Research Article

Path Recognition of the Regional Education Expansion Based on Improved Dragonfly Algorithm

Fang Liu (ID)

Scientific Research Department, Xijing University, Xi'an 710123, China

Correspondence should be addressed to Fang Liu; 20020041@xijing.edu.cn

To solve the problems of low recognition rate, high misrecognition rate, and long recognition time, the path recognition method of the regional education scale expansion based on the improved dragonfly algorithm is proposed. Through a variety of different behaviors utilized in the optimization process, the dragonfly algorithm model has been constructed. The step size and the position vector are introduced to update the dragonfly's location. The dragonfly's foraging behaviors are accurately simulated. Afterward, the dragonfly algorithm is combined with the flower authorization algorithm. The conversion probability is added, and the dragonfly's global development ability is adjusted in real-time. Then, the dragonfly algorithm is improved. The improved dragonfly algorithm is employed to extract the features of the expansion path of the regional education scale. The improved support vector machine is utilized as a classifier to realize the recognition of the regional education scale expansion path. The experimental results denote that the proposed method has a high recognition rate of the regional education scale expansion path and can effectively reduce the misrecognition rate and shorten the recognition time.

1. Introduction

China's higher education has been currently in a brand-new period from scale expansion to connotation enhancement. Because of the obvious differences occurring in the development of higher education in various regions, it is very significant to recognize the path of the regional education scale expansion [1]. Research has been conducted in this field. For example, Dong and Zhang [2] analyzed the diffusion process of the subject and integrated network topology analysis and subject semantic mining for path recognition, which could effectively improve the accuracy of the path recognition research, but it had a longer recognition time. Cai et al. [3] introduced particle filter technology into the field of path deception recognition. They constructed corresponding state equations and observation equations, utilized dynamic data to extract key parameters, and then achieved path recognition. Their method could effectively improve the recognition efficiency, but the feature selection was not considered, which led to a high misrecognition rate of this method. Dealing with the above problems, a path recognition method for the expansion of the regional

education scale based on the improved dragonfly algorithm is proposed in this manuscript.

According to the many different behaviors utilized in the optimization process of the dragonfly algorithm, the dragonfly algorithm model is constructed. Then, the step length and position vector are introduced. The position of the dragonfly is updated, and the foraging behavior of the dragonfly is simulated. Combining with the flower authorization algorithm, the conversion probability is added to conduct global search work. Thus, the dragonfly algorithm is improved. The improved dragonfly algorithm is employed to extract the features of the expansion path of the regional education scale, and the improved support vector machine is utilized as a classifier to recognize the expansion path of the regional education scale. The simulation experiment results show that the proposed method can effectively reduce the misrecognition rate and the recognition time while improving the recognition rate.

The rest of the manuscript is organized as follows. Section 2 gives detailed information on both the dragonfly algorithm and flower authorization algorithm. Then, the improved algorithm has been proposed. Section 3 deals with the implementation of the proposed method. Section 4 is

allocated to present the simulation experiments using different datasets. The conclusion is presented in Section 5.

2. Improved Dragonfly Algorithm

2.1. Dragonfly Algorithm. Compared with most intelligent optimization algorithms, the dragonfly algorithm has the global search capability as the foraging behavior of the dragonfly; the local exploration capability is the migration behavior of the dragonfly [4, 5]. The dragonfly optimization algorithm contains many different behaviors in the optimization process.

The first behavior is called collision avoidance behavior. When a dragonfly flies, safety issues need to be prioritized as it needs to keep a safe distance from other dragonflies. The second behavior is called team behavior. Dragonflies are also animals living in groups. Therefore, dragonflies will move in groups in the entire population. To avoid collisions, it is necessary to maintain an appropriate flight distance between each dragonfly [6]. The third behavior is called gathering behavior. As a group, dragonflies have a leader. The dragonflies around the leader need to be close together, and the spacing between the dragonflies needs to be consistent [7, 8]. The fourth behavior is called foraging behavior. To get better and more food, dragonflies need to pass messages between them and keep moving closer to get more and better food. The last behavior is called enemy avoidance behavior. In the process of searching for food, dragonflies may be caught by other creatures [9]. Therefore, when a dragonfly sees its natural enemy, it will avoid it, but the probability of encountering a natural enemy is low.

