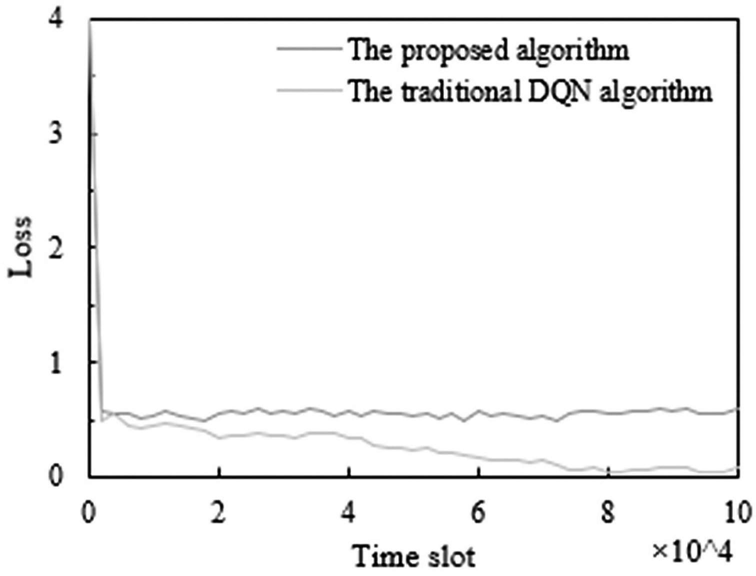

Figure 8. Comparison of successful decision counts of different decision algorithms under external interference.

and 10, respectively. In contrast, the traditional DQN algorithm's initial success count at time slot 0 is also 0, but it grows more slowly in subsequent slots, reaching 1400, 2800, 4200, 5600, and 7000 times at time slots 2, 4, 6, 8, and 10, respectively. The random strategy's initial success count is also 0, and in time slots 2, 4, 6, 8, and 10, the success counts are 1000, 2000, 3000, 4000, and 5000 times, respectively, showing a relatively stable but slow growth trend. Comprehensive analysis of these results allows us to conclude that the proposed algorithm integrating LSTM and DQN significantly improves decision success counts when dealing with internal and external interferences, outperforming the traditional DQN algorithm and the random strategy. These results indicate that the proposed algorithm integrating LSTM and DQN can effectively enhance decision success counts under external interference conditions, demonstrating stronger adaptability and optimization capabilities, providing more reliable support for strategy management in complex environments.

From the data in Figure 9, it can be further observed that the proposed algorithm integrating LSTM and DQN demonstrates significant advantages in scenarios with multiple organizational members. Specifically, the performance of the proposed algorithm gradually increases from 0.26 at time slot 0, reaching 0.4 at time slot 5, 0.62 at time slot 10, and finally 0.72 at time slot 15. In contrast, the performance of the traditional DQN algorithm starts at 0.24 at time slot 0, showing a slower improvement trend, with 0.14 at time slot 5, 0.24 at time slot 10, and reaching 0.28 at time slot 15. On the other hand, the random strategy maintains a performance of 0.58 throughout all time slots, indicating its lack of optimization capability over time.

From the data in Table 1, it can be seen that the proposed algorithm integrating LSTM and DQN exhibits a higher decision success probability under different numbers of organizational members. Specifically, the proposed algorithm's decision success probability is 0.941, 0.945, 0.923, 0.908, and 0.912 for 2, 5, 10, 15, and 20 organizational members, respectively, being significantly higher than that of the traditional DQN algorithm and the random strategy. For 2 members, the traditional DQN algorithm's decision success probability is 0.765, while for the random strategy is 0.521. As the number of organizational members increases, the traditional DQN algorithm's decision success probability gradually decreases, dropping to 0.378 for 20 members, whereas the random strategy's success probability remains relatively stable, with a slight upward trend from 0.521 to 0.591.

