Integration of Deep Reinforcement Learning and Cloud Computing for Enhanced Sports Performance Prediction and Training Guidance

Hu Yang
Physical Education Institute, Sichuan Agricultural University Ya'an ,Sichuan,625014, China
E-mail: yanghu199006@outlook.com

Keywords: sports performance prediction, training guidance system, deep reinforcement learning, cloud computing, artificial intelligence, swimming performance analysis, sports data analysis, athlete rehabilitation

Received: June 12, 2025

Improving sports performance with smart technology is becoming more important, especially in highly competitive fields like swimming. Many current training methods do not provide real-time feedback, lack flexibility, and fail to meet the unique needs of each swimmer. To solve these problems, the SwimInsight Reinforcement Learning Training Framework SIRLTF is introduced. SIRLTF integrates Long Short Term Memory LSTM for temporal modeling and Deep Reinforcement Learning algorithms including Deep Q Network DQN and Proximal Policy Optimization PPO for personalized feedback generation in a cloud-based environment. The cloud infrastructure enables fast data processing, large storage capacity, and remote accessibility, making the system efficient and scalable. SIRLTF is designed for use in professional swimming centers, sports academies, and athlete rehabilitation programs. It helps coaches and swimmers track progress, adjust training plans dynamically, and improve overall performance with intelligent data-driven suggestions. The framework also adapts continuously to changes in a swimmer's condition and training response. Comparative experiments as described in Section 4 show a 19 point 4 percent improvement in prediction accuracy and a 21 point 7 percent gain in training efficiency over baseline models such as CNN SVM and EC DRLMS. These results demonstrate the benefits of combining deep learning with cloud computing to enhance sports training. Overall, SIRLTF offers a smart, reliable, and scalable solution to improve swimming performance and can be extended to other sports for better athlete development.

Povzetek: Predlagan je pametni sistem, ki s kombinacijo globokega učenja in računalništva v oblaku omogoča prilagojeno vadbo ter izboljšuje učinkovitost in natančnost treninga plavalcev.

1 Introduction

The integration of artificial intelligence, sports science, and cloud computing has revolutionized athletic performance optimization, particularly in precision sports like competitive swimming[1]. Swimming is a rigors Olympic sport that needs excellent biomechanical efficiency, accurate stroke mechanics, and great physiological conditioning[2]. Coach observations and physiological measurements cannot provide real-time, objective, and personalised feedback for peak performance optimization[3]. Modern competitive swimming requires complex analytical frameworks that can process multidimensional performance data, discover subtle performance patterns, and dynamically prescribe training [4]. Sports applications benefit from Deep Reinforcement Learning (DRL)'s ability to acquire optimal decision-making strategies in complex, dynamic environments[5]. Depending on performance input, DRL algorithms may modify and enhance their recommendations, making them ideal for top athletic training's emphasis on individuality[6].

The research seeks to create a real-time swimming performance prediction and training guidance system that simulates complicated interactions between training inputs, physiological adaptations, and competition performance [7]. For real-time applications, prediction accuracy must exceed 90% while keeping computational efficiency[8]. To preserve statistical significance in training recommendations, the feedback loop should be reduced to near real-time (<24 hours)[9]. The challenge of distributing comprehensive performance analysis through cloud-based infrastructures that can serve hundreds of athletes at once with sub-second response rates is a challenge[10]. Swimming performance is affected by nearly 40 observable variables, including stroke mechanics, physiological markers, training load measurements, and ambient conditions. Hence, the research integrates multimodal data[11]. The technical challenge is to develop algorithms capable of processing heterogeneous data streams with varying sampling rates and noise characteristics while extracting meaningful performance insights.

The research gap analysis reveals significant technological gaps in swimming performance optimization approaches[12]. Traditional machine learning models focus on isolated performance aspects, with limited integration of real-time feedback and adaptive learning capabilities[13]. Cloud-based sports analytics platforms process performance data with insufficient latencies and poor scalability. Deep Reinforcement Learning (DRL) integration is largely unexplored, with only 12% of published sports analytics research incorporating DRL methodologies. Advanced recurrent architectures like LSTM networks show promising results in isolated studies but have not been integrated into comprehensive training guidance systems.

