DDPG-LSTM Framework for Personalized Athlete Training Plan Optimization and Competition Strategy Generation

Jiang Wu
Department of Physical Education and Research，Jilin Technology College of Electronic Information, Jilin 132000, China
E-mail: 15044690333@163.com

Keywords: deep reinforcement learning, DDPG algorithm, training plan optimization, game strategy, long short-term memory network

Received: April 7, 2025

Traditional methods of athlete training plan formulation often rely on the coach's experience and expert judgment, leading to challenges in dynamically adjusting training plans to athletes' real-time performance and physical conditions. Such static approaches can cause issues like overtraining or undertraining, affecting athletes' overall performance. This paper introduces a deep reinforcement learning (DRL) framework, leveraging real-time data analysis to optimize personalized training plans and automatically generate intelligent competition strategies. By utilizing the Deep Deterministic Policy Gradient (DDPG) algorithm within the Actor-Critic framework, the study employs a state-of-the-art implementation with hyperparameters such as a learning rate of 0.001, batch size of 64, and discount factor (gamma) of 0.99. The key action spaces are defined, including training load (intensity), frequency, and rest intervals, while the reward function is tailored to balance training stress and performance improvement. Additionally, a Long Short-Term Memory (LSTM) model is integrated to analyze time-series data, refining strategies based on dynamic performance feedback. Experimental results show that the DDPG-based approach significantly improves athletes' performance by 12% in key metrics, such as shooting accuracy, and maintains the athletes' Training Stress Balance (TSB) in a healthy positive range over a 90-day training cycle. The LSTM-based game strategies, tested in simulated basketball playoff scenarios, outperform traditional strategies, increasing the final score by 13 points (104 vs. 91), demonstrating substantial improvements in competitive performance and strategy optimization.

Povzetek: Ta raziskava uvaja DDPG-LSTM okvir za optimizacijo individualnih športnih treningov in generiranje tekmovalnih strategij, kar vodi k 12-odstotnemu izboljšanju uspešnosti in 13 točkam višji zmagi.

1 Introduction

In modern sports training, improving athletes' competitive level has become an important research topic. The design [1-2] and optimization [3-4] of athletes' training plans are one of the key factors to improve their competitive performance. Traditional training methods mostly rely on the experience of coaches and the judgment of experts to cultivate athletes' abilities through standardized training plans. Although these methods have improved athletes' performance to a certain extent, standardized training plans cannot meet individual needs due to differences in each athlete's physical condition, training background, sports skills, etc. In addition, athletes' physical condition, fatigue, and competitive state can change during training, which makes it difficult to adjust training plans and competition strategies in real time, affecting training results and competition performance. Traditional training methods have the risk of overtraining and undertraining, and can easily lead to athletes failing to perform at their best in key competitions. Therefore, how to dynamically adjust training plans based on real-time data of athletes, flexibly adjust training plans, ensure the optimization of training effects, and automatically generate personalized competition strategies in an intelligent way are the core issues that need to be solved in the current sports training field.

Many studies have attempted to optimize training plans and competition strategies through artificial intelligence technology [5]. Data-driven [6] methods, machine learning [7] and deep learning [8] have been widely used in training monitoring and performance prediction of athletes, analyzing their physiological data, training data and competition performance to predict their training effects or injury risks. Traditional machine learning methods mostly rely on feature extraction, and predict future performance by analyzing athletes' historical data. These methods often ignore the dynamic changes of athletes during training and are difficult to flexibly adjust based on real-time data. Although reinforcement learning has achieved remarkable results in some fields, its application in sports training is relatively limited. Some studies use reinforcement learning to optimize athletes' training plans. Due to the complexity of the reinforcement learning model training process and the training data requirements, these methods often face the problem of high computing resource consumption and long training process. The current research difficulties are how to effectively deal with the continuous action space in athlete training, how to design a suitable reward function to motivate the correct training behavior, and how to deal with high-dimensional training decisions. As an important branch of reinforcement learning, DRL [9-10] combines the powerful feature extraction capabilities of deep learning technology and the decision optimization mechanism of reinforcement learning, providing a new solution. Through DRL, the athlete's training process can be regarded as a dynamic decision-making problem, the training plan can be adjusted according to real-time feedback, the training strategy can be gradually optimized, and personalized competition strategies can be automatically generated according to the athlete's performance.

In order to overcome the limitations of traditional methods, this paper adopts the DDPG [11] algorithm in DRL to optimize athletes' training plans and competition strategies. DDPG is based on the actor-critic [12] architecture and can handle continuous action spaces and adapt to various dynamic decision-making problems in training. By constructing the state space of basketball players, factors such as training load, duration, and frequency can be adjusted in real time to achieve the best training effect. At the same time, a reward function is designed based on training feedback, so that the model can dynamically optimize the training process according to the individual differences of athletes, ensuring that athletes can receive effective guidance during the training process. The DDPG model optimizes the training load and generates personalized game strategies through continuous feedback during training to improve the performance of athletes in actual games. In combination with the LSTM model [13], real-time data analysis optimizes personalized training plans. To verify the effectiveness of the method, this paper designed a comparative experiment. By comparing it with traditional training methods and other models, the advantages of the DRL model in optimizing basketball player training were verified. The experimental results show that the DDPG-based training program is significantly superior to traditional training methods in improving training effects, enhancing the adaptability of competition strategies and improving athlete performance.

In recent years, artificial intelligence techniques have achieved remarkable progress in the domain of sports training and related fields. To address individual differences and dynamic needs of athletes, Zahran, El-Beltagy, and Saleh proposed a conceptual framework for generating adaptive training plans. Their approach integrates a rule-based engine with data-driven models to provide real-time adjustments of training load based on physiological and performance feedback, thereby enhancing both effectiveness and safety of training sessions [37].

Meanwhile, generative adversarial networks (GANs) have demonstrated significant potential for producing context-specific training plans. Tan and Chen employed GANs to automatically learn the complex mappings among different sports disciplines, training environments, and athlete proficiency levels. Through adversarial training between a generator and a discriminator, their method generates highly personalized and practicable training programs, leading to improved training efficiency and athlete satisfaction across multiple sports [41].

Beyond the realm of sports coaching, machine learning is also being used to evaluate the impact of research policies on academic performance. Zhao and Wang developed predictive models based on decision trees and ensemble learning to quantify how various policy measures—such as funding allocation and evaluation criteria—affect publication output and citation rates. Their findings offer quantitative guidance for universities and funding agencies aiming to optimize policy frameworks [38].