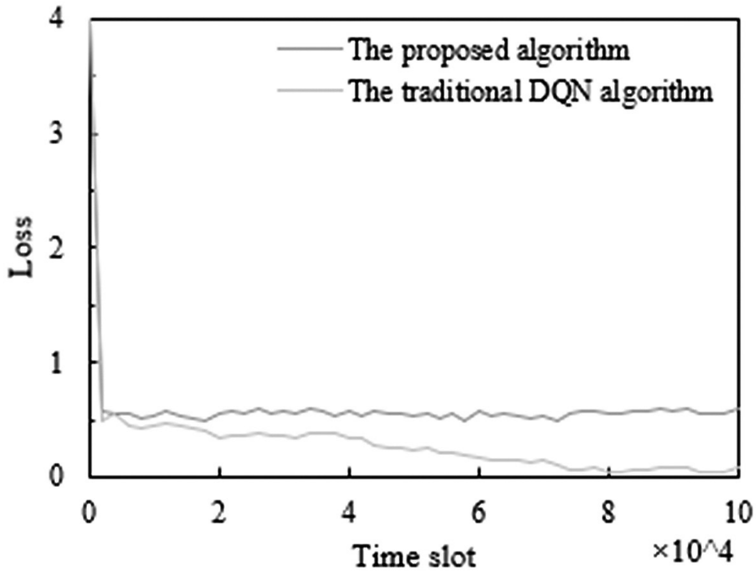

Figure 9. Performance comparison of different decision algorithms in multi-organizational member scenarios.

Table 1. Decision success probability statistics table.

Group	Number of organizational members	Total decisions	Decision success probability		
			The proposed algorithm	**Traditional DQN algorithm**	**Random strategy**
1	2	5	0.941	0.765	0.521
2	5	10	0.945	0.489	0.536
3	10	20	0.923	0.423	0.568
4	15	30	0.908	0.405	0.589
5	20	40	0.912	0.378	0.591

Comprehensive analysis of these results leads us to the following conclusions: the algorithm proposed in this paper, which integrates LSTM and DQN, significantly improves decision-making success rates across different scales of organizational members, outperforming the traditional DQN algorithm and random strategy. This indicates that by capturing important features in time series through the LSTM network and optimizing strategies with DQN, it can effectively handle complex organizational data and adapt to the management needs of different scales, thereby enhancing the accuracy and effectiveness of decision-making.

In addition to the comparison of quantitative metrics, the experimental results also indicate that the integrated LSTM and DQN algorithm exhibits strong adaptability in handling time-dependent features, providing stable decision support across organizational environments of varying scales. Furthermore, the algorithm demonstrates high robustness when dealing with external interference, maintaining a high decision success rate. Compared to the traditional DQN, the proposed algorithm shows better scalability, making it suitable for more complex environments and changing conditions, highlighting its flexibility and practicality.

5. Conclusion

The research in this paper focuses on two main aspects: the design and construction of a system model, and the development and application of an automated decision-making algorithm for strategy management. First, this paper establishes a flexible and scalable decision support framework capable of effectively handling complex organizational data and adapting to different management needs. Second, it proposes an automated decision-making algorithm for strategy management that integrates LSTM and DQN algorithms, leveraging the time series feature capture ability of the LSTM network combined with the strategy optimization function of the DQN algorithm, thereby significantly improving decision accuracy and efficiency. Through this research, innovative technical means are provided for strategy management in organizational activities, laying a theoretical and practical foundation for the future development of intelligent decision support systems.

Experimental results indicate that the proposed algorithm integrating LSTM and DQN performs excellently under various conditions. In scenarios with different scales of organizational members, the decision success probability of the algorithm is significantly higher than that of the traditional DQN algorithm and random strategy, demonstrating its adaptability and efficiency in complex decision environments. Additionally, the comparison analysis of error function curves, algorithm performance under internal and external interferences, and decision success counts further validate the stability and robustness of the algorithm under various interference conditions. The research findings demonstrate that the proposed integrated LSTM and DQN algorithm performed exceptionally well under various conditions, with a significantly higher decision success rate than traditional methods and high stability and robustness under interference conditions. The contribution of this paper lies in providing a novel algorithmic solution for strategy management in complex decision-making environments, advancing the application of time series analysis and reinforcement learning in strategic management. The proposed integrated LSTM and DQN algorithm offers an innovative solution for automated decision-making in strategy management, enabling efficient and precise decisions in complex and dynamic environments. This not only enhances the practical value of decision support systems in organizational management but also provides a flexible framework to meet the management needs of organizations of different scales. Additionally, this study enriches the application of machine learning algorithms in strategic management, promotes theoretical development of combining time series feature extraction with reinforcement learning, and provides new perspectives for future algorithm improvements and applications.