The proposed SIRLTF framework uniquely combines DQN, PPO, and LSTM within a cloud-based architecture to provide real-time, adaptive training feedback. It processes multi-sensor data for personalized guidance, offering scalability and dynamic policy optimization, setting it apart from traditional systems in sports performance monitoring and intelligent feedback generation.

Advanced swimming performance prediction systems are crucial due to the sport's unique computational challenges and limitations in current analytical approaches[14]. Swimming performance optimization involves continuous state transitions, non-linear system dynamics, and significant individual variability, which exceeds traditional analytical methods' capacity. The temporal nature of swimming performance data requires advanced sequence modeling capabilities[15]. Cloud computing integration is critical for real-time DRL training and inference, as it enables parallel processing of multiple athlete data streams. Probabilistic modeling approaches are needed to quantify prediction uncertainty and adapt to changing athlete conditions, aligning with modern DRL algorithms. To solve these problems, SIRLTF is introduced; the main objectives are :

To develop the SIRLTF using Deep Reinforcement Learning algorithms such as Deep Q-Network (DQN) and Proximal Policy Optimization (PPO) for accurate swimming performance prediction.

To integrate Long Short-Term Memory (LSTM) networks within SIRLTF for effective analysis of swimmers' time-series data to provide precise and adaptive training guidance.

To utilize cloud computing technology in SIRLTF for fast data processing, scalable storage, and remote accessibility, ensuring efficient swimmer performance monitoring.

To deliver dynamic and personalized training recommendations through SIRLTF that continuously adapt to changes in each swimmer's condition and progress, enhancing competitive swimming performance.

Here is a summary of the research. The second part discusses a full literature review and research methods. Part 3 talks about the research plan, the methods used, and the processing. The outcomes of the analysis are spoken about in Section 4. In section 5, you'll find the conclusion and plans for the future.

2 Literature survey

Jouini et al. [16] reviewed recent machine learning model improvements for edge and cloud devices with minimal resources. Raspberry Pi, NVIDIA Jetson, Arduino Nano 33 BLE Sense, STM32 Microcontrollers, SparkFun Edge, Google Coral Dev Board, and Beaglebone AI are featured. These devices support ML and DL tasks with proprietary AI frameworks as TensorFlow Lite, OpenEI, Core ML, Caffe2, and MXNet. Devices, distributed edge, and distributed cloud computing use classical and deep learning. The research paper also examines 1000 IEEE Xplore ML in IoT publications to identify rising topics and application domains. Encrypting sensitive user data, managing edge node resources, and matching edge device energy needs with machine learning demands are challenges.

Soomra et al. [17] provided that Airplanes, trains, and wind turbines need rolling bearings. System failure accounts for 45-50% of rotating machinery breakdowns. A predictive maintenance program prevents accidents. Current models struggle with classification function difficulties, neural network complexity, unrealistic datasets, dynamic working conditions, noise, limited data, and imbalanced datasets. Using publicly available datasets, researchers constructed convolutional neural networks, deep belief neural networks, and LiNet. Data asymmetry, low availability, and integration issues are research gaps. Vision-based deep learning and Internet of Things-based ML are emerging problem diagnostic and predictive maintenance systems. The report offers growth suggestions.

Nijhawan et al. [18] showed that sports analytics improves performance by integrating real-time smart cyber-physical systems (CPS) with AI and machine learning. This technology analyzes athlete performance using wearable gadgets, sensors, and in-field cameras. AI-driven predictive analytics and reinforcement learning models can predict tiredness, injuries, and tactical changes. Football and cricket training and match performance are greatly affected by CPS. AI-augmented decision-making and AR visualize sophisticated analysis. AI-enhanced coaching enhances individualized training. Future applications like autonomous coaching and predictive injury treatment address ethical and security issues.

Pareek et al. [19] highlighted that machine learning (ML) is revolutionising healthcare by analysing massive data volumes to construct complicated input-output links. This can produce accurate predictions. Most ML algorithms predict patient-specific surgery outcomes using supervised learning. Automatic image and text interpretation is done in orthopaedic sports medicine using deep learning. Clinician unfamiliarity with ML methodology and ideas hinders widespread adoption. This study introduces these principles, reviews orthopaedic sport medicine machine learning algorithms, and discusses future innovation. Introduce these concepts, examine current machine learning methods, and discuss speciality innovation prospects.