The abovementioned behaviors are abstracted, and the corresponding algorithm model is formed at the same time. Among them, the displacement of a single dragonfly in the collision behavior can be calculated as follows:

$$S_i = -\sum_{j=1}^{N} X_i - X_j, \quad (1)$$

where S_i represents the displacement formed by the ith dragonfly individual in the collision behavior; X_i represents the current specific coordinate position of the ith dragonfly; X_j represents the iterative position of the ith dragonfly adjacent to the jth dragonfly in the dragonfly population; and N represents the population of dragonflies [10].

Equation (2) is used to give the displacement of a single dragonfly individual in the pairing behavior, namely,

$$A_i = \frac{\sum_{j=1}^{N} X_j}{N}, \quad (2)$$

where A_i represents the displacement formed by the ith dragonfly in the group.

Dragonflies also form displacements in their aggregation behavior [11, 12], and the specific expression is defined by

$$C_i = \frac{\sum_{j=1}^{N} X_j}{N} - X_i, \quad (3)$$

where C_i represents the displacement formed by the ith dragonfly in the group gathering behavior.

The displacement formed by the individual dragonfly during foraging is as follows:

$$F_i = X_{\text{food}} - X_i, \quad (4)$$

where F_i represents the displacement formed by the ith dragonfly in the foraging process and X_{food} represents the exact coordinate position of the food that the dragonfly needs to obtain under the set number of iterations [13–15].

When a dragonfly individual is avoiding the enemy, it will produce a certain displacement. The specific expression is as follows:

$$E_i = X_e + X_i, \quad (5)$$

where E_i represents the displacement formed by the dragonfly individual in the process of avoiding enemies and X_e represents the position of the natural enemy discovered by the dragonfly under the set number of iterations.

In the entire dragonfly algorithm, it is assumed that the entire foraging process of the dragonflies is a combination of the five behaviors mentioned above. To improve the recognition accuracy of the dragonfly algorithm, it is necessary to continuously update the specific position of the dragonflies in the iterative process and, at the same time, accurately simulate the behavior of the dragonflies in foraging. Thus, two vectors are introduced.

(1) Step vector ΔX

(2) Position vector ΔY

The above two vectors are mainly used to update the position of the dragonfly in the next iteration, where the step vector ΔX represents the flying position of the dragonfly individual in the current iteration; the step vector of the ith dragonfly in the $t + 1^{\text{th}}$ iteration is

$$\Delta x_{\text{id}}^{t+1} = \Delta Y \left(s S_{\text{id}}^t + a A_{\text{id}}^t + c C_{\text{id}}^t + f F_{\text{id}}^t + e E_{\text{id}}^t \right) + \Delta X, \quad (6)$$

where $\Delta x_{\text{id}}^{t+1}$ represents the d-dimensional step vector corresponding to the ith dragonfly during the $t + 1^{\text{th}}$ iteration process [16]; s represents the impact weight of the collision behavior; S_{id}^t represents the deviation of the corresponding d-dimensional due to the collision of the ith dragonfly individual during the ith iteration process; a represents the influence weight of team behavior; A_{id}^t represents the deviation of the ith dragonfly due to team behavior during the ith iteration; c represents the inertia weight; C_{id}^t represents the deviation of the corresponding d-dimensional of the ith dragonfly individual during the ith iteration due to inertia; f represents the influence weight of dragonfly position; F_{id}^t represents the deviation of the corresponding d-dimensional of the ith dragonfly individual during the ith iteration due to the position of the dragonfly; e represents the influence weight of the dragonfly individual; and E_{id}^t represents the corresponding d-dimensional of the ith dragonfly individual during the ith iteration due to the formation of the dragonfly individual's deviation.