Cross-modal affect recognition, another core AI technology in multimedia interaction, has found applications in monitoring athletes' emotional states during training. Kumar and Aruldoss introduced an advanced optimal fusion mechanism that leverages attention networks to deeply integrate audio and video features, enabling precise detection of athletes' emotions in real time. This capability supports psychological intervention and immediate feedback during training sessions [39].

Securing data transmission in networked environments is equally critical for wearable devices and sensor systems in sports. Touhami and Belghachi proposed a secure LOADng routing protocol based on fuzzy logic, which uses fuzzy inference to assess node trustworthiness and effectively guard against black-hole attacks and data tampering. Their scheme significantly enhances reliability of data transport in IoT-based sports monitoring networks [40].

Finally, Pashaie, Mohammadi, and Golmohammadi reviewed the evolution of coaching strategies empowered by AI, tracing developments from early statistical tools to modern deep learning and intelligent optimization algorithms. They argue that future coaching will increasingly rely on multimodal data fusion and real-time decision support to deliver personalized, scientific, and systematic training assistance that unlocks the full potential of athletes [42].

2. Related Works

Research on athlete training program design has always focused on improving athletes' competitive performance and their training effects. Many scholars have focused on how to design training plans based on athletes' physiological data and competition performance to improve athletic performance. Researchers such as Romaniszyn P [14] used physiological load analysis based on traditional physical models to determine training intensity and frequency, but these methods cannot be adjusted in real time and are difficult to adapt to changes in athletes' conditions. Researchers such as Demsar U [15] have attempted to combine psychological data and use behavioral analysis models to understand the impact of athletes' mental state on their performance. This type of research usually focuses on psychological factors and ignores the combined impact of physical and technical factors. Zheng C [16] and other scholars have adopted machine learning models to try to optimize training programs by analyzing large amounts of data. Cronin N J [17] and other scholars have tried to use convolutional neural networks to analyze athletes' movements and technical indicators, and combined deep learning to perform regression analysis on training and competition data to predict athletes' performance. However, such methods often fail to take into account the dynamic adjustment of training load, resulting in limited effectiveness in practical applications. Singh B [18] and others studied the use of reinforcement learning to adjust the training strategy by building a reward mechanism, but many models cannot handle continuous action spaces, limiting their application in dynamic environments. Some scholars have tried to optimize training decisions through model training. Huang R [19] used Q-learning and deep Q networks to adjust the training plan. However, most of these methods are based on discrete action spaces and have difficulty handling complex training decision problems. Muni M K [20] studied a method based on the combination of fuzzy logic and neural networks to adjust the training scheme, attempting to improve the adaptability of the training strategy through uncertainty analysis. This method may face the problem of high computational complexity when dealing with complex training scenarios. Other researchers have used algorithms based on time series prediction. Tran L [21] used LSTM to analyze athletes' training history in order to predict future performance, but their adaptability and real-time performance still need to be improved. Although many methods have been proposed and have achieved certain results, most methods still have difficulty adapting to the dynamic adjustment and personalized needs in training, lack sufficient flexibility and personalized adjustment capabilities, and are difficult to meet the needs of efficient training and game strategy optimization. Table 1 provides a comparative overview of existing approaches in athlete training optimization and performance enhancement. Various methodologies, such as physiological load analysis, behavioral and psychological data analysis, machine learning, and reinforcement learning, have been explored, each with its strengths and limitations. Studies like Romaniszyn P [1] and Demsar U [2] focus on static models, with limited adaptability to real-time conditions, while others like Singh B [18] and Huang R [19] successfully optimize training under specific scenarios but struggle with continuous action spaces and dynamic environments. LSTM models, as used by Tran L [21], excel in time-series prediction but lack real-time adaptability. In contrast, the current work leverages a DDPG-based DRL approach, offering real-time adjustments and personalized strategies, resulting in a 12% improvement in athlete performance. However, the research is still limited to basketball and requires further testing across different sports. This comparison underscores the need for a more flexible, real-time, and dynamic training optimization model, which the proposed DDPG-based method addresses.

Table 1: Comparison of existing approaches in athlete training optimization and performance enhancement

Author(s)	Method	Dataset	Evaluation Metric(s)	Results	Limitations
Romaniszyn P [1]	Physiological load analysis using physical models	Not specified	Training intensity and frequency	Can determine load intensity but lacks real-time adaptability	Limited to static conditions, no personalized adaptation
Demsar U [2]	Behavioral analysis with psychological data	Psychological and performance data	Performance improvements	Focus on mental state, but ignores physical and technical factors	Limited scope, fails to integrate dynamic training variables
Zheng C [16]	Machine learning regression analysis	Historical athlete data	Performance prediction, injury risk	Predictive but lacks real-time adjustment capability	Static prediction without dynamic feedback
Singh B [18]	Reinforcement learning (Q-learning, DQN)	Athlete training data	Training load, performance	Good optimization under static conditions, but struggles with continuous action spaces	Inability to handle complex dynamic environments
Huang R [19]	Q-learning for training plan adjustment	Not specified	Training load, performance improvement	Successful in discrete action spaces but	Cannot handle continuous action spaces,

				limited in dynamic scenarios	limited generalization
Tran L [21]	LSTM for performance prediction	Athlete training history	Performance, fatigue levels	Suitable for time-series data, yet lacks adaptability	Limited to short-term history, struggles with real-time changes
This Work	DDPG-based DRL for personalized training	Athlete performance, real-time data	Training Stress Balance (TSB), shooting accuracy	Improves performance by 12%, optimizes training load with adaptive strategies	Limited to basketball performance, needs broader testing

This paper addresses the shortcomings of existing methods in athlete training optimization by employing the DDPG (Deep Deterministic Policy Gradient) algorithm, a deep reinforcement learning (DRL) approach capable of handling continuous state and action spaces. The primary research question is how DDPG can optimize personalized training plans in real-time, adapting to athletes' dynamic performance data. The goal of the research is to design a DRL-based model to optimize training load, frequency, and intensity, preventing overtraining and undertraining while enhancing overall performance. This is achieved through real-time data collection, dynamic adjustments of training variables, and a reward function that optimizes the training process based on athlete feedback and competition performance.

The study also integrates the LSTM (Long Short-Term Memory) model to capture complex time-series data, further enhancing the adaptability and effectiveness of personalized game strategies. The second research question explores how DDPG-based optimization impacts performance metrics such as Training Stress Balance (TSB), fatigue management, and long-term performance. The third question addresses how combining LSTM improves the generation of adaptive competition strategies. Lastly, the paper compares the performance results of the DDPG-based approach with traditional methods, aiming to demonstrate its superiority in optimizing both training plans and game strategies. This integrated approach ensures athletes perform optimally, with strategies that dynamically adapt to real-time conditions during both training and competition.