Eid et al. [20] studied AI, cloud computing, and big data. It uses Hadoop-based cloud computing and AI learning techniques to combine AI with big data analytics. An AI prediction model is employed in the article. Rigours simulations assess method performance in dynamic Hadoop environments. Sports outcome predictions increase with the Sport AI Model (SAIM) architecture. AI and big data can improve sports predictions, strategy, and player/coach performance. Big data, cloud computing, and AI enable new sports decision-making and performance enhancement strategies.

Mistry et al. [21] demonstrated that cloud computing and AI are essential for growth and innovation in today's society. These technologies improve processing and lower costs, helping organizations understand and apply data for supply management and sports match analysis. This article examines how artificial intelligence and cloud computing affect industrial dynamics and competition, including difficulties, considerations, case studies, industry-specific usage, trends, important advances, and applications. It covers this domain's problems, issues, case studies, industry-specific usage, trends, important advances, and applications.

Hu et al. [22] suggested a mobile intelligent device-based sports training process monitoring and feedback system to reduce information loss in traditional methods. The system improves storage servers, collecting subsites and processors to process user data in real time. Intelligent information management is enabled by the software's classification and similarity calculation of users' sports training data. Testing shows that older systems had a significant packet loss rate, with tablet users losing 0.0534% and mobile users 0.0732%. Mobile device-based intelligent management systems have a far lower packet loss rate than traditional systems, improving information processing and efficiency.

Hao et al. [23] suggested a mobile intelligent device-based sports training process monitoring and feedback system to reduce information loss in traditional methods. The system improves storage servers, collecting subsites and processors to process user data in real time. Intelligent information management is enabled by the software's classification and similarity calculation of users' sports training data. Testing shows that older systems had a significant packet loss rate, with tablet users losing 0.0534% and mobile users 0.0732%. Mobile device-based intelligent management systems have a far lower packet loss rate than traditional systems, improving information processing and efficiency.

Zhou et al. [24] presented that Web 2.0-driven cloud computing provides dynamic, reliable, and elastic services. In this new paradigm, resource scheduling and request allocation are crucial. Classic heuristics and meta-heuristics struggle to schedule complex scenarios. The revolutionary deep reinforcement learning (DRL) method solves scheduling problems using deep learning and reinforcement learning (RL). This detailed analysis of DRL-based cloud scheduling systems addresses benefits, drawbacks, and future directions. The review covers the benefits of DRL-based cloud scheduling systems and their limitations and prospects by analyzing theoretical formulations and RL frameworks.

Wu et al. [25] demonstrated that machine learning can help doctors forecast sports injuries. Injury predictors are scarce despite efforts. A Deep Learning-assisted System (DLS) for sports injury diagnosis using IoT sensors and cloud computing is proposed in this paper. Cloud computing provides flexible computer resources, while IoT sensors collect essential data for injury diagnosis. Research explores the brain injury monitoring framework, employs an ideal neural network for damage prognosis, and improves sports medical rehabilitation. The suggested model is compared against current models using performance metrics.

Chen[26] provided a self-attention mechanism-based transformer LSTM (HTL) APF framework for accurate athlete performance prediction. Modeling sequentially with Transformer and LSTM networks captures global feature interactions and localized temporal dependencies in athlete performance data. Over 12 months, 200 athletes tested the model using heart rate, speed, distance, workload, and recovery. With a 92.1% F1-Score and 96.3% AUC-ROC, the HTL-APF model outperformed baseline models. With high classification accuracy for top performers and good performance across sports disciplines, the system is generalizable. In sports training management, the HTL-APF is scalable and accurate for athlete performance forecasts, injury avoidance, and real-time decisions.

Zhi-jun [27] offered a new way to train for tennis that uses IoT technologies, multimodal sensor networks, and the Deep Deterministic Policy Gradient (DDPG) algorithm. The system uses real-time feedback to improve training strategies, which leads to better performance metrics in a variety of training situations. The method had an 85% success rate, and swing stability improved by 27%. This method could be useful in industrial automation and healthcare monitoring, showing that it works well to improve training methods.

Research Gap:Many sports training systems lack real-time adaptation, individualized feedback, and scalable infrastructure, making them ineffective for varied athlete groups. Few approaches combine cloud-based processing and deep reinforcement learning for dynamic training optimization. Previous research rