ingegneria sismica
International Journal of Earthquake Engineering
Anno XLIII - Num. 1 - 2026

Application of Artificial Intelligence in Real-Time Strategic Decision Support System for Sports Competitions

Tao Chen[1,*]

[1] Luoyang Vocational College of Culture and Tourism, Luoyang, Henan, 471000, China

SUMMARY: *Various high-tech technologies are used extensively around the world in sports competitions especially in athletic events where scientific real-time decision-making is essential to improve competitive efficiency and match outcomes. This research proposes the introduction of the concept of entropy by applying the ID3 algorithm with the use of the attribute entropy value change as the selection criterion to develop the decision tree model for real-time sports competition data processing. Meanwhile, an enhanced Monte Carlo tree search algorithm can select the maximum UCT function node to ensure the optimal solution. The study shows that there is a percentage of players who have used advanced strategies in their basketball games ranging from 0 to 0.04. The developed decision system can make real-time strategy evaluation for sports competitions, taking a decision-making time of about 3.13 seconds on average. In addition, the system makes a 74% win rate and 84% decision rationality which indicates that the decision system is quite ideal and can serve as a good reference in future sports competition real-time decision-making practices.*

KEYWORDS: *ID3 algorithm; Monte Carlo tree search; UCT function; real-time strategy for sports competitions*

1 Introduction

The Real-Time Strategic Decision Support System for Sports Competitions combines data acquisition, intelligent analysis, and decision-making support. This system offers coaches, athletes, and event organizers real-time strategic optimization support and competition management. With the use of AI technologies, intelligent functionality is ensured in the system which is critical during pre-competition preparation, game adaptation, and post-event evaluation [1-4].

Pre-competition preparation and organization are stages where artificial intelligence allows organizing authorities to plan better [5, 6]. Using big data analytics, AI estimates the number of people that would attend the events, and accordingly plan seating, traffic flow, and catering services. Thus, it helps organize a transportation schedule to increase the transport carrying capacity, and allocate parking areas, which will prevent traffic jams [7, 8]. Meanwhile, AI gives advice to caterers regarding the preferences and eating behavior of spectators [9-11]. In the in-game adjustment phase, real-time AI-powered systems are used for real-time analysis and officiating [12, 13]. By collecting data on athletes' movements, speed, and positioning using fast cameras and sensors, an AI-powered real-time strategic decision-making system provides tactical advice to coaches and athletes. The latter can make changes in the team's strategy immediately, such as substitute one athlete for another,

*13939919251@163.com
https://doi.org/10.65102/is2026367
Published: 30 April, 2026

increasing their chances of winning [14-16]. The system may also predict outcomes and perform post-match analyses. It uses different information types including past performance, injuries, and other factors to predict outcomes [17]. Systematic post-match analyses help players understand better how the game ended, learn from mistakes, and be ready for future games [18-22].

AI advancement has made tremendous impacts in all aspects of human society. In sports competitions, AI technology makes important contributions to ensuring the fairness of the sport, prediction of athletic performance, and development of the sports industry. As mentioned in Reference [23], AI technology is used to optimize the organization of important international martial art contests such as World Wushu Championships and World Youth Wushu Championships. Besides promoting international friendship, these practices help spread Chinese martial art culture around the world. They promote innovative development in the martial art contest industry. Reference [24] discusses the importance of AI technology in sports competitions with instant replay systems as an example. Instant replay systems use high-speed cameras to capture footage from the matches. They eliminate any possible misjudgment in the sport event and ensure objectivity and fairness of the competition results. Reference [25] analyzes the significance of AI technology in decision-making on site and immediate display of basketball game information in sports competitions. The article creates an AI-based system for decision-making on site in basketball matches in order to investigate the application of AI technology in basketball games and validate the system's effectiveness. Reference [26] explains AI technology applications in sports, focusing on the impact of the technology on sports competition fairness and strategic decision-making by teams based on AI analysis.

Reference [27] introduced an automated scoring system based on artificial intelligence technology, which combines convolutional neural networks and long short-term memory networks, to identify complicated game moves, record athlete information, and score accurately and fairly, thus increasing scoring efficiency and impartiality. Reference [28] illustrated applications of artificial intelligence in sports events management, revealing its efficacy in improving the arrangement of competitions and resources and training athletes using specialized training programs. Reference [29] created an automatic evaluation system based on artificial intelligence to identify passing violations in sports, which was proven efficient by performing actions such as training classification models, acquiring crucial judgment frames, and creating state-judgment models. Reference [30] initiated research into artificial intelligence and big data applications in sports event services, stressing that in the information era, such technologies could effectively boost China's capacity for providing high-quality services in sports events not only in terms of fulfilling user needs but also in promoting the growth of the sports sector. Reference [31] developed a program based on artificial intelligence technology to forecast the performance of wrestling athletes. The results indicated that neural networks and machine learning algorithms would help improve the quality of athlete selection, allowing for individualized training procedures and increased training outcomes for young wrestlers. Reference [32] discussed the mechanisms, importance, and ways of implementing artificial intelligence in developing professional sports events from three angles, including how AI technology operates, its practical worth, and its specific practices.

This research work will focus on classifying the important aspects of sports information and summarizing the nature and components that define sports competitions. This study presents an approach for developing real-time sports event decision support systems, including three main components: data, business, and presentation. For developing decision tree models, an enhanced version of ID3 algorithm is used. The proposed models are further incorporated with the help of firefly algorithm for analyzing the real-time data associated with sports competitions. Combining Monte Carlo trees with policy value networks, UCT selection function is used as the ideal approach to create the real-time decision support results for the sports competitions. Real-time data analysis of sports competitions is carried out by measuring the indicators such as strategies used and strategies involved.

2 Preliminary Concept for a Real-Time Decision Support System in Sports Competitions

2.1 The Importance of Sports Information in Sports Competition Decision-Making

2.1.1 Basic Concepts and Classification of Sports Information

Definition of sports information Sports information can be defined as the generic name of the various types of information, materials, and data created in the process of competitive sports activities. The main content of sports information mainly consists of athlete information, training information, game information, and management information. From the perspective of information category, it can be divided into the following types: 1. Basic information (such as information about athletes and information about physical fitness) 2. Training information (such as training plan, training volume, recovery situation, etc.) 3. Game information (such as tactics, performance evaluation, etc.) 4. Management information (such as manpower management).

2.1.2 Characteristics and Elements of Competitive Sports Decision-Making

Competitive sports decision-making involves time sensitivity, complexity, risk, and systemics [33]. Time sensitivity reflects the necessity of quick decisions and reactions. Complexity occurs because of a number of impacting factors. Risk appears because of potential loss that can result from incorrect decision-making. Finally, systemics reflects the fact that different elements should be considered when making decisions. Main elements of sports decision-making include: decision-makers (coaches, managers, etc.); decision subjects (athletes, training programs, strategies, etc.); decision environment (internal and external environments); and decision bases (sporting information and other data sources).

2.1.3 Primary Applications of Current Sports Information in Decision-Making

In general, the utilization of sports information in decision-making for competitive sports mainly revolves around three fields, namely training decision-making, competition decision-making, and management decision-making. In terms of training decision-making, its main applications include developing training programs, regulating training intensity, analyzing the training effectiveness, and preventing sports injury. For competition decision-making, it mainly serves as the basis of strategy analysis and formulation, collecting opponent information, implementing real-time decision support, and summarizing experience after an event. In management decision-making, the application includes personnel selection and development, optimizing resources distribution, logistics support management, and medical rehabilitation services. The above mentioned methods for application constitute a relatively perfect decision support system.

2.1.4 Trends in the Development of Information Technology in Competitive Sports Decision-Making

Along with the fast development of information technology, there are some new development trends regarding the application of information in competitive sports decision-making. Technologically, these trends are mainly embodied in the more advanced big data analytics, wide adoption of artificial intelligence-assisted decision-making, extensive use of wearable devices, as well as novel application of virtual reality technology. In terms of applications, the main trends include intelligence, timeliness, accuracy, and comprehensiveness. Future developments can be further advanced along such directions as constructing intelligent decision-making platform, integrating multiple information sources, designing personalized decision-making solutions, and predicting analytics capability improvement.