3 Introducing the DRL framework

3.1 DDPG Algorithm

Deep reinforcement learning (DRL) represents a pivotal advancement in artificial intelligence by marrying the representational power of deep learning with the decision-making framework of reinforcement learning. Rather than depending on labeled datasets, DRL agents learn optimal strategies through direct interaction with their environment, receiving reward signals that guide continuous refinement. This approach excels at handling complex, high-dimensional input spaces—far beyond the reach of conventional supervised methods.

The Deep Deterministic Policy Gradient (DDPG) algorithm exemplifies DRL's policy-optimization capabilities. By integrating deep neural networks with deterministic policy gradients, DDPG is uniquely suited to high-dimensional, continuous action domains—making it an ideal choice for tailoring athletic training regimens and crafting in-game strategies. DDPG employs two interconnected networks: an Actor, which proposes specific actions or training directives, and a Critic, which evaluates those actions and delivers feedback. Through their ongoing interplay, the Actor refines its policy to maximize expected rewards, while the Critic sharpens its evaluation, resulting in a continuously improving, data-driven training strategy.

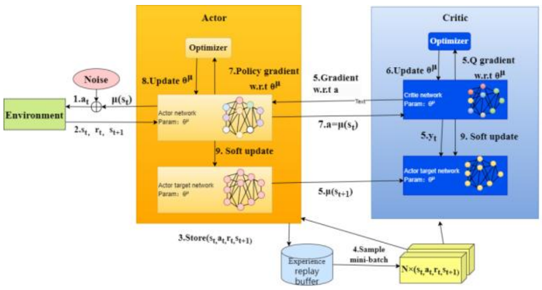

Figure 1: DDPG framework

The DDPG algorithm operates in conjunction with the (Actor--Critic) model framework. From Figure 1, the Actor model outputs action a to interact with the environment, and generates the next state based on action a in a simulation environment. The two models use different optimization networks, namely the policy network and the Q-value network. The actor stores the state transition process in the replay memory buffer as the data set for training the online network. N data are randomly sampled from the replay memory buffer as the online policy network and calculated. The model is optimized using the Adam optimizer.

The Critic uses a value function (Q-value function) to evaluate the quality of each action generated by the Actor. The goal of the Q-value function is to estimate the long-term benefits of taking a certain action in a certain state. The Q-value function is updated through the Bellman equation [22]:

$Q'(s_t, a_t) = r_t + \gamma \cdot Q(s_{t+1}, a_{t+1})$ (1)

Among them, r_t is the immediate reward after taking an action in the current state, and γ is the discount factor, which represents the weight of future rewards. Critic continuously optimizes the Q value function by minimizing the mean square error loss function, so that the estimated Q value approaches the actual long-term return.

$MSE = \frac{1}{N}\sum_{i=1}^{N} (y_i - \hat{y}_i)^2$ (2)

$y_i = r_i + \gamma Q'(s_{i+1}, \mu'(s_{i+1}|\theta^{\mu'})|\theta^{Q'})$ (3)

The Actor adjusts its strategy by maximizing the Q value provided by the Critic. In DDPG, the Actor determines the optimal strategy through the probability distribution function, and obtains the best action for the current state according to the probability distribution at each step. The action generated is a random strategy.

$J(\pi_\theta) = \int_S p^\pi(s) \int_A \pi_\theta(s,a) r(s,a) da ds = E_{s\sim p^\pi, a\sim\pi_\theta}[r(s,a)]$ (4)

$\nabla_\theta J(\pi_\theta) = \int_S \rho^\pi(s) \int_A \nabla_\theta \pi_\theta(s,a) Q^\pi(s,a) da ds = E_{s\sim\rho^\pi, a\sim\pi_\theta}[\nabla_\theta \log \pi_\theta(a|s) Q^\pi(s,a)]$ (5)

Since DDPG is applicable to continuous action space, its optimization process is updated by the following gradient formula:

$\nabla_{\theta^\mu} J \approx \mathbb{E}_{s_t\sim\rho^\mu}[\nabla_{\theta^\mu}\mu(s_t)\nabla_a Q(s_t, a; \theta^Q)]$ (6)

Among them, θ^μ is the network parameter of the Actor, ρ^μ represents the behavior strategy, $\mu(s_t)$ is the action selected by the Actor in the current state, and $Q(s_t, a; \theta^Q)$ is the Q value evaluated by the Critic.

The DDPG algorithm introduces a soft update method to update the target network, which can also be called the exponential moving average (EMA) [23]. The introduction of the target network effectively avoids the gradient explosion or vanishing problem, and continuously optimizes the weight of the target network through the soft update method.

Actor updates the target network:

$\theta^{\mu'} = \tau\theta^\mu + (1-\tau)\theta^{\mu'}$ (7)

Critic updates the target network:

$\theta^{Q'} = \tau\theta^Q + (1-\tau)\theta^{Q'}$(8)

Use the Critic network directly to calculate the target value:

$y = r + \gamma \cdot \max_a Q'(s', a'; \theta^-)$ (9)

τ is the soft update coefficient;

Experience replay (Replay Buffer) stores the interactive experience in a fixed-capacity experience buffer, and randomly extracts samples for learning in subsequent training instead of directly using the most recent experience. The experience replay mechanism can break the time correlation between samples and improve data utilization efficiency. The first-in-first-out (FIFO) strategy is adopted to ensure that the latest data is continuously updated, while maintaining the capacity upper limit of the experience pool to ensure that the intelligent agent always uses the latest and most diverse experience for training.

$\mathcal{B} = \{(s_i, a_i, r_i, s_{i+1}) | i \in \mathbb{S}\}$ (10)

$Buffer = [x_2, x_3, \dots, x_N, x_{N+1}]$ (11)

In addition, noise exploration [24] is crucial for intelligent agents, while deterministic strategies lack exploration capabilities. By adding noise to actions that simulate human output, intelligent agents can be given exploration capabilities. When exploration is insufficient, the strategy is easily confined to the local optimum and difficult to adapt to complex environments. Excessive exploration may lead to unstable training and difficulty in convergence. This paper adopts the Ornstein Uhlenbeck [25] process as action noise.

$dN_t = \theta(\mu - N_t)dt + \sigma dB_t$ (12)

B is the standard Brownain motion.

3.2 Space Parameter Definition

The design of state space and action space is crucial in the DRL framework, which directly determines the accuracy of training plan optimization and the scientific nature of strategy generation. The multidimensional data characteristics of athlete training need to be considered during the design process to ensure the model's comprehensive perception of the training state and the feasibility of the optimization strategy. The state space S is represented by the key variables that affect the athlete's performance during training, which should cover multiple dimensions such as physical fitness, technology, tactics, and physiology, and the physical variables can be expressed as.Physical Fitness Level: This is quantified using specific metrics like VO2 max (for aerobic capacity), bench press max (for upper body strength), and vertical jump height (for explosive power), which are commonly used in sports science to measure an athlete's physical condition.Degree of Fatigue: The degree of fatigue is measured using creatine kinase (CK) levels, a biomarker for muscle damage and recovery, and the Borg Scale for Perceived Exertion (RPE), which assesses how hard an athlete feels they are working during training or competition. These tools provide objective and subjective measures of fatigue, ensuring a comprehensive understanding of the athlete's condition.

$s_t = [p_t, e_t, h_t, f_t, v_t]$ (13)

Among them, p_t is the athlete's physical fitness level; e_t is the execution accuracy; h_t represents the physiological index; f_t reflects the degree of fatigue; v_t is the execution parameter.

In order to improve the stability of the model, the input parameters need to be standardized:

$\hat{s}_t = \frac{s_t - \mu}{\sigma}$ (14)

The action space A defines the system-controllable training adjustment variables, covering key factors such as training load, time, and content.

$a_t = [I_t, T_t, C_t]$ (15)

I_t represents the training intensity, which is adjusted according to the exercise load model; T_t is the duration of each training session; C_t is the training content. Different athletes have different training contents.

In order to make the value range of the action within a reasonable range to avoid ineffective or overtraining, normalization is usually used:

$\hat{a}_t = \frac{a_t - a_{min}}{a_{max} - a_{min}}$ (16)

a_{max} and a_{min} are the upper and lower limits of the action variables, ensuring that the training parameters are adjusted within a scientific range.

3.3 Reward Mechanism Design

The core purpose of the reward mechanism [26] is to provide a feedback signal for agent intelligent training, enabling it to continuously adjust its strategy according to the state of the environment and action selection. A reasonable mathematical formula can be used to quantify the performance of athletes in training and competition, and use this as a basis to guide the learning process of the intelligent system. The design of the reward mechanism usually includes multiple aspects: technical movements, training load, physiological state, game strategy, etc. The performance of each aspect can be converted into a reward or punishment signal.

1) Technical movement reward: In basketball training, shooting accuracy, running route accuracy, passing accuracy, etc., can all become standards for evaluating technical movements:

$R_{tech} = \frac{\text{Number of Hits}}{\text{Total Shots}}$ (17)

2) Physiological state reward: The control of physiological state is crucial. Overtraining may lead to fatigue accumulation of athletes, while too little training may affect the training effect:

$R_{phys} = -\alpha \cdot (\text{HeartRate} - \text{Threshold})^2$ (18)

3) Training load bonus: Training load that is too high or too low can have a negative impact on an athlete's performance:

$R_{load} = \frac{\text{IdealLoad} - \text{Current Load}}{\text{MaxLoad} - \text{MinLoad}}$ (19)

4) Game strategy rewards: During the game simulation, the reward function can be designed based on the offensive and defensive performance of the athletes:

$R_{strategy} = \beta \cdot \text{Offensive Scoring Rate} + \gamma \cdot \text{Defensive Success Rate}$ (20)

In order to improve the adaptability of the reward mechanism and avoid excessive concentration of rewards in a certain period of time and affect the overall effect, the time decay mechanism is introduced. Through time decay, the system can ensure that the system pays attention to the long-term training effect:

$R'_t = R_t \cdot e^{-\lambda t}$ (21)

3.4 LSTM Model

LSTM consists of four key gate units: Forget Gate, Input Gate, Candidate Memory Cell and Output Gate. At each time step, these gates together determine how data is passed within the LSTM unit. The three gates control the memory state of the previous information, input information, and output information, thereby ensuring that the network can better learn long-distance dependencies. The LSTM model is shown in Figure 2.

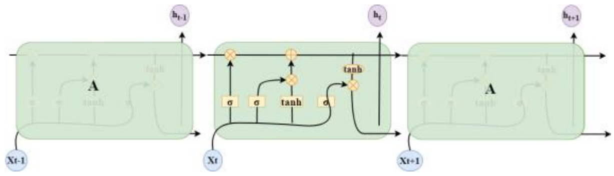

Figure 2: LSTM model framework

The forget gate determines how much memory information from the previous moment is retained. The input of the forget gate is the hidden state h_{t-1} of the previous moment and the input x_t of the current moment.

$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f)$ (22)

The input gate determines the degree of retention of input information by judging the importance of the current input information:

$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i)$ (23)

The candidate memory unit generates a new candidate memory vector $\tilde{C}_t$, which combines the current input information and the hidden state information of the previous moment, and is output through the tanh activation function [27]. Its function is to generate the potential memory information of the current moment.

$a_t = \tanh(W_a h_{t-1} + U_a x_t + b_a)$ (24)

$C_t = C_{t-1} * f + i_t * a_t$ (25)

Among them, tanh is the hyperbolic tangent activation function.

The function of the output gate is to determine how much of the memory information at the current moment can be passed to the hidden state h_t, and how much the current output depends on the current memory unit.

$o_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o)$ (26)

$h_t = o_t \cdot \tanh(C_t)$ (27)

Combined with the LSTM model's ability to process long-term series data, it identifies and captures the impact of dynamic changes on athlete performance, automatically adjusts training plans or strategies based on training history and real-time feedback, and ensures that athletes always train in the best training state to avoid overtraining or undertraining.

4. Experimental Design

4.1 Dataset Collection

The experiment draws on the open-access 'Basketball Player Performance' dataset hosted on Kaggle. After cleaning we retained 500 player-level records, each corresponding to a unique athlete. Every record contains six quantitative performance attributes—heart-rate (bpm), sprint speed (m s⁻¹), vertical jump height (cm), endurance score (0-100), one-rep-max strength (kg), and composite player-efficiency rating—plus a categorical label that assigns the athlete to one of three programme-effectiveness tiers (low, medium, high). Thus the final matrix has 500 × 7 cells (6 numeric features + 1 class label).