The limitations of the model are primarily reflected in the following aspects: 1) The combination of LSTM and DQN algorithms may have high computational resource requirements and long training times when handling large-scale complex data, particularly in real-world organizational management environments where data volumes are vast, and real-time processing is required. 2) LSTM may face the problem of gradient vanishing when capturing time series features in long sequences, which can lead to certain long-term dependencies not being effectively captured. Additionally, DQN's strategy optimization process relies on high-quality state-action pair data, and if the data is not rich or diverse enough, the model's decision-making effectiveness may be suboptimal. 3) The model is generally suitable for relatively stable organizational environments, where the input data needs to have clear time dependencies and sufficient historical records for the LSTM to function effectively. In highly dynamic and rapidly changing environments, the model may struggle to adapt.

Future research could further explore the applicability of this algorithm in larger-scale and more complex organizational environments, particularly in the context of real-time data processing and big data. Additionally, it could investigate how to incorporate more external factors, such as market changes and competitive dynamics, into the algorithm to further enhance its adaptability and accuracy in decision-making. Future studies might also consider combining other deep learning algorithms with reinforcement learning to further optimize the model's efficiency and robustness, expanding its application scenarios in other fields.

Declaration of Competing Interests

The authors declare no conflict of interest.

Acknowledgement

This paper was supported by Guangdong Province Philosophy and Social Sciences 13th Five Year Plan Project (Grant No: GD20CSH09), Guangzhou Civil Aviation College 2024 Research Backbone Project (Grant No: 24X4306), 2023 Campus level Quality Engineering Education and Teaching Projects of Guangzhou Civil Aviation College (Grant No: JG202309), Guangzhou Civil Aviation College 2023 College level College Students Innovation and Entrepreneurship Project (Grant No: CY23017 and CY23011) and 2024 Guangdong Provincial Education Planning Project (Higher Education Special) (Grant No: 2024GXJK175).

Data Availability

Data used in this article is available upon request from the authors.

References

[1] A. D. Vorobyov *et al.*, "A Unified Methodology of Strategic Management and a Knowledge Management Model", *TEM Journal-Technology Education Management Informatics*, vol. 8, no. 2, pp. 554–563, 2019. https://doi.org/10.18421/TEM82-31

[2] L. Smidt *et al.*, "Current Use of the Risk Register to Integrate Strategy and Risk- and Performance Management: A Case of a University of Technology in South Africa", *Journal of Accounting, Finance and Auditing Studies*, vol. 8, no. 4, pp. 140–171, 2022. https://doi.org/10.32602/jafas.2022.031

[3] S. Liu *et al.*, "Control Mechanisms and Senior Management Commitment in Shaping the Performance of Strategic Information Systems Projects: A Resource Dependence Perspective", *IEEE Transactions on Engineering Management*, vol. 71, pp. 5197–5211, 2022. https://doi.org/10.1109/TEM.2022.3214545

[4] B. Novarlić *et al.*, "Natural Hazards and Their Environmental Impact: Flood Risks in the Systemic Management of Non-Hazardous Municipal Waste", *Opportunities and Challenges in Sustainability*, vol. 3, no. 2, pp. 96–107, 2024. https://doi.org/10.56578/ocs030203

[5] M. Alnoukari, "A Framework for Big Data Integration Within the Strategic Management Process Based on a Balanced Scorecard Methodology", *Journal of Intelligence Studies in Business*, vol. 11, no. 1, pp. 33–47, 2021. https://doi.org/10.37380/jisib.v1i1.693

[6] A. A. Adepeju *et al.*, "Influence of Client Relationship Management Strategy on Organisational Performance", *International Journal of Sustainable Development and Planning*, vol. 19, no. 8, pp. 3165–3174, 2024. https://doi.org/10.18280/ijsdp.190829

[7] K. Y. Lin and L. Hu, "Supply and Demand Optimization of Agricultural Products in Game Theory: A State-of-the-Art Review", *Journal of Engineering Management and Systems Engineering*, vol. 1, no. 2, pp. 76–86, 2022. https://doi.org/10.56578/jemse010205

[8] C. Song *et al.*, "A Review of Optimal Energy Management Strategies Using Machine Learning Techniques for Hybrid Electric Vehicles", *International Journal of Automotive Technology*, vol. 22, pp. 1437–1452, 2021. https://doi.org/10.1007/s12239-021-0125-0