2.2 Preliminary System Concept

The sports competition real-time decision support system has three layers: the data layer, the business layer, and the presentation layer. The data layer acts as the foundational support platform supplying data to the business layer. The business layer plays the role of the core execution layer executing business logic processing. The presentation layer presents the information in the form of visualized graphics. The structure of the sports competition real-time decision support system is depicted in Figure 1 below.

The data layer mainly focuses on data gathering and construction of a data warehouse. A data warehouse refers to the effective integration of many different types of data sources, which have been reorganized according to specific themes to enable decision-making and analytical data processing. In the sports competition real-time decision support system, the following data are collected and stored in a data warehouse using data acquisition: physiological parameters (physical condition, injury state, specialties, etc.), training data (personal record, average score of training, field position, etc.), data on athletes during competitions (historical records of competing performances, direct scores from the games, indirect scores, assistive offense, etc.) and strategy data.

The statistical report in the presentation layer uses tables and graphs to show and compare various forms of data in a wide range of perspectives and levels. This may include personal and coach data of each player, various training and match statistics. Decision analysis in this layer is done through creating particular themes in data analysis, identifying the optimal solutions to provide assistance for the coach in making decisions. The electronic sand table in the presentation layer uses digital technology to simulate and present positioning of the two teams and their routes for offense and defense on the field.

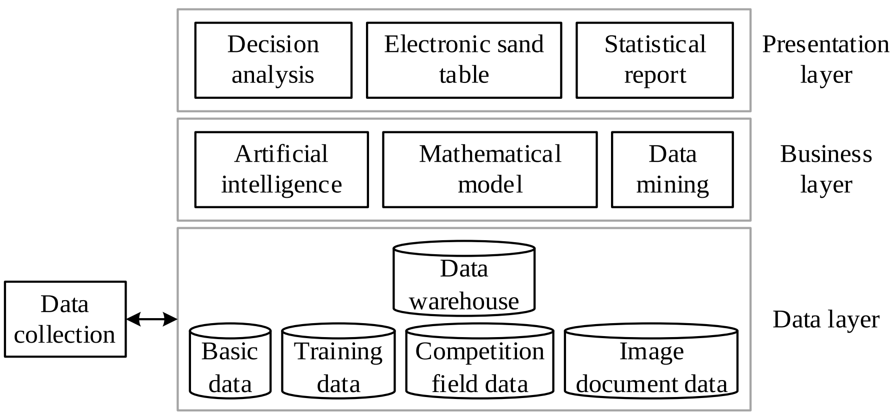


Figure 1: The real-time decision support system framework of sports competition

3 System Detailed Design and Implementation

3.1 Real-Time Data Processing for Sports Competitions

3.1.1 ID3 Algorithm and Its Improvements

The core of the ID3 algorithm lies in introducing the concept of entropy, where the magnitude of entropy values determines the stability of attribute features. This approach uses the degree of entropy variation in attributes as the node selection rule to construct decision trees [34].

The subsequent introduction to the ID3 algorithm includes relevant concepts and formula calculations.

Assume D is a training sample dataset containing d training data points, where the class label attribute can be categorized into n classes. Each individual category can be represented as D_t, and the sample data within each category can be represented as d_i. The probability of each category occurring is $p(i)=\frac{d_i}{d}$. The expected information for the final sample classification is:

$$I(d_1,d_2,...d_n)=-\sum_{i=1}^{n}p(i)\log_2 p(i) \tag{1}$$

Next, assuming attribute B has k distinct values, the sample data D can be partitioned based on the values of attribute B into k distinct subsets: $\{B_1,B_2,...B_k\}$, Each category is denoted as $B_j(1\le j\le k)$. Let the number of samples in subset B_j within category D_i be d_{ij}. The number of samples in the jth category after classification by attribute B is d_j. The probability that the jth category belongs to the ith class label is $p(ij)=d_{ij}/d_j$. Finally, the expected information formed after classification by attribute B is:

$$I(d_{1j},d_{2j},...d_{nj})=-\sum_{i=1}^{n}p(ij)\log_2 p(ij) \tag{2}$$

The information entropy of attribute B is then derived as:

$$E(B)=-\sum_{j=1}^{k}\frac{s_j}{s}I(d_{1j},d_{2j},...d_{nj}) \tag{3}$$

The information gain for attribute B is:

$$Gain(B)=I(d_1,d_2,...d_n)-E(B) \tag{4}$$

Based on the formula listed above, the value of Gain(xxx attribute) can be calculated. A higher value indicates greater stability and accuracy when partitioning based on that attribute, thereby identifying the attribute feature node most suitable for sample partitioning at the current stage. By repeatedly performing the above calculation until sample partitioning concludes, a top-down decision tree can be constructed.

3.1.2 Firefly Algorithm

The Firefly Algorithm derives its name from its conceptual resemblance to the aggregation process of firefly swarms. It is frequently employed to search for optimal solutions to problems: each feasible solution can be regarded as an individual firefly, with the solution's fitness mapped to the firefly's luminous intensity—the stronger the fitness, the greater the light intensity [35]. In the firefly swarm, the fireflies that have lower luminous intensity keep moving towards the ones that have higher luminous intensity. In the iterations, the position data keeps changing. Finally, all the solutions gravitate around the best solution, thus making the process complete.

The Firefly Algorithm works on the following assumptions:

(1) All fireflies are gender-neutral, meaning each firefly can attract other nearby individuals.

(2) A firefly's attractiveness is determined by its brightness and relative distance. Brighter fireflies attract dimmer ones nearby, while the brightest firefly flies randomly. Attraction gradually diminishes as the relative distance between two fireflies increases.

(3) A firefly's brightness is determined by its fitness function for the corresponding problem. Greater fitness corresponds to greater brightness.

Under these assumptions, the following relevant formulas can be derived:

(1) The relative brightness of a firefly is expressed as follows:

$$I = I_0 e^{-\gamma r} \tag{5}$$

In the equation, I_0 denotes the maximum brightness value, γ is a constant parameter representing the degree of light decay, and r represents the relative distance between any two fireflies.

(2) The attractiveness of a firefly is calculated as follows:

$$\beta = \beta_0 e^{-\gamma r^m} \tag{6}$$

In the formula, β_0 represents the maximum attractiveness, i.e., the attractiveness at the closest point (distance r=0).

(3) The distance between two fireflies is determined by the Cartesian distance, calculated using the following formula:

$$r_{ij} = \sqrt{\sum_{p=1}^{d} (X_i, p - X_j, p)^2} \tag{7}$$

In the formula, x_i denotes the i-th firefly, while p represents several component values of the spatial coordinates.

(4) The position update formula between fireflies is as follows:

$$x_i(t+1) = x_i(t) + \beta_0 e^{-\gamma r_{ij}^2} (x_j(t) - x_i(t)) + \alpha(rand - 0.5) \tag{8}$$

In the equation, X corresponds to the coordinate value in position space, α represents the range parameter indicating the movement of the firefly word, and rand denotes the function operation for generating random values.

3.2 Real-Time Policy Support Algorithms

3.2.1 Strategy Value Network Model

(1) DenseNet Network Architecture

In the task of generating sports competition strategies, the policy network outputs probabilities for 2048 strategy classes, while the value network outputs probabilities for 17 scoring classes. Given the large number of categories, the network architecture capable of meeting these classification requirements must not only be deep but also effectively extract input features. As such, a densely connected convolutional neural network is presented for consideration. The training error increases dramatically for conventional convolutional neural networks on several data sets when the depth of the networks exceeds a particular point. Gradient propagation through backpropagation becomes less efficient in later layers due to activation functions, resulting in the vanishing gradient effect and ineffective training.

The residual network solves the vanishing gradient problem without compromising the depth of the network. This is done by incorporating an identity mapping in the conventional neural network, whereby the information at the current layer is transferred directly to the subsequent layer. This part of the output ignores computation performed in the current layer. This direct pathway is referred to as a “skip connection,” a typical activation function in neural networks, expressed by formula (9):

$$f(x) = \max(0, x) \tag{9}$$

Residual networks use skip connections in backpropagation such that gradients can flow directly from the succeeding layer to the previous layer. Network architectures utilizing residual blocks can be extended to hundreds or even thousands of layers without suffering from network degradation, thereby achieving superior classification performance.