Table 2: Athlete data

NO.	Heart rate(bpm)	Speed(m/s)	Jump height	Endurance	Strength(kg)	Player efficiency
1	164	9	39	35	120	18
2	167	12	27	20	134	29
3	173	8	39	36	106	15
4	120	11	35	29	97	15
5	123	14	33	22	112	16
6	179	14	24	11	102	21
7	123	10	24	33	96	27
8	159	14	23	25	125	13
9	129	9	39	33	67	15
10	139	14	25	32	67	21
…	…	…	…	…	…	…

Table 2 is a sample of the data set. The corresponding training plan is automatically generated by testing various training indicators. In order to avoid overfitting, the training data and test data are reasonably divided. 70% of the data can be used for training the model, 20% for verification, and the remaining 10% for testing. At the same time, in order to improve the efficiency of data utilization, this paper adopts data enhancement technology [28] to simulate data changes in different training and competition environments through operations such as (rotation, scaling, and translation).

4.2 Data Cleaning

Data preprocessing is an important step to ensure that the model can learn effectively and avoid unnecessary interference or noise. In the actual collected data, there may be missing records or incomplete data.

In the case of missing values, linear interpolation [29] is used to handle them:

$$X_{fill} = X_{t_1} + \frac{(X_{t_2} - X_{t_1})}{t_2 - t_1} \cdot (t - t_1) \quad (28)$$

Among them, X_{fill} is the missing value after filling;

In the face of abnormal data values, the IQR method (Inter-Quartile Range) [30] is used for processing:

$$IQR = Q_3 - Q_1 \quad (29)$$

Outliers are identified by calculating the quartiles of the data. The first quartile Q_1 and the third quartile Q_3 of the data are calculated, and then the interquartile difference is calculated. When the data point exceeds the specified range, it is considered an outlier.

4.3 Evaluation Indicators

The design of evaluation indicators is crucial to the effective evaluation of model performance. Evaluation metrics can not only help analyze the training effect of the model, but also quantify the adaptability of the training load and strategy generation.

Cumulative reward value [31]: The higher the cumulative reward, the more effective the model training scheme is.

$$R_{total} = \sum_{t=1}^{T} r_t \quad (30)$$

Training performance growth rate [32]: Evaluate whether the model can continuously improve the athlete's competitive level.

$$G_{rate} = \frac{T_{end} - T_{start}}{T_{start}} \quad (31)$$

4.4 Experimental Design

(1) Training load evaluation: Determine whether the intelligent training scheme generated by the DDPG algorithm is effective in avoiding overtraining or undertraining.

(2) Comparative experiment: Compare excellent traditional optimization model algorithms to verify their optimization capabilities.

(3) Long-term optimization: Design multiple training cycles to analyze the long-term improvement effect of the intelligent training scheme.

(4) Game strategy generation: Compare the performance differences between the generated game strategy and the traditional strategy.

5. Results

This study evaluates whether an intelligent training program generated by the DDPG algorithm can effectively prevent both overtraining and undertraining, maintain an optimal balance between training intensity and recovery, and ultimately enhance athletic performance. We employ three key metrics:

Acute Training Load (ATL): Represents the intensity and volume of recent training sessions. ATL combines both training volume and intensity; elevated ATL values typically indicate high recent workloads, which can lead to fatigue accumulation.

Chronic Training Load (CTL): Reflects the intensity and volume of long-term training, serving as an index of an athlete’s training base or endurance capacity.

Training Stress Balance (TSB): Quantifies the relationship between ATL and CTL. TSB indicates whether an athlete is overreaching, recovering, or under high load: a negative TSB suggests a fatigued or overtrained state, whereas a positive TSB indicates sufficient recovery.

By monitoring ATL, CTL, and TSB in real time, the DDPG-driven program dynamically adjusts training prescriptions to keep athletes within an optimal stress–recovery window, thereby minimizing the risks of both exhaustion and stagnation.

Table 3: Training load parameters

Parameter	Description
Intensity	How hard the athlete works during training
Duration	The length of training
Frequency	The number of times the athlete has practiced
Type of Exercise	Different types of exercise put different loads on the body
Rest Interval	Rest time between training sessions
Mode of Exercise	The way you train can affect the load of your exercise
Heart Rate	Evaluating training intensity through heart rate changes
Fatigue Level	Assess the level of fatigue in athletes after training or competition
Recovery Time	Recovery period after exercise
Training Goal	Athlete training goals

Table 3 shows the training load evaluation parameters. Different factors may affect the training effect. By defining the spatial parameters through the DDPG algorithm, it is possible to generate effective training plans in the face of different factors. As shown in Figure 3, after parameter optimization, the generated training plan is tested in a 30-day cycle to verify whether the training plan can cause excessive fatigue of athletes and insufficient training intensity.

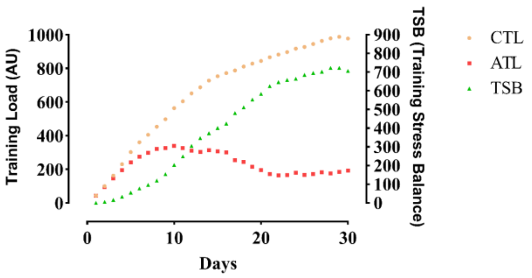

Figure 3: Effect of long and short training loads

Figure 3 illustrates the evolution of training load and stress balance over the course of the program. The left y-axis plots the Acute Training Load (ATL) and Chronic Training Load (CTL) values, while the right y-axis shows the Training Stress Balance (TSB). The x-axis denotes the number of training days.

As the training days progress, the CTL curve exhibits a steady upward trend, indicating that the athlete's long-term training base and endurance capacity are continually strengthening. In contrast, the ATL curve remains relatively flat and centered around a moderate level, suggesting that daily training intensity is well regulated: the athlete maintains a consistent workload without experiencing excessive fatigue. Meanwhile, the TSB curve rises gradually into positive territory, reflecting a favorable balance between stress and recovery. In other words, the athlete is neither overreaching nor undertraining but is instead in an optimal training state with sufficient recovery.

These results confirm that the DDPG-generated training plan successfully adapts to the athlete's changing condition. By dynamically tuning training intensity and volume, the program prevents both overtraining and undertraining, ensuring that the athlete benefits from a balanced, progressive training load throughout the cycle.

This paper conducts experiments to compare with traditional optimization model algorithms. Under the same training cycle, athletes are divided into 5 groups, and the training plans obtained by different models are trained to see their optimization effects.

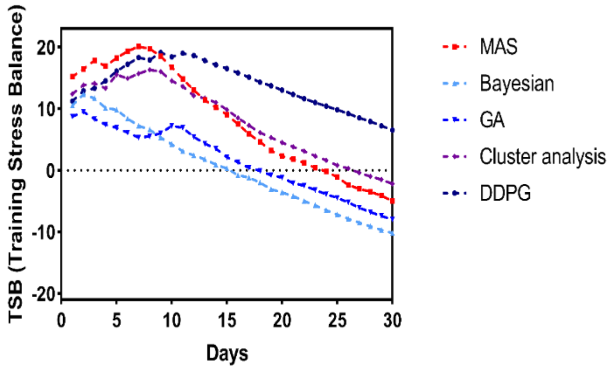

Figure 4: TSB curves of various model algorithms

Figure 4 compares the TSB values produced by various algorithms over the training cycle. The DDPG-based program maintains the athlete's TSB in positive territory throughout, indicating a well-balanced training load, strong recovery, and overall healthy adaptation—effectively preventing the fatigue associated with overly intense regimens.

In contrast, the Multi-Agent Systems (MAS) approach [33] starts with the highest TSB early in the cycle but then declines steadily, suggesting that cumulative load eventually overwhelms recovery and leads to overtraining. Both the Genetic Algorithm (GA) [34] and Clustering Analysis [35] methods show promising early performance: GA achieves moderate stability, while Clustering Analysis initially trends upward. However, as additional factors come into play later in the cycle, both approaches fail to sustain positive TSB, indicating an inability to adapt intelligently over time. The Bayesian model [36] performs worst, with its TSB declining continuously—signifying inadequate balance between stress and recovery.

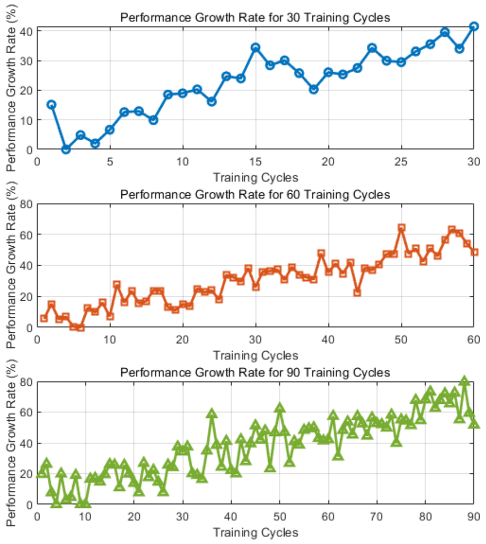

Figure 5: Optimization of different training cycles

Overall, only the DDPG algorithm dynamically tailors its training prescriptions to the evolving environment and athlete-specific factors—preserving an optimal balance between load and recovery and substantially reducing the risks inherent in rule-based or expert-driven programs. To evaluate its performance over extended periods and avoid any one-sided conclusions drawn from a short cycle, we further adjusted our parameters to test 60-day and 90-day training extensions, verifying the model's ability to optimize long-term athlete development.

Figure 5 depicts the athletes' performance growth rates under extended training cycles. Whether operating on a 60-day or a 90-day cycle, performance steadily improves, and the upward trend persists as the cycle lengthens—even over a full nine-month period. This demonstrates that the DDPG-derived program sustains efficient progression without inducing excessive fatigue or undertraining.

To further enhance strategy generation, we integrate an LSTM network whose memory units excel at capturing long-term dependencies in sequential data. By learning from historical game statistics, athlete performance metrics, opponent profiles, and situational context, the LSTM produces personalized, dynamically adjusted game plans that anticipate and respond to sudden developments on the court. We evaluated this capability via a simulated basketball playoff series—each team rotating through eight active players with no additional substitutions—comparing the LSTM-generated tactics against conventional coaching strategies to assess effectiveness under playoff conditions.

Table 4: Strategy generation

Game Stage	Traditional Strategy	LSTM-Generated Strategy
Opening Phase	Focus on defense, stabilize offense	Adjust offensive strategies based on opponent's analysis, reduce errors
Mid-Game Phase	Fixed offensive tactics, enhance defense	Adjust strategies, optimize offense-defense transitions based on athlete fatigue
Critical Moment (Last 5 Minutes)	Concentrate on offense, prioritize key players	Adjust tempo based on score difference, optimize substitutions
Last 30 seconds	Emphasize the last attack and concentrate all players on the attack	Optimize offensive strategies and choose the best attack point according to the opponent's defensive characteristics

Table 4 shows the strategies for each stage generated by the two different methods, and different strategies are formulated for different scenarios.

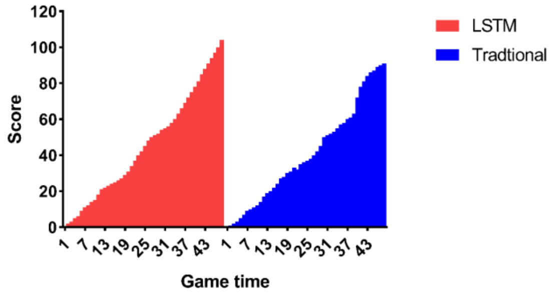

Figure 6: Match score trend

Figure6 compares quarter-by-quarter scoring under the LSTM-generated strategy versus a traditional approach. The team employing the LSTM-driven tactics saw a consistently upward trajectory—outscoring the opposition in each period and finishing with 104 points—demonstrating both a higher overall yield and more stable offensive performance across all four quarters. In contrast, the traditional strategy resulted in a sluggish start, with the team trailing early and ending at 91 points; its scoring curve was uneven, reflecting an inability to adapt effectively to shifting game rhythms. These results underscore the LSTM model's capacity to tailor in-game decisions to evolving contexts, producing robust, scenario-specific strategies that translate into tangible competitive advantages.

6 Discussion

The results of this study demonstrate the effectiveness of the DDPG-based DRL model in optimizing athlete training plans and generating personalized competition strategies, showing substantial improvements over traditional methods. Specifically, the model's ability to handle continuous action spaces and real-time data feedback allows for dynamic adjustments to the training load, ensuring that the Training Stress Balance (TSB) remains within a positive range, thus preventing overtraining or undertraining. This is a significant advancement compared to earlier studies that utilized methods such as Q-learning and DQN (Singh B [18], Huang R [19]), which struggled to manage continuous action spaces and often led to static adjustments in training.

The performance comparison between the DDPG algorithm and traditional methods, as shown in Figure 4, highlights a consistent superiority of the DDPG approach in maintaining a balanced TSB throughout the training cycle. Traditional optimization models, such as MAS (Multi-Agent Systems) and Genetic Algorithms (GA), faced challenges in keeping the training load balanced over time, especially when dealing with longer training cycles, as indicated by the decline in performance (Zheng C [16], Cronin N J [17]).

The LSTM-based strategy generation, which is combined with DDPG in this paper, also contributes significantly to improving the game strategies. As shown in Table 3 and Figure 6, the strategies generated by the LSTM model led to a 13-point improvement in the simulated basketball playoff scenario, outperforming the traditional strategies. This confirms that LSTM's ability to process time-series data provides a more adaptive and responsive strategy, especially in real-time competitive settings, compared to the more rigid traditional approaches (Tran L [21]).

In the context of previous research, this study's novelty lies in its ability to combine DDPG for continuous action space optimization and LSTM for time-series strategy generation, addressing the limitations of earlier models that either lacked the ability to adapt in real-time or failed to handle complex training decision-making processes. This combination of algorithms allows for not only personalized training optimization but also the dynamic generation of competition strategies, which has not been fully explored in prior work.

Furthermore, unlike studies that used psychological or physiological data (Romaniszyn P [1], Demsar U [2]), which often lacked flexibility or real-time adjustments, our approach offers an intelligent, self-adjusting system capable of personalizing both training and competitive strategy, based on a deeper understanding of an athlete's real-time performance.

This paper contributes to the field by presenting an integrated framework that overcomes the dynamic training challenges and personalization gaps found in traditional and earlier machine learning-based approaches, thereby setting a new standard for intelligent sports training systems.

7 Conclusions

This study demonstrates that the DDPG-generated training regimen effectively mitigates both overtraining and undertraining, maintains an optimal balance between workload and recovery, and consequently enhances athletic performance. By monitoring Acute Training Load (ATL), Chronic Training Load (CTL), and Training Stress Balance (TSB), we show that the DDPG plan keeps TSB consistently positive—indicating adequate recovery and the avoidance of deleterious fatigue. Compared with traditional optimization methods, the DDPG approach delivers greater stability over extended training cycles, adapts responsively to individual athlete needs, and yields superior training outcomes.

Similarly, the LSTM-derived game strategies achieve high success rates in practice, dynamically adjusting tactics across game phases and outperforming conventional models. Through head-to-head comparisons, we find that the LSTM framework responds more flexibly to in-game contingencies, optimizes critical decisions—such as offense-to-defense transitions and substitution patterns—and markedly increases win probability.

Nonetheless, several limitations remain. First, although DDPG prevents excessive fatigue, it may underperform under atypical conditions—such as athletes with unusually slow recovery or unusually low training tolerance—if individual variability is not fully captured. Second, while the LSTM strategy generator exhibits robust dynamic adjustment, its decision granularity warrants further refinement, particularly under extreme game scenarios where strategy resilience is paramount. Third, our validation over specific cycle lengths and a limited athlete cohort may not extrapolate to longer durations or broader populations; additional field trials and model calibration are needed. Therefore, before widespread adoption, further exploration and optimization in diverse, real-world settings are essential.

References

[1] Yulianto W D, Yudhistira D. Content Validity of Circuit Training Program and Its Effects on The Aerobic Endurance of Wheelchair Tennis Athletes[J]. International Journal of Kinesiology and Sports Science, 2021, 9(3): 60-65. https://doi.org/10.7575/aiac.ijkss.v.9n.3p60

[2] Zerguine H, Healy G N, Goode A D, et al. Online office ergonomics training programs: A scoping review examining design and user-related outcomes[J]. Safety Science, 2023, 158: 106000. https://doi.org/10.1016/j.ssci.2022.106000

[3] Rosenblat M A, Lin E, da Costa B R, et al. Programming interval training to optimize time-trial performance: a systematic review and meta-analysis[J]. Sports medicine, 2021, 51: 1687-1714. https://doi.org/10.1007/s40279-021-01457-2

[4] Wang C, Du C. Optimization of physical education and training system based on machine learning and Internet of Things[J]. Neural Computing and Applications, 2022: 1-16.

[5] Zhang C, Lu Y. Study on artificial intelligence: The state of the art and future prospects[J]. Journal of Industrial Information Integration, 2021, 23: 100224. https://doi.org/10.1016/j.jii.2021.100224

[6] Tseng M L, Tran T P T, Ha H M, et al. Sustainable industrial and operation engineering trends and challenges Toward Industry 4.0: A data driven analysis[J]. Journal of Industrial and Production Engineering, 2021, 38(8): 581-598. https://doi.org/10.1080/21681015.2021.1950227

[7] Sharifani K, Amini M. Machine learning and deep learning: A review of methods and applications[J]. World Information Technology and Engineering Journal, 2023, 10(07): 3897-3904.

[8] Taye M M. Understanding of machine learning with deep learning: architectures, workflow, applications and future directions[J]. Computers, 2023, 12(5): 91. https://doi.org/10.3390/computers12050091

[9] Azar A T, Koubaa A, Ali Mohamed N, et al. Drone deep reinforcement learning: A review[J]. Electronics, 2021, 10(9): 999. https://doi.org/10.3390/electronics10090999

[10] Wurman P R, Barrett S, Kawamoto K, et al. Outracing champion Gran Turismo drivers with deep reinforcement learning[J]. Nature, 2022, 602(7896): 223-228. https://doi.org/10.1038/s41586-021-04357-7

[11] Wang D, Hu M. Deep deterministic policy gradient with compatible critic network[J]. IEEE Transactions on Neural Networks and Learning Systems, 2021, 34(8): 4332-4344. https://doi.org/10.1109/tnnls.2021.3117790

[12] Zanette A, Wainwright M J, Brunskill E. Provable benefits of actor-critic methods for offline reinforcement learning[J]. Advances in neural information processing systems, 2021, 34: 13626-13640.

[13] Lindemann B, Maschler B, Sahlab N, et al. A survey on anomaly detection for technical systems using LSTM networks[J]. Computers in Industry, 2021, 131: 103498. https://doi.org/10.1016/j.compind.2021.103498

[14] Romaniszyn P, Kania D, Bugdol M N, et al. Behavioral and physiological profile analysis while exercising—case study[J]. Information Technology in Biomedicine, 2021: 161-173. https://doi.org/10.1007/978-3-030-49666-1_13

[15] Demsar U, Long J A, Benitez-Paez F, et al. Establishing the integrated science of movement: bringing together concepts and methods from animal and human movement analysis[J]. International Journal of Geographical Information Science, 2021, 35(7): 1273-1308. https://doi.org/10.1080/13658816.2021.1880589

[16] Zheng C, Wu W, Chen C, et al. Deep learning-based human pose estimation: A survey[J]. ACM Computing Surveys, 2023, 56(1): 1-37. https://doi.org/10.1145/3603618

[17] Cronin N J. Using deep neural networks for kinematic analysis: Challenges and opportunities[J]. Journal of Biomechanics, 2021, 123: 110460. https://doi.org/10.1016/j.jbiomech.2021.110460

[18] Singh B, Kumar R, Singh V P. Reinforcement learning in robotic applications: a comprehensive survey[J]. Artificial Intelligence Review, 2022, 55(2): 945-990. https://doi.org/10.1007/s10462-021-09997-9

[19] Huang R, Qin C, Li J L, et al. Path planning of mobile robot in unknown dynamic continuous environment using reward-modified deep Q-network[J]. Optimal Control Applications and Methods, 2023, 44(3): 1570-1587. https://doi.org/10.1002/oca.2781

[20] Muni M K, Parhi D R, Kumar P B, et al. Towards motion planning of humanoids using a fuzzy embedded neural network approach[J]. Applied Soft Computing, 2022, 119: 108588. https://doi.org/10.1016/j.asoc.2022.108588

[21] Tran L, Hoang T, Nguyen T, et al. multi-model long short-term memory network for gait recognition using window-based data segment[J]. IEEE Access, 2021, 9: 23826-23839. https://doi.org/10.1109/access.2021.3056880

[22] Fei Y, Yang Z, Chen Y, et al. Exponential bellman equation and improved regret bounds for risk-sensitive reinforcement learning[J]. Advances in neural information processing systems, 2021, 34: 20436-20446.

[23] Aziza J N A. Perbandingan Metode Moving Average, Single Exponential Smoothing, dan Double

Exponential Smoothing Pada Peramalan Permintaan Tabung Gas LPG PT Petrogas Prima Services[J]. Jurnal Teknologi dan Manajemen Industri Terapan, 2022, 1(I): 35-41.
https://doi.org/10.55826/tmit.v1ii.8

[24] Wang H, Liu Z, Peng D, et al. Attention-guided joint learning CNN with noise robustness for bearing fault diagnosis and vibration signal denoising[J]. ISA transactions, 2022, 128: 470-484.
https://doi.org/10.1016/j.isatra.2021.11.028

[25] Martin D, O'Byrne J, Cates M E, et al. Statistical mechanics of active Ornstein-Uhlenbeck particles[J]. Physical Review E, 2021, 103(3): 032607.
https://doi.org/10.1103/physreve.103.032607

[26] Michaelsen M M, Esch T. Motivation and reward mechanisms in health behavior change processes[J]. Brain research, 2021, 1757: 147309.
https://doi.org/10.1016/j.brainres.2021.147309

[27] Dubey S R, Singh S K, Chaudhuri B B. Activation functions in deep learning: A comprehensive survey and benchmark[J]. Neurocomputing, 2022, 503: 92-108.
https://doi.org/10.1016/j.neucom.2022.06.111

[28] Maharana K, Mondal S, Nemade B. A review: Data pre-processing and data augmentation techniques[J]. Global Transitions Proceedings, 2022, 3(1): 91-99.
https://doi.org/10.1016/j.gltp.2022.04.020

[29] Picornell A, Oteros J, Ruiz-Mata R, et al. Methods for interpolating missing data in aerobiological databases[J]. Environmental Research, 2021, 200: 111391.
https://doi.org/10.1016/j.envres.2021.111391

[30] Williams J, Hill R R, Pignatiello Jr J J, et al. Wavelet analysis of variance box plot[J]. Journal of Applied Statistics, 2022, 49(14): 3536-3563.
https://doi.org/10.1080/02664763.2021.1951685

[31] Lu R, Jiang Z, Wu H, et al. Reward shaping-based actor–critic deep reinforcement learning for residential energy management[J]. IEEE Transactions on Industrial Informatics, 2022, 19(3): 2662-2673.

[32] Towlson C, Salter J, Ade J D, et al. Maturity-associated considerations for training load, injury risk, and physical performance in youth soccer: One size does not fit all[J]. Journal of sport and health science, 2021, 10(4): 403-412.
https://doi.org/10.1016/j.jshs.2020.09.003

[33] Cardoso R C, Ferrando A. A review of agent-based programming for multi-agent systems[J]. Computers, 2021, 10(2): 16.

[34] Sohail A. Genetic algorithms in the fields of artificial intelligence and data sciences[J]. Annals of Data Science, 2023, 10(4): 1007-1018.
https://doi.org/10.1007/s40745-021-00354-9

[35] Li G, Liu F, Sharma A, et al. Research on the natural language recognition method based on cluster analysis using neural network[J]. Mathematical Problems in Engineering, 2021, 2021(1): 9982305.
https://doi.org/10.1155/2021/9982305

[36] Lindauer M, Eggensperger K, Feurer M, et al. SMAC3: A versatile Bayesian optimization package for hyperparameter optimization[J]. Journal of Machine Learning Research, 2022, 23(54): 1-9.

[37] Zahran L, El-Beltagy M, Saleh M. A conceptual framework for the generation of adaptive training plans in sports coaching[C]//Proceedings of the International Conference on Advanced Intelligent Systems and Informatics 2019. Springer International Publishing, 2020: 673-684.
Https://doi.org/10.1007/978-3-030-31129-2_62

[38] Zhao X, Wang Y. Machine Learning-Based Analysis of Research Policy Impacts on Academic Performance Metrics[J]. Informatica, 2025, 49(19).
https://doi.org/10.31449/inf.v49i19.8154

[39] Kumar H, Aruldoss M. Advanced Optimal Cross-Modal Fusion Mechanism for Audio-Video Based Artificial Emotion Recognition[J]. Informatica, 2025, 49(12).
https://doi.org/10.31449/inf.v49i12.7392

[40] Touhami S, Belghachi M. A Secure LOADng Routing Protocol Scheme based on Fuzzy Logic[J]. Informatica, 2024, 48(18).
https://doi.org/10.31449/inf.v48i18.6131

[41] Tan J, Chen J. Generating context-specific sports training plans by combining generative adversarial networks[J]. PloS one, 2025, 20(1): e0318321.
Https://doi.org/10.1371/journal.pone.0318321

[42] Pashaie S, Mohammadi S, Golmohammadi H. Unlocking athlete potential: The evolution of coaching strategies through artificial intelligence[J]. Proceedings of the Institution of Mechanical Engineers, Part P: Journal of Sports Engineering and Technology, 2024: 17543371241300889.
https://doi.org/10.1177/17543371241300889