[9] H. Peng *et al.*, "Machine Learning-based Control for Fuel Cell Hybrid Buses: From Average Load Power Prediction to Energy Management", *Vehicles*, vol. 4, no. 4, pp. 1365–1390, 2022. https://doi.org/10.3390/vehicles4040072

[10] M. E. Wu *et al.*, "Kelly-based Options Trading Strategies on Settlement Date via Supervised Learning Algorithms", *Computational Economics*, vol. 59, no. 4, pp. 1627–1644, 2022. https://doi.org/10.1007/s10614-021-10226-2

[11] R. Ochoa-Barragán *et al.*, "A Hybrid Machine Learning-mathematical Programming Optimization Approach for Municipal Solid Waste Management During the Pandemic", *Environment Development and Sustainability*, vol. 26, no. 7, pp. 17653–17672, 2024. https://doi.org/10.1007/s10668-023-03354-2

[12] M. Andrejić and V. Pajić, "Optimizing Personnel Selection in Transportation: An Application of the BWM-CoCoSo Decision-Support Model", *Journal of Organizations, Technology and Entrepreneurship*, vol. 1, no. 1, pp. 35–46, 2023. https://doi.org/10.56578/jote010103

[13] R. Bloomfield *et al.*, "Disruptive Innovations and Disruptive Assurance: Assuring Machine Learning and Autonomy", *Computer*, vol. 52, no. 9, pp. 82–89, 2019. https://doi.org/10.1109/MC.2019.2914775

[14] F. Kitsios and M. Kamariotou, "Artificial Intelligence and Business Strategy Towards Digital Transformation: A Research Agenda", *Sustainability*, vol. 13, no. 4, 2025, 2021. https://doi.org/10.3390/su13042025

[15] I. Nurcahyani and J. W. Lee, "Role of Machine Learning in Resource Allocation Strategy over Vehicular Networks: A Survey", *Sensors*, vol. 21, no. 19, 2021. https://doi.org/10.3390/s21196542

[16] C. Li *et al.*, "A Study of Fuzzy Modeling Analysis of Factors Influencing Socially Regulation of Learning Performance in an Online Environment", *Journal of Intelligent & Fuzzy Systems*, vol. 41, no. 3, pp. 4639–4649, 2021. https://doi.org/10.3233/JIFS-189724

[17] D. P. Sousa *et al.*, "Leakage Detection in Water Distribution Networks Using Machine-learning Strategies", *Water Supply*, vol. 23, no. 3, pp. 1115–1126, 2023. https://doi.org/10.2166/ws.2023.054

[18] F. Boumaza *et al.*, "An Improved Harris Hawks Optimization Algorithm Based on Bi-Goal Evolution and Multi-leader Selection Strategy for Multi-objective Optimization", *Ingénierie des Systèmes d'Information*, vol. 28, no. 5, pp. 1135–1150, 2023. https://doi.org/10.18280/isi.280503

[19] E. Karaaslan *et al.*, "A Novel Decision Support System for Long-term Management of Bridge Networks", *Applied Sciences-Basel*, vol. 11, no. 13, 2021. https://doi.org/10.3390/app11135928

[20] D. C. Le and N. Zincir-Heywood, "A Frontier: Dependable, Reliable and Secure Machine Learning for Network/system Management", *Journal of Network and Systems Management*, vol. 28, no. 4, pp. 827–849, 2020. https://doi.org/10.1007/s10922-020-09512-5

[21] M. A. Dudhedia *et al.*, "Impact of Strategy Optimization on Game-based CR-MAC Protocol Performance", *Ingénierie des Systèmes d'Information*, vol. 27, no. 3, pp. 473–478, 2022. https://doi.org/10.18280/isi.270314

Received: August 2024
Revised: September 2024
Accepted: September 2024

Contact address:
Meiling Lu
School of Business Administration
Guangzhou Civil Aviation College
Guangzhou
China
e-mail: 10000550@gcac.edu.cn

MEILING LU graduated from Shihezi University in 2009 with a master's degree. She is currently a lecturer in Big Data and Accounting at Guangzhou Civil Aviation College. Her main research interests include digital economy and intelligent digital finance.