DenseNet, a dense-connected convolutional network, builds upon the residual network model. By increasing the number of layers while incorporating feature reuse and bypass connections, DenseNet significantly reduces the number of parameters while mitigating the gradient vanishing problem to some extent. DenseNet establishes dense connections between all preceding layers and subsequent layers, as illustrated in Figure 2, hence its name. The input to layer i is not only related to layer i-1 but also to the outputs of all preceding layers, as follows:

$$X_i = H_i([X_0, X_1, ..., X_{i-1}]) \tag{10}$$

In Equation (10), x represents the output of different layers, H denotes the nonlinear transformation function, and [] indicates concatenation—combining the features from all preceding layers of the current layer by channel. The nonlinear transformation here selects batch normalization and ReLU. Batch normalization refers to normalizing the output of each layer in the network. This is because during neural network training, the weights of each layer constantly change. Consequently, the distribution of input data alters (ICS) as it passes through the layer to the next layer, preventing the neural network from learning useful data distributions.

Batch normalization approximates the data distribution at each layer to a normal distribution. For the ReLU activation function, applying batch normalization ensures that gradients at that layer either exist or are maximized. Consequently, batch normalization stabilizes and accelerates training while preventing gradient vanishing issues.

The complete network depicted consists of multiple dense convolution blocks, with a transition layer between each pair of blocks. The transition layer consists of one 1 × 1 Conv layer with one pooling layer following that. Since the input feature maps of the previous layer pass through a Dense block leading to an increased number of output channels, the need for 1 × 1 Conv kernel arises. The main parameter of the transition layer is the reduction ratio between 0 and 1. In this paper, reduction is set to 0.5, meaning the number of channels passed to the next Dense Block is halved.

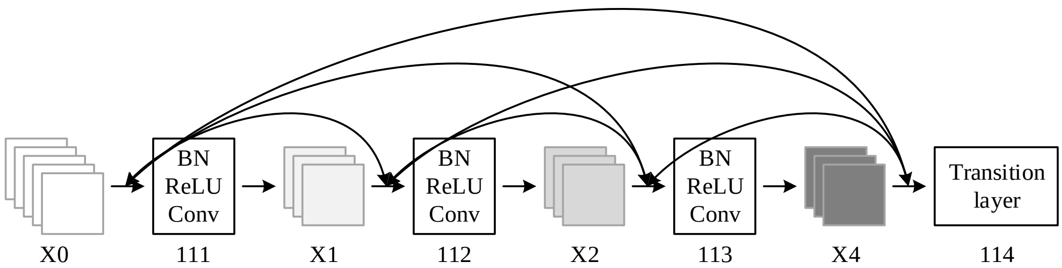


Figure 2: Schematic diagram of dense connection

(2) DenseNet-Based Policy-Value Network

This paper adopts a network architecture similar to AlphaGo_Zero to construct a DenseNet-based common network. This network produces two outputs: policy output and value output. During training, the policy-value network is trained jointly. The policy-value network used in this paper is illustrated in Figure 3.

Inputs first pass through a convolutional layer with a 3×3 kernel. After batch normalization and ReLU activation, the output yields 32×32×32 image information. This convolutional layer reduces the input channel count to a specified number for DenseNet utilization. The convolutional layer output connects to four dense convolutional blocks. Each pair of convolutional blocks is separated by a transfer layer. Within the dense convolutional blocks, varying numbers of bottleneck layers are present. Each bottleneck layer consists of a 1×1 convolution, ReLU activation function, batch normalization, 3×3 convolution, ReLU activation function, and batch normalization. The four dense convolution blocks contain 4, 8, 12, and 6 bottleneck layers respectively. The transition layers connecting each pair of convolution blocks comprise batch normalization, ReLU activation function, and a 1×1 convolution layer. The policy-value network contains three transfer layers in total. Thus, the common part of the policy-value network comprises 64 convolutional layers (the first convolutional layer contains 1 convolutional layer, the 30 bottleneck layers in DenseNet contain 60 convolutional layers, and the three transfer layers contain 3 convolutional layers, totaling 64 convolutional layers). The common network connects to two outputs: the Policy head and the Value head. The Policy head consists of two convolutional layers, producing a final image size of 32×32×2. This represents the probabilities of selecting policies at 2048 positions. The Value head is made up of one Conv layer and two fully connected layers that will give the probability of attaining 17 different values. This is all the structure of the Policy-Value Network.

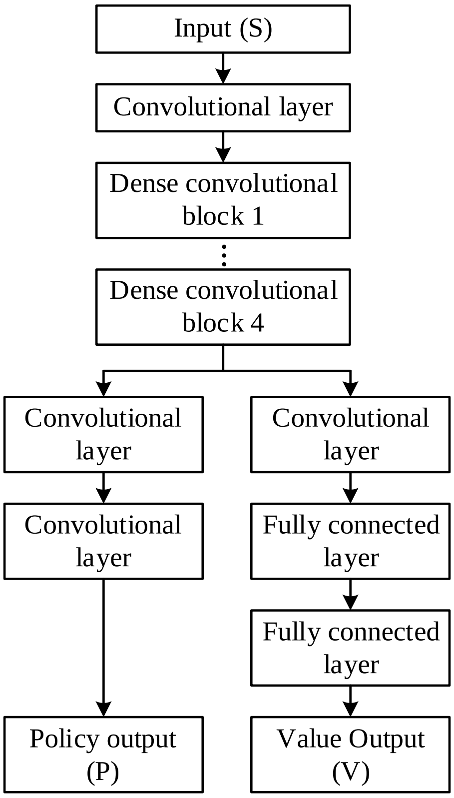


Figure 3: Schematic Diagram of the strategic value network

3.2.2 Monte Carlo Tree Search and Its Improved Algorithms

The core idea of the Monte Carlo Tree Search algorithm is to repeatedly explore the Monte Carlo tree, updating the value and visit count of nodes during each exploration, ultimately selecting the optimal node [36]. The following sections detail the main steps of the Monte Carlo Tree Search algorithm: selection, expansion, simulation, and backpropagation. Here, "selection" does not refer to choosing the final strategy to execute, but rather selecting a child node of a given node during each search iteration to update its value and visit count. In traditional Monte Carlo Tree Search, the selection function is the Upper Confidence Bound (UCT) function, defined by the following formula. The node with the maximum UCT value among the child nodes is ultimately selected:

$$UCT(v_i, v) = \frac{Q(v_i)}{N(v_i)} + c\sqrt{\frac{\log(N(v))}{N(v_i)}} \tag{11}$$

In Equation (11), v represents the parent node, v_i represents the child node, Q denotes the total value of the node (updated with each visit), and N denotes the number of visits to the node. Figure 4 shows a schematic diagram of the Monte Carlo tree.

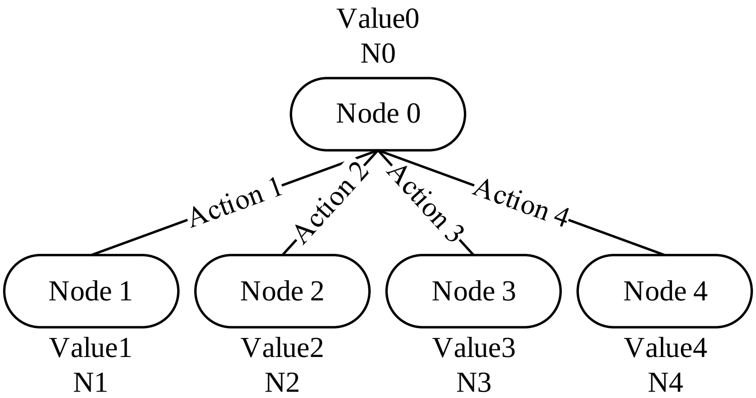


Figure 4: Schematic diagram of the Monte Carlo tree

The first term on the right side of the UCT formula represents the average value of the node (total reward / total visits = average reward per visit), reflecting the estimated win rate of child nodes. The second term on the right side represents the ratio of visits to the parent node versus child nodes, favoring child nodes with fewer visits. A classic problem in reinforcement learning is exploration versus exploitation. Exploitation refers to using currently acquired information to select the strategy that yields the maximum reward, while exploration involves trying different actions to gain more experience.