Analyzing the influence weights of five different behaviors, we can know that the algorithm can adjust dragonfly's global development capability and local detective

capability in real-time during the process of obtaining the optimal solution [17, 18]. Neighboring individuals have a great influence on dragonflies. Therefore, in the entire algorithm, it is assumed that each single dragonfly individual has a certain radius neighbor.

When the step vector calculation is completed, the position of the dragonfly is updated by

$$X_{\text{id}}^{t+1} = X_{\text{id}}^t + \Delta X_{\text{id}}^{t+1}, \tag{7}$$

where the position update formula of the ith dragonfly is

$$X_{\text{id}}^{t+1} = X_{\text{id}}^t + \text{Levy}(d) \times X_{\text{id}}^t, \tag{8}$$

where the Levi flight calculation formula is

$$\text{Levy}(x) = 0.01 \times \frac{r_1 \times \sigma}{|r^2|^{1/p}}, \tag{9}$$

where $\text{Levy}(x)$ represents the Levi random path, σ represents the step size scaling factor, and p and r represent a random number subject to a uniform distribution.

2.2. Flower Authorization Algorithm.

The behavior of flowers using pollination to reproduce offspring is very common in nature. It is necessary to reproduce offspring through all the flowers, and the process of pollination must be combined with energy. Due to the certain differences between different flowers, the pollination methods of flowers also differ greatly. In nature, there exist all kinds of plants, and the flowers that occupy a larger proportion are blooming flowers. In addition to various extreme environments, there exist blooming flowers in all regions. Since most of the blooming flowers cannot exist, the entire pollination process of flowers requires the help of external forces such as bees and insects [19].

In the flower authorization algorithm, the number of pollens is set as N and the dimension of the solution space is D. When pollen is authorized for cross pollen, pollen mainly conducts global search and updates. When pollen is pollinated, local exploration work is mainly performed, and location updates are also required.

In the process of flower authorization, to effectively ensure the simple structure of the whole algorithm and strong optimization ability, the conversion probability p is added to the pollen authorization algorithm to balance and adjust strategy transformation. In the early stage of pollination, r, a random value, will be formed, and the value of r must conform to the law of uniform distribution.

2.3. Improvement of the Dragonfly Algorithm.

Combining the dragonfly algorithm with the flower authorization algorithm, we set the value range of each dragonfly during initialization to $[S_{\min}, S_{\max}]$ through the difference of the problem-solution space. To effectively avoid the situation where the dragonfly algorithm gathers together in the initial stage, the uniform random distribution is used in the initialization process to perform random initialization processing on each dimension of each dragonfly, namely,

$$x_{\text{id}}^0 = S_{\min} + (S_{\max} - S_{\min}). \tag{10}$$

After preprocessing, the two algorithms are further fused to guide the dragonfly to fly to a better position, thereby filling the shortcomings of the dragonfly algorithm.

3. Path Recognition of the Regional Education Expansion Based on Improved Dragonfly Algorithm

To better recognize the regional education scale expansion path, it is necessary to prioritize the feature selection of the regional education scale expansion path. If the value of any feature is 0, it means that the feature does not need to be selected; if the value corresponding to any feature is 1, then the feature is selected. The solution space of the problem can be transformed into two trends $\{0, 1\}$. To make the initial features have a certain difference, each dragonfly needs to obey the distribution as much as possible. In the improved dragonfly algorithm, one-dimensional initialization processing is performed on each dragonfly in the dragonfly population through

$$x_{\text{id}}^0 = \begin{cases} 0, & r < 0.5, \\ 1, & r \geq 0.5, \end{cases} \tag{11}$$

where x_{id}^0 represents the specific coordinate position of the ith dragonfly in the d-dimensional ith during the initial iteration, and the unit of r must conform to the random value of the uniform distribution rule.

After the above calculations, the position of the dragonfly can be updated in combination with related formulas, and a global search can be performed. At the same time, the support vector machine is set as the classifier. For the dataset $\{x_i, y_i\}$, d represents the total number of features in the feature set and y_i represents the total output of the dataset. Under normal circumstances, the support vector machine algorithm needs to set a linear decision function, and its specific expression is

$$f(x) = \langle w, \phi_i(x) \rangle + b, \tag{12}$$

where w represents the weight in the support vector machine; b represents a random constant [20]; and ϕ represents the nonlinear function that maps the input feature to a higher feature space. Equation (12) can be transformed into

$$R(C) = \frac{C}{N \sum_{i=1}^n L_s(f(x_i), y_i) + 1/2\|w\|^2}, \tag{13}$$

where $L_s(f(x_i), y_i)$ represents the loss function of ε. When $|f(x) - y| \geq \varepsilon$, $L_s(f(x_i), y_i)$ can be solved by equation (15), and the specific expression is

$$L_s(f(x_i), y_i) = |f(x) - y| - \varepsilon. \tag{14}$$

In the process of increasing the slack variables ξi and ξi^*, it is necessary to satisfy the three constraints of equation (15) at the same time and minimize the problem, as shown in equation (16)

TABLE 1: Comparison results of the recognition rate of the regional education scale expansion path by the different methods.

Dataset	The recognition rate of the expansion path of the regional education scale (%)		
	The proposed method	The method in [2]	The method in [3]
01	98.63	96.47	94.25
02	97.14	94.88	92.02
03	96.22	93.96	91.00
04	98.96	92.74	88.36
05	97.14	91.22	87.74
06	96.20	90.52	86.74

TABLE 2: Comparison results of the recognition time of the regional education scale expansion path by the different methods.

Data volume (MB)	Recognition time of the regional education scale expansion path (s)		
	The proposed method	The method in [2]	The method in [3]
20	1.25	3.41	2.36
40	2.34	5.47	4.42
60	3.37	7.52	5.48
80	4.23	9.41	7.37
100	6.10	10.32	9.23
120	8.20	16.32	12.27

$$\begin{cases} y_i - \langle w, x_i \rangle - b \le \varepsilon + \xi_i, \\ \langle w, x_i \rangle + b - y_i \le \varepsilon + \xi_i^*, \\ \xi_i, \xi_i^* \ge 0, \end{cases} \tag{15}$$

$$R(w, \xi_i^*) = \frac{1}{2}\|w\|^2 + C\sum_{i=1}^{n}(\xi_i + \xi_i^*). \tag{16}$$

Based on the above analysis, the regional education scale expansion path classification is carried out through the classifier, and the recognition is finally realized:

$$K(x_i, x_j) = \phi(x_i) \cdot \phi(x_j). \tag{17}$$

4. Simulation Experiment

To verify the comprehensive validation of the proposed method, a simulation experiment was performed on the Windows 10 64-bit system, with Python language as a programming environment.

4.1. The Recognition Rate of the Expansion Path of Regional Education Scale (%).
The experiment selected the recognition rate of the regional education scale expansion path as the test index and selected the methods in [2] and in [3] for comparison. The specific experimental comparison results are shown in Table 1.

Analyzing the experimental data in Table 1, it can be seen that, due to the different datasets, the recognition rate of the regional education scale expansion path of the different methods had changed to different degrees. However, when compared with the other two methods, the proposed method remained a relatively more stable and larger recognition rate of the regional education scale expansion path, while the other two methods showed a linear decline. The main reason was that

the proposed method used an improved dragonfly algorithm to optimize and recognize it so that it could accurately recognize each path and obtain a higher recognition rate.

4.2. Recognition Time of the Expansion Path of Regional Education Scale (s).
The following experiment mainly compared the recognition time of the regional education scale expansion path of the different methods. The specific experimental comparison results are shown in Table 2.

Analyzing the experimental data presented in Table 2, as the amount of test data increased, the recognition time of the regional education scale expansion path increased for all three methods. When the volume of the test data reached 120 MB, the recognition time of the regional education scale expansion path based on the method in [2] was 16.32 s, that of method in [3] was 12.2 7 s, and that of the proposed method was only 8.20 s. It could be seen that the proposed method had a shorter recognition time for the expansion of the regional education scale. This was because the proposed method combined the dragonfly algorithm with the flower authorization algorithm and added the conversion probability, thereby achieving the real-time adjustment of the dragonfly's global development capabilities and reducing the recognition time.

4.3. Misrecognition Rate (%).
To fully verify the superiority of the proposed method, the following experimental tests compared with the misrecognition rates of the different methods. The specific experimental comparison results are shown in Figure 1.

The results in Figure 1 show that, as the amount of test data increased, the misrecognition rate of the different methods increased. When the test data volume reached 120 MB, the misrecognition rate of the method in [2] was 15%, that of the method in [3] was 18.2%, and that of the proposed method was 8.3%. It could be seen that the miss-recognition rate of the proposed method was significantly
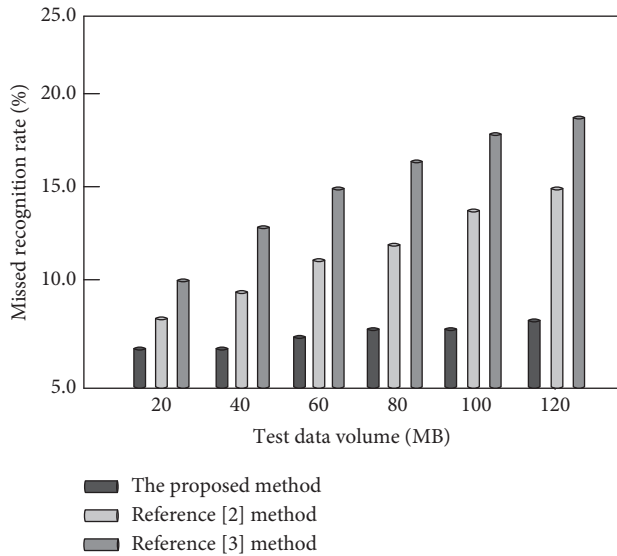
Figure 1: Comparison results of misrecognition rates of different methods.

lower than that of the other two methods, and it also verified the effectiveness and feasibility of adding the improved dragonfly algorithm to the path recognition of the regional education scale expansion.

5. Conclusion

To improve the recognition rate of the regional education scale expansion path, reduce the misrecognition rate, and shorten the recognition time, a method based on the improved dragonfly algorithm is proposed for the recognition of the regional education scale expansion path. Dragonfly algorithm combined with flower authorization algorithm leads to the improved dragonfly algorithm utilizing both step size and position vector and the conversion probability.

Therefore, the improved dragonfly algorithm could be used to extract the features of the expansion path of the regional education scale. Afterward, the improved support vector machine is utilized to recognize the expansion path of the regional education scale. The simulation experiment results show that the proposed method can effectively improve the recognition rate of the regional education expansion path and reduce the recognition time and the miss-recognition rate.

Data Availability

All data are available from the author upon reasonable request.

Conflicts of Interest

The author declares no conflicts of interest.

References

[1] B. Liu, C. Yang, and L. Xiong, "Path recognition method of picking robot in orchard natural environment," *Jiangsu Journal of Agriculture*, vol. 35, no. 5, pp. 1222–1231, 2019.

[2] K. Dong and B. Zhang, "Research progress on path recognition of subject knowledge diffusion network," *Information Studies: Theory & Application*, vol. 40, no. 8, pp. 139–144, 2017.

[3] Z. Cai, R. Ju, X. Xu, and S. Wang, "Deceptive path recognition based on particle filter technology," *Systems Engineering and Electronics*, vol. 42, no. 2, pp. 374–380, 2020.

[4] Z. Lin, "Path identification of internet entrepreneurship opportunities based on long tail theory," *China Adult Education*, vol. 10, no. 5, pp. 66–68, 2017.

[5] H. Wang, N. Wang, and J. Lian, "Vibration transmission path identification of hydropower plant based on transfer entropy," *Journal of Hydraulic Engineering*, vol. 49, no. 6, pp. 732–740, 2018.

[6] C. Ding and H. Xiao, "The characteristics and path identification of China's animal husbandry production fluctuations," *Research of Agricultural Modernization*, vol. 39, no. 2, pp. 256–264, 2018.

[7] J. Li, Y. Li, and J. Peng, "Research on path recognition method of agricultural robot visual navigation," *Computer Engineering*, vol. 44, no. 9, pp. 38–44, 2018.

[8] P. Deng, R. Yang, and S. Deng, "A new LPMD-based scattering path recognition algorithm," *Journal of Southwest Jiaotong University*, vol. 53, no. 1, pp. 182–188, 2018.

[9] S. Yan, G. Duan, and S. Zhang, "A device-independent indoor positioning algorithm based on line-of-sight path recognition," *Computer Engineering and Science*, vol. 40, no. 8, pp. 80–87, 2018.

[10] L. Zhao, C. Ye, Y. Zhang, X. Xu, and J. Chen, "Path recognition method of robot visual navigation in an unstructured environment," *Acta Optics*, vol. 38, no. 8, Article ID 0815028, 2018.

[11] H. Wang, W. Linjin, and Q. Zhang, "Path identification and change of the Kuroshio to the east of Taiwan," *Ocean & Limnology*, vol. 49, no. 2, pp. 271–279, 2018.

[12] Z. Mao and Q. Chen, "AGV multi-branch path recognition and tracking based on PCA-LDA and SVM," *Progress in Laser and Optoelectronics*, vol. 55, no. 9, pp. 148–155, 2018.

[13] J. Ding and X. Zhang, "Research on subway passenger path recognition based on mobile phone signaling data," *Journal of China Academy of Electronics*, vol. 14, no. 11, pp. 1194–1201, 2019.

[14] F. Wu, X. Chen, and L. Huang, "Research on technology multi-topic evolution path recognition method based on semantic similarity," *Journal of Information*, vol. 37, no. 5, pp. 91–96, 2018.

[15] S. Fan, X. An, and L. Shan, "Research on subject evolution type and evolution path identification method based on medical reference," *Information Theory and Practice*, vol. 42, no. 3, pp. 114–119, 2019.

[16] Y. Bai and L. Shi, "Spatial spillover of technological innovation in the Yangtze river economic belt: effect measurement, path recognition and cooperative amplification," *Journal of Wuhan University (Natural Science Edition)*, vol. 72, no. 1, pp. 122–135, 2019.

[17] X. Yu and J. Xu, "A bayesian recognition method of highway ambiguity path based on directed graph," *Highway and Transportation Science and Technology*, vol. 35, no. 11, pp. 110–115, 2018.

[18] H. Zou, R. Jia, and C. Zhang, "Research on freescale smart car track recognition and control strategy," *Machine Tool & Hydraulics*, vol. 46, no. 15, pp. 101–105, 2018.

[19] Y. Hou, X. Lv, Y. Chen, J. Zhao, Q. Li, and H. Chen, "Application of deep neural network in visual recognition of forest trails," *Journal of Computer Science and Exploration*, vol. 13, no. 2, pp. 263–274, 2019.

[20] Z. Qian, P. Li, and G. Zhou, "Uncertain and negative scope recognition based on two-way LSTM network," *Journal of Software*, vol. 29, no. 8, pp. 2427–2447, 2018.