Classic Monte Carlo Tree Search (MCTS) starts from the root node. If a child node of the root is a terminal node (ending a game), the reward for that state is propagated back to the child node, and its visit count is incremented. If the node is not a terminal node, it must be determined whether the node is fully expanded. If not fully expanded, the child node is added to the Monte Carlo tree. Simulation then proceeds from this child node using the Rollout method: a strategy is randomly selected from the state space until the game ends, and the resulting reward is propagated back. If fully expanded, the child node is selected using the UCT function, and the simulation and reward propagation process is repeated. Repeat the above steps until all search steps are completed. Finally, select the policy corresponding to the node with the highest visit count as the optimal policy, i.e.:

$$a^* = \arg\max_{a \in A_i} N(a) \tag{12}$$

In equation (12), A denotes the policy space, N represents the number of visits, and a^* signifies the optimal policy.

3.3 Design and Implementation of Athlete Performance Data Evaluation

Based on the theoretical introduction to the ID3 algorithm in Section 3.1.1, incorporating this algorithm to construct decision trees enables a comprehensive evaluation of athletes' competitive performance. However, the application of the approach in practice proves that there are still several problems to be solved, such as difficulties in handling continuous variables, sensitivity to attribute missing data, and a long processing time. This paper proposes the following improvement strategies.

3.3.1 Discretization of Continuous Attributes

During the process of calculating attribute gains using the ID3 algorithm, attributes in the training and testing sets are actually categorized into discrete and continuous types. Since most attributes in the training and testing data sets contain continuous values, this section focuses on implementing the discretization of continuous attributes to optimize the ID3 algorithm for constructing decision trees.

3.3.2 Generate discretized motion data samples

Through the aforementioned implementation approach, all continuous attribute values in the sample data can ultimately undergo effective discretization processing to obtain real-time athletic competition data for athletes.

3.3.3 Calculation of Default Values for Continuous Attributes

For a given training item, traverse its initial sample array, where each data element represents a training continuous attribute with variable values. However, through discretization processing, its values can be computed based on the threshold array. The core approach of this algorithm is to integrate the rating-based threshold array proposed earlier in this paper. It calculates the probability of samples appearing within each rating interval. Using the following formula (13), it combines the interval probability with the interval threshold to comprehensively compute the final default value. This ensures the default value aligns with the selection range and specifications for continuous attribute values:

$$\text{defaultValue} = \sum_{index=0}^{3} (range_{index} \times piex_{index}) \tag{13}$$

where index denotes the subscript of the threshold array, $range_{index}$ denotes the ith threshold of the interval division, and $piex_{index}$ denotes the number of samples on the interval $[range_{index}, range_{index+1}]$ as a proportion of the total sample data. .

3.3.4 Simplified calculation of information gain

It can be found that the traditional ID3 algorithm takes a long time to calculate because its own calculation formula contains a large number of logarithmic operations, which invariably increases the amount of operations, and with the growth of attribute fields and data size, the logarithmic calls will be more frequent, and the cost of time increases significantly. Therefore, the focus of this improved algorithm lies in simplifying the complexity of the related calculation formula and improving the speed of information gain value calculation:

$$I(d_{1j}, d_{2j}, \ldots d_{nj}) = -\sum_{i=1}^{n} p(ij) \log_2 p(ij) \tag{14}$$

This improved algorithm utilizes the power series formula to achieve an approximation to the original formula with reference to the following equation (15):

$$Ln(1+x) = -\sum_{n=0}^{\infty} \frac{(-1)^n}{n+1} * x^{n+1} = x - \frac{1}{2}x^2 + \frac{1}{3}x^3 \ldots + \frac{(-1)^n}{n+1} x^{n+1} x \in [0,1) \tag{15}$$

Now let $p_i = \frac{1-k_i}{1+k_i}$, $p_i \in (0,1], k_i \in (0,1]$, and substituting the previous Eqn. (14), we can derive the following Eqn. (16), in which $k_i = \frac{1-p_i}{1+p_i}$, $p_i \in (0,1]$:

$$-\sum_{i=1}^{m} p_i \log_2 p_i = -\sum_{i=1}^{m} \frac{1-k_i}{1+k_i} \times \frac{Ln\frac{1-k_i}{1+k_i}}{Ln2} \tag{16}$$

Expanding Eq. (16) in conjunction with the power expansion (15) yields:

$$-\sum_{i=1}^{m} p_i log_2 p_i = \frac{2}{Ln2} \sum_{i=1}^{m} \frac{1-k_i}{1+k_i} \times \left(k_i + \frac{1}{3}k_i^3 + \frac{1}{5}k_i^5 + \frac{1}{7}k_i^7 + \cdots \right) \tag{17}$$

As can be seen from equation (17), the accuracy of its calculation increases gradually as the power series increases gradually. However, since the ultimate goal is to compare the sizes by the calculated values, and the values are not required to be completely accurate, the first two terms of the formula can be retained to make an approximate calculation and obtain the following formula (18):

$$-\sum_{i=1}^{m} p_i \log_2 p_i \approx \frac{1}{d} \sum_{i=1}^{n} \frac{d_i (d^3 - d_i^3)}{(d+d_i)^3} \tag{18}$$

Eventually, the original attribute entropy calculation formula, by expanding, deforming, approximating and other operations, is simplified into Eq. (18), which contains only simple formula calculations and does not contain complex logarithmic operations.

3.4 Athlete performance prediction design and implementation

3.4.1 Data pre-processing

In the process of realizing this function, the relevant BP neural network model is introduced and constructed according to the relevant technical theories in Chapter 3.1 for grade prediction. At the same time, the firefly algorithm is implemented and combined to optimize the weights in the neural network model appropriately, so as to improve the accuracy of the prediction model.

Before using the neural network model for prediction, since the data range and data unit are different for different test items, in order to improve the generality of the model, the sample data can be prioritized for normalized preprocessing operation, and the processed data can be used as the input of the model, which is shown in the following processing formula (19):

$$\hat{x}_\iota = \frac{(x_\iota - x_{\min})}{(x_{\max} - x_{\min})} \tag{19}$$

where x_i denotes the original data values, $x_{\max}$, $x_{\min}$ denote the maximum and minimum values in the set of sample data, respectively.

3.4.2 Firefly Algorithm Iterative Search for Optimal Solutions

After setting the relevant parameters, the distribution of the firefly population can be initialized by means of the fitness function. Then the iterative calculation of position update starts. In this paper, with reference to the relevant theory, the adopted position formula (20) is shown below, which realizes the behavior of the swarm gradually gathering to the firefly with the highest luminosity:

$$x_i(t+1) = x_i(t) + \beta_0 e^{-\gamma r_{ij}^2}(x_j(t) - x_i(t)) + \alpha(rand - 0.5) \tag{20}$$

After the position is updated, the optimal solution is searched by continuing to re-estimate the firefly brightness based on the fitness function as a send generation search. Eventually, after reaching the maximum number of iterations, the optimal weights of the searched neural network are obtained.

After the model is built, it is possible to exercise the test set and get the corresponding performance prediction results. Due to the previous normalization operation, the resultant data need to be transformed correspondingly in the end to get the final real prediction result value, and its transformation formula is shown in (21) below:

$$x_i = \hat{x}_l \times (x_{\max} - x_{\min}) + x_{\min} \tag{21}$$

3.5 Strategy search with Monte Carlo trees

The Monte Carlo tree search uses the introduction of a UCT selection function to select the optimal strategy, followed by a gradual expansion algorithm to determine when to expand the nodes, a strategy network to initialize the nodes instead of the expansion process of the original Monte Carlo tree search, and a value network instead of the simulation process of the original algorithm to represent the back propagation process to update the value of the nodes with the number of visits. In the online game, the overall algorithm is executed 1000 times for a certain state, and the optimal strategy is obtained as the one that has been selected the most times.

In combining the Monte Carlo tree with the strategy value network, the motion process is extended to the Monte Carlo tree along with the probabilities of the strategy network outputs, replacing the original extension step of the Monte Carlo tree search.

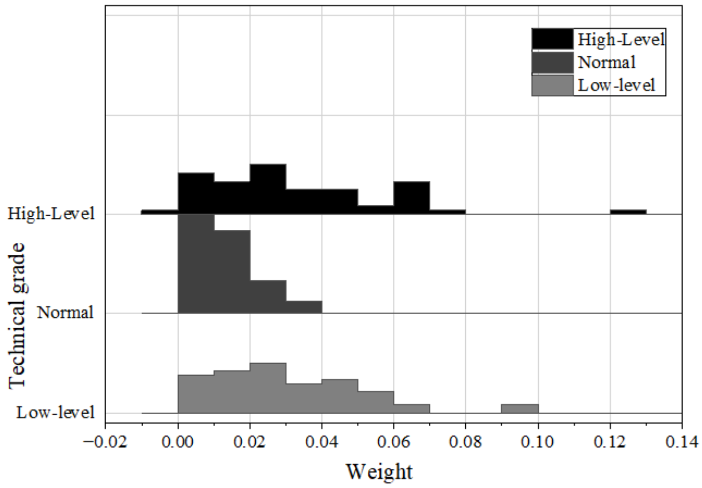
The trained strategy network is able to output the probability of choosing different movements more accurately based on the current game state. In the Monte Carlo expansion step, assuming that the current node is the parent node and the node has no children, the corresponding game state of the parent node is input into the strategy network, the output of the network is the probability of different landing points, the top 20 landing points corresponding to the action with the highest probability are selected in the output, and the nodes corresponding to the 20 actions are expanded to the parent node.

4 Real-time strategy decision support analysis for sports competitions

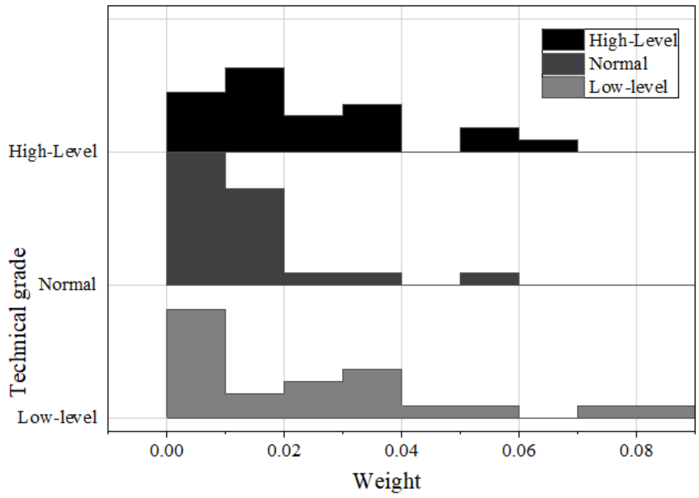
4.1 Real-time motion data analysis for sports competitions

4.1.1 Motion data analysis

This paper takes a basketball game as an example and uses the strategy support system designed in this paper to analyze the game in real time. Figure 5 shows the analysis of athletes' game technology and strategy, Figure (a) is the analysis of game technology, and Figure (b) is the analysis of game strategy. In the figure, the percentage of techniques with ordinary technology level is concentrated around 0~0.04, while the percentage of beginner and advanced techniques is concentrated around 0~0.07 and 0~0.08, respectively. The proportion of advanced strategies used by the athlete in the match strategy reached between 0 and 0.04.



(a)Technical analysis of the competition

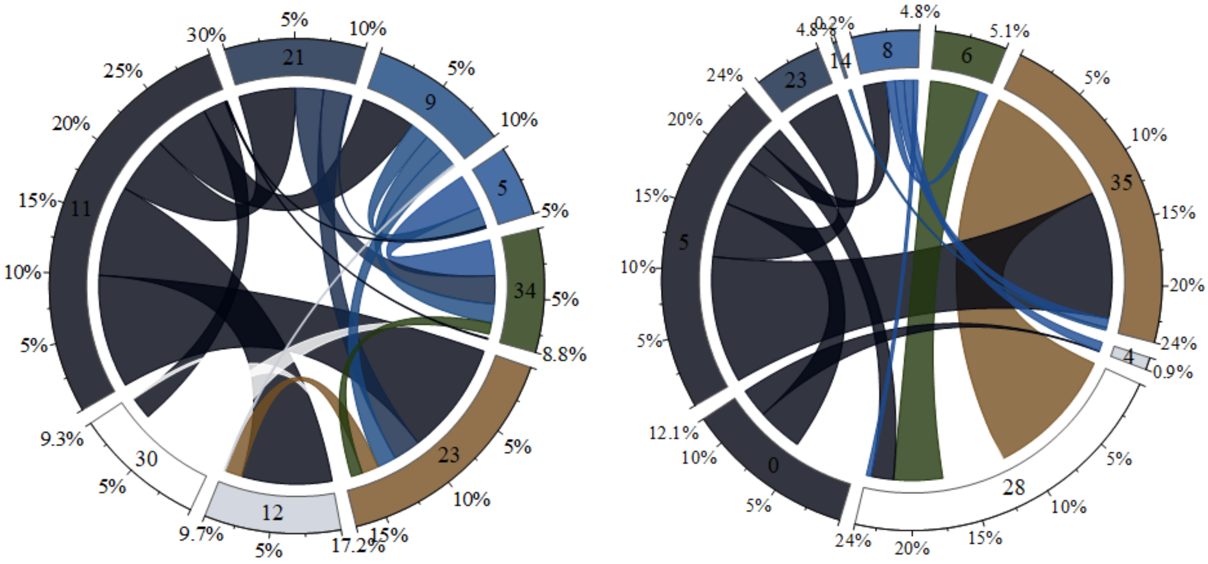


(b)Analysis of match tactics

Figure 5: Technical and tactical analysis of athletes' competition

4.1.2 Strategy Engagement

This section evaluates the strategy decision support system by first observing the general overview of the strategies used by both teams in this game, as shown in Fig. 6, with Team A in Fig. (a) and Team B in Fig. (b), to get a general idea of the players' participation in the strategies of both teams.The players involved in the blocking of Team A are concentrated in No. 11, No. 23, and No. 21 and No. 9, and the three players' participation in the blocking strategy is 30%, 17.2%, 17.2%, 17.2% and 10%, respectively, 10%, and 10%, with No. 11 acting mainly as a ball carrier, No. 21 acting mainly as a blocker, and No. 23 acting as both a ball carrier and a blocker.No. 23 plays different roles when executing blocking with different players; when executing blocking with No. 11, No. 23 acts mainly as a blocker, whereas when executing blocking with No. 21 and No. 12 No. 23 acts mainly as a ball carrier, which Team A has a large number of strings, indicating that the team's blocking strategy is very diversified, with each player blocking with a number of other players, and there are many combinations of blocking strategies that can be executed.Team B's blocking strategies mainly involve players #35, #5, and #28, with 24% of the blocking strategies and a small number of strings in the graph. The number of blocking strategies is small, and the blocking strategies are concentrated in a few fixed combinations of players, such as No. 28 and No. 35, No. 5 and No. 35, No. 5 and No. 23, and the richness of blocking strategies is not enough. Moreover, there is a lack of players within the team who can play different roles like No. 23 of Team A. No. 5 and No. 28 only play as ball carriers, and No. 35 basically only plays as a blocker. As can be seen from this view, Team A used more flexible blocking combinations and more varied strategy options in this game.



(a)A team (b)B team

Figure 6: Participation in the player strategy

4.1.3 Strategic Analysis

Firstly, we observe the number of passes between the main players of team A. As shown in Fig. 7, the number of passes between No. 11 and No. 9 as well as the number of times they dribble is very high, in which the number of times the two players dribble themselves reaches 134 and 147 indicating that the team's passing and dribbling tasks are mainly carried out by the two men. 40 and No. 11 also have the most number of passes, which reaches 150, indicating that there is a high degree of cooperation between the two men. The team's cooperation is very high. This data helped us to gain a general understanding of the players' individual styles and their involvement in the team's dribbling strategy activities, as well as identify the core player of the team's passing system, #11, to facilitate subsequent real-time strategy decisions.

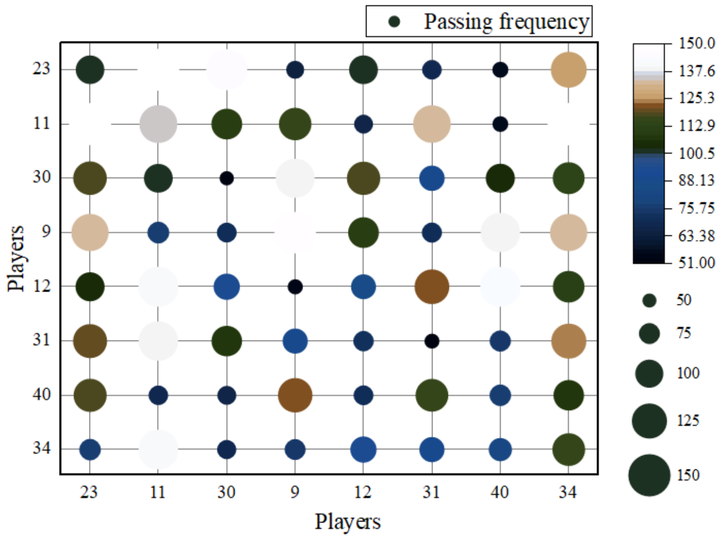


Figure 7: The number of passes between the main players

4.2 Real-time strategy selection

4.2.1 Analysis of competition results

The set of basketball matchup scenarios is defined during the execution of the UCT-based bootstrapped Monte Carlo tree algorithm as M={A,B,C,D,E,F,G,H,I}= {Initial Attack Matchups, Triangle Attacks, Overall Attacks, Aggressive Defenses, Pinch Defenses, Fastbreak Strategies, Main Attacks, Complementary Defensive Matchups, Swap Defenses}. In addition the Monte Carlo tree algorithm is set to have a minimum confidence level of 35% and a minimum support level of 30% or more. The Monte Carlo tree algorithm performs the strategy search process by defining the set of all terms as $G=\{g_1,...,g_k\}$, which is the set of terms of the set G.

In the process of system testing, five video datasets of live matches were collected from two teams of players, and the results of the matches of the two teams as well as the results of the two decision-making modes were counted, and the results of the matches are shown in Table 1.Team A has a slightly better victory rate of 80% based on the decision-making system, and the victory rate of Team B is 20%, which is a little inferior to that of Team A. Team A's match-up scenarios are more diversified, whereas Team B's strategy is more homogeneous, with unclear offensive and defensive strategies. Team A has more diversified matchups, while Team B has a more homogeneous strategy with an unclear offense and defense. This result shows to some extent the effectiveness of the decision-making in this paper, which can cope with the manual decision-making on the field and achieve a more satisfactory result.

Table 1: Competition result

Team name	Round One	Round two	Round Three	Round Four	Round Five	Beat rate
A team	Win	Win	Lose	Win	Win	80%
B team	Lose	Lose	Win	Lose	Lose	20%
Team A Strategy Support System solution	{A,B}	{B,C,D}	{C,F}	{C,D,E}	{A,C,E}	-
Team B's manual plan	{G,H}	{D,H}	{B,I}	{E,H}	{B,D}	-

4.2.2 Decision-making performance

In order to avoid the randomness of the system performance test, 20 basketball games were then launched in two days, and Team A still used the strategic decision support system designed in this paper, and the performance of Team A using the decision system in 100 basketball games was counted. Meanwhile, five coaches were invited to manually review the strategy solutions of the decision system after the games, and Figure 8 shows the performance evaluation of the decision system.

In 100 games, the decision-making system designed in this paper was able to make strategy judgments on the current basketball games according to the expected settings, and there was no unsolved phenomenon, which proved that the system was more reliable. It takes at least 3~5 min to make a strategy decision program manually, and very rarely can make a decision in an instant, while the artificial intelligence assisted system calculates the data very fast, and the average time to make a decision is about 3.13s, and the decision can be generated in a few seconds even for the most complicated game scenes, which saves a lot of time compared to manual decision making. The victory rate of the system is 74%, which means that in 100 games, team A won 74 games under the decision-making scheme of the intelligent system, but the victory rate can only explain the effectiveness of the system to a certain extent, and the victory or defeat depends on the environment of the field, the players' status, the players' strength, etc. 5 coaches were satisfied with the system as high as 95%, and the reasonableness of the system's decision-making is considered to be as high as 84% after a careful analysis. 84%, that is to say, most of the time the system gives a reasonable solution to the game, with a certain scientific basis. In contrast, the response time of traditional Apriori algorithm and sequence mining algorithm to make a decision program is as high as 5.36s and 6.14s, respectively, and the decision-making sensitivity is not ideal, the reason why the algorithm of this paper makes a program decision in a short period of time. In addition, the victory rate of sequence mining is only about 47%, after coaching on the decision-making results of the seminar learned that the algorithm's decision-making rationality is only 42%, the reliability is relatively poor, the traditional Apriori algorithm decision-making performance is also the same, there is no great improvement. It can be seen that the decision-making system designed in this paper has achieved a more satisfactory decision-making effect, and its application value and popularization in real-time decision-making of sports competitions are strong.

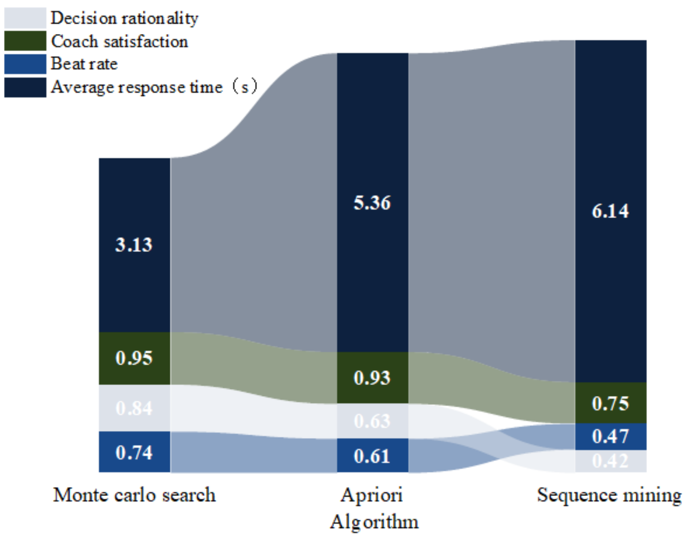


Figure 8: Decision-making system can be evaluated

4.3 Strategy prediction scores and effects

4.3.1 Predicted results

Figure 9 shows the athlete data processing, first of all, zero mean processing of the data, the data is full of 75 points, the calculated mean is 54.794 variance is 6.249, the use of the mean value of the athlete's comprehensive analysis of the data of zero homogenization. Subsequently, the firefly algorithm was used to predict the real-time performance of the athletes based on the above basic data of the athletes.

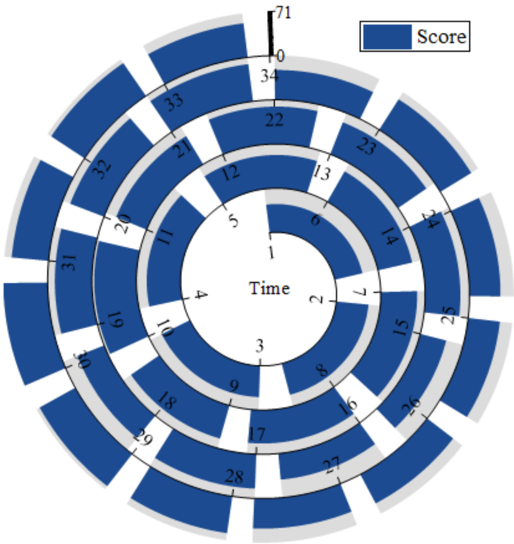


Figure 9: Athlete data processing

Figure 10 shows the control between the predicted and actual values, and the two sets of data are basically consistent. The average rating of the actual values is 0.386, and the average rating of the predicted values is -0.056, and the average error is 0.442, which puts the prediction error within the acceptable range.Next, the prediction error of the firefly algorithm is further analyzed.

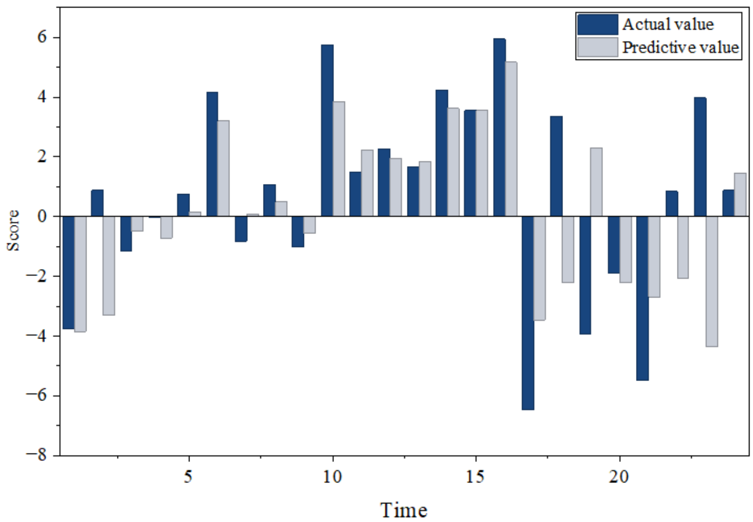


Figure 10: The comparison between the predicted value and the actual value

Figure 11 shows the prediction error, from which it can be seen that the prediction error is concentrated in the range of [-2,2], accounting for 79.167% of the whole prediction results, with a high confidence level. There are individual errors in the figure are relatively large, through the data analysis of the data samples, it is found that this is because the given sample values are obtained from the clinical monitoring, because of human negligence, so that the individual indicators of the monitoring items have been omitted, and these values can be filtered it out. Therefore, the feasibility of the athlete performance prediction model based on the firefly algorithm is high.

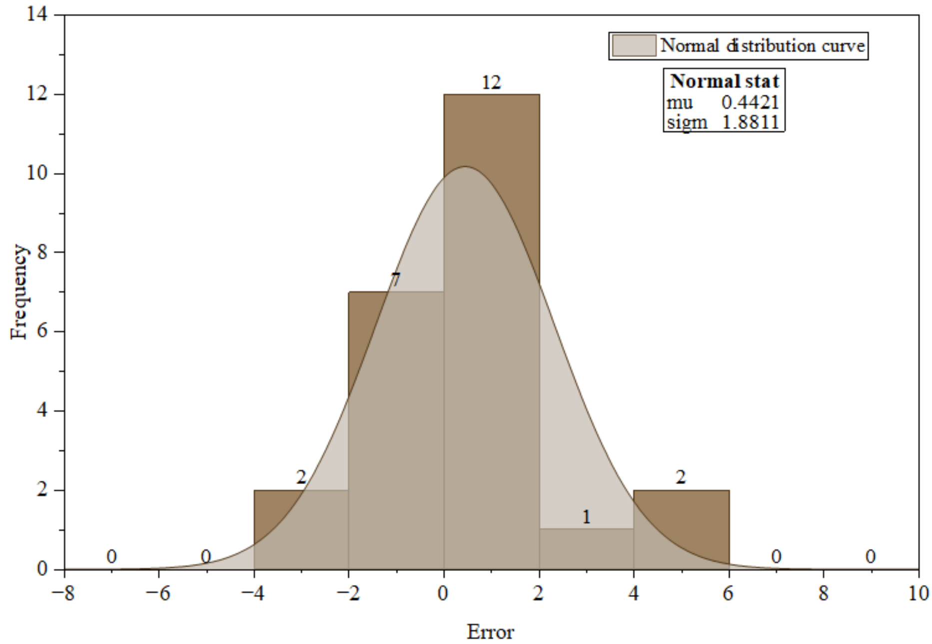


Figure 11: Prediction error

4.3.2 Predicted effects

Table 2 shows the comparison between before and after the application of the strategic decision support system, and the application results show that in the new season, the team's average number of rushes increased by 22%, the ball possession rate increased by 8.33%, and the injury rate of players decreased by 16.67%. By optimizing formations and strategies, the team achieved more victories against stronger teams.

Improvements in individual players' technical statistics are also evidence of the results of big data analysis. For example, a striker improved his shot accuracy from 18% the previous season to 30% after personalized data analysis. Data analytics in the system revealed deficiencies in his pass selection and timing in specific areas, and targeted training helped him improve his skills in these areas.

The value of big data analytics is further demonstrated by the analysis of correlations between game results and the quality of strategy execution. For instance, during one season, the team analyzed its data and found out that the team mostly scores through diagonal passes and fast counterattacking plays. Once it established that this strategy was effective, the team focused more on it in future matches, and it emerged that the team scores an additional 0.5 goals in every game compared to when using other strategies. Additionally, the winning percentage of the team increases by 31% when the fast counterattacking strategy is applied in the matches. It can be concluded that the identification and improvement of the strategy traits of a team through big data analysis could help increase the winning percentage of the matches.

The methods that can be used in validating the outcomes of the decision support system analysis include comparative analysis, which entails analyzing the differences in team performance prior to and after implementing big data analytics. Through this method, the effectiveness of the analysis can be determined. For instance, the number of goals per game rose from 1.5 to 2.2. Secondly, backtesting is another validation method, where analysts look back at historical data to test the accuracy of the analytics model's predictions of past matches, such as predicting how a team would perform against a specific opponent. Finally, simulation experiments are also commonly used to validate analytic strategies. For example, different formations are simulated against the same opponents to assess which formation is better able to control the pace of the game and create scoring opportunities. The combined application of these methods allows for a comprehensive assessment of the effectiveness of AI-based analysis of strategic decision-making systems in enhancing the performance of sports competitions.

Table 2: The strategy decision support system is compared before and after the application

Index	Pre-use of decision support system	The decision support system is used	Rate of change
League rankings	8	4	+4
Average sprint	18 secondary/match	22 secondary/match	+22%
Control rate	48%	52%	+8.33%
The player is in the rate	12%	10%	-16.67%
Fight against a strong team	30%	50%	+20%
Passing success	65%	74%	+50% Success rate
The team averaged	1.5/match	2.2/match	+46.67%
Forward shot accuracy	18%	30%	+66.67%
The success rate of the midfield pass	75%	83%	+10.67%
Number of midfield assists	3 number/season	8 number/season	+100%
The border transfer leads to the number of goals	-	Significant increase	-
Quick fight leads to the number of goals	-	Significant increase	-
Quick strike rate	24%	55%	31%

5 Conclusion

In this paper, following the development trend of decision-making in competitive sports, the preliminary conception of a decision support system for live sports competitions is proposed. The concept of entropy is introduced on the basis of ID3 algorithm so as to establish a decision tree model, and, the optimal solution is searched by firefly algorithm to realize the processing of real-time data of sports competitions. On the basis of Monte Carlo tree search algorithm, the UCT selection function is introduced to select the best strategy, and the strategy that has been selected the most times is obtained as the optimal strategy.

Taking basketball game as an example to analyze the real-time sports data of sports competitions, the players in the subject team A who use the blocking strategy more often are

No. 11, No. 23 and No. 21 and No. 9, and the percentage of them using this strategy is 30%, 17.2%, 10% and 10%, respectively. Observing the number of passes between the main players of team A from the obtained data, two players, No. 11 and No. 9, had more passes with their own dribbling, and their own dribbling amounted to 134 and 147 times, respectively. The number of passes between teammates was also the highest between No. 40 and No. 11, reaching 150 passes, indicating a high degree of cooperation between the two.

Using Monte Carlo tree algorithm to search for the optimal strategy, the average time for the system designed in this paper to make a decision is about 3.13s, the system's victory rate is 74%, and the coach's satisfaction with the system is as high as 95%, and after analyzing the system's strategy, the system's strategy reasonableness is judged to reach 84%, and most of the time, the system gives the reasonable match-up plan.

About the Author

Tao Chen was born in Luoyang, Henan Province in 1974. He graduated from Shanghai University of Sport with a master's degree and currently works as a senior lecturer in the Department of Public Basic Teaching at Luoyang Vocational College of Culture and Tourism. His main research focus is on Physical Education and Sports Training.

References

[1] He, Y. M., & Luo, W. D. (2025). Research on Sports Data Mining and Sports Training Decision Support System Based on Big Data Technology. International Journal of High Speed Electronics and Systems, 34(01), 2540133.

[2] Usmanova, N., Igamberdiyev, O., & Allamuratov, M. (2025). Sustainable Management of Sports Events through Intelligent Systems and Economic Strategies. In SHS Web of Conferences (Vol. 216, p. 02007). EDP Sciences.

[3] Narne, S., Adedoja, T., Mohan, M., & Ayyalasomayajula, T. (2024). AI-driven decision support systems in management: enhancing strategic planning and execution. International journal on recent and innovation trends in computing and communication, 12(1), 268-276.

[4] Ye, X. (2022, November). Competitive sports activities based on artificial intelligence technology. In International Conference on Artificial Intelligence and Intelligent Information Processing (AIIIP 2022) (Vol. 12456, pp. 79-84). SPIE.

[5] Ma, C., & Shou, M. (2021). Sports competition assistant system based on fuzzy big data and health exercise recognition algorithm. Mobile Information Systems, 2021(1), 7687178.

[6] Du, T., & Bi, N. (2025). Application of Artificial Intelligence Advances in Athletics Industry: A Review. Concurrency and Computation: Practice and Experience, 37(3), e8372.

[7] Atasoy, B., Efe, M., & Tutal, V. (2021). Towards the artificial intelligence management in sports. International Journal of Sport Exercise and Training Sciences-IJSETS, 7(3), 100-113.

[8] Dinca-Panaitescu, T., & Dinca-Panaitescu, S. (2023). Artificial intelligence in the sports industry. In AI and Society (pp. 113-125). Chapman and Hall/CRC.

[9] Nagesha, K. V., Yedukondalu, G., Atmakuri, P., Babu, S. T., Sreenivasgoud, P., & Gupta, M. (2023, April). Analysis on Implementation of Artificial Intelligence in the sports Activity. In 2023 Eighth International Conference on Science Technology Engineering and Mathematics (ICONSTEM) (pp. 1-7). IEEE.

[10] Yu, S. (2016). Design and practice of the schedule arrangement system of track and field sports competition based on artificial intelligence. Revista Ibérica de Sistemas e Tecnologias de Informação, (18B), 61.

[11] Bajwa, S. S. (2024). Role of Artificial Intelligence (AI) in the Promotion of Sports. Asian Journal of Research in Social Sciences and Humanities, 14(5), 1-7.

[12] Shi, R. (2025, June). Application and development trends of artificial intelligence in competitive sports. In Of Papers Presented at 2025 6th Asia Sport Science Conference (ASSC).

[13] Hsu, C. Y., Huang, B. R., Wang, M. Y., & Yun-Jie, F. (2023). Exploring competitive sports technology development: using a MCDM model. Journal of Physical Education & Sport, 23(8).

[14] Feng, J. (2023). Designing an Artificial Intelligence-based sport management system using big data. Soft Computing, 27(21), 16331-16352.

[15] Krstić, D., Vučković, T., Dakić, D., Ristić, S., & Stefanović, D. (2023, November). The application and impact of artificial intelligence on sports performance improvement: A systematic literature review. In 2023 4th international conference on communications, information, electronic and energy systems (CIEES) (pp. 1-8). IEEE.

[16] Sirawattana, C., & Poolsamral, C. (2024). The Use of Artificial Intelligence in Sports Science to Enhance Athlete Performance. Journal of Arts Management, 8(4), 700-710.

[17] KINALIOĞLU, İ. H., & KUŞ, C. (2023). Prediction of football match results by using artificial intelligence-based methods and proposal of hybrid methods. International Journal of Nonlinear Analysis and Applications, 14(1), 2939-2969.

[18] Fialho, G., Manhães, A., & Teixeira, J. P. (2019). Predicting sports results with artificial intelligence–a proposal framework for soccer games. Procedia computer science, 164, 131-136.

[19] Tan, X. (2023). Enhanced sports predictions: A comprehensive analysis of the role and performance of predictive analytics in the sports sector. Wireless Personal Communications, 132(3), 1613-1636.

[20] Kasera, M., & Johari, R. (2021). Prediction using machine learning in sports: a case study. In Data Analytics and Management: Proceedings of ICDAM (pp. 805-813). Singapore: Springer Singapore.

[21] Keshtkar Langaroudi, M., & Yamaghani, M. (2019). Sports result prediction based on

machine learning and computational intelligence approaches: A survey. Journal of Advances in Computer Engineering and Technology, 5(1), 27-36.

[22] Rojas-Valverde, D. (2025). Big Data and Artificial Intelligence in Sports Analytics. In Digitalisierung und Innovation im Sport und in der Sportwissenschaft: Handbuch Sport und Sportwissenschaft (pp. 1-14). Berlin, Heidelberg: Springer Berlin Heidelberg.

[23] Ke, Y. (2021, September). Application of artificial intelligence in large-scale international wushu (taolu) events. In 2021 2nd international conference on big data & artificial intelligence & software engineering (ICBASE) (pp. 575-578). IEEE.

[24] Xu, S., & Tang, C. (2021, December). Application of AI technology in sports competitions. In International Conference on Artificial Intelligence, Virtual Reality, and Visualization (AIVRV 2021) (Vol. 12153, pp. 158-163). SPIE.

[25] Liu, Z. (2020, February). Application of artificial intelligence technology in basketball games. In IOP Conference Series: Materials Science and Engineering (Vol. 750, No. 1, p. 012093). IOP Publishing.

[26] Huang, Y. (2021, November). The Application of Artificial Intelligence Technology in the On-site Decision System of Sports Competitions. In 2021 International Conference on Big Data, Artificial Intelligence and Risk Management (ICBAR) (pp. 106-109). IEEE.

[27] Fu, Z., & Wu, Y. (2025). The application of artificial intelligence in sports competition scoring. Journal of Computational Methods in Sciences and Engineering, 25(2), 1923-1937.

[28] Dong, H. (2024, December). Overview of the Application of Artificial Intelligence in Sports Events. In Proceedings of the 2024 International Conference on Sports Technology and Performance Analysis (pp. 324-331).

[29] Chen, X. (2021, June). Research on the application of artificial intelligence technology in the field of sports refereeing. In Journal of Physics: Conference Series (Vol. 1952, No. 4, p. 042048). IOP Publishing.

[30] Wang, D., & Liang, F. (2021, April). Application of Artificial Intelligence and Big Data in Sports Event Service——Take Guilin as an Example. In Journal of Physics: Conference Series (Vol. 1881, No. 3, p. 032056). IOP Publishing.

[31] Nagovitsyn, R. S., Valeeva, R. A., & Latypova, L. A. (2023). Artificial intelligence program for predicting wrestlers' sports performances. Sports, 11(10), 196.

[32] Chenxi, L. I., Zhihua, Y. I. N., Siyuan, Y. U., Zhen, G. U. O., & Bo, L. I. U. (2024). Working Mechanism, Application Value, and Practice Paths of Artificial Intelligence Promoting the Development of Professional Sports Events. Journal of Chengdu Sport University, 50(5), 27-37.

[33] Lin Liu & Guannan Sheng. (2022). Application of Training Load Prediction Model based on Improved BP Neural Network in Sports Training of Athletes. International Journal of Advanced Computer Science and Applications (IJACSA),13(10),

[34] Huang Ke & Wang Tingxuan. (2024). Optimized Application of the Decision Tree ID3 Algorithm Based on Big Data in Sports Performance Management. International Journal of e-Collaboration (IJeC),20(1),1-20.

[35] Ashish Kumar Singh & Anoj Kumar. (2025). Multi-objective: hybrid particle swarm optimization with firefly algorithm for feature selection with Leaky ReLU. Discover Artificial Intelligence,5(1),192-192.

[36] Helen Lai Christos Kannas Alan Kai Hassen Emma Granqvist Annie M. Westerlund Djork Arné Clevert. & Samuel Genheden. (2025). Multi-objective synthesis planning by means of Monte Carlo Tree search.Artificial Intelligence in the Life Sciences, 7, 100130-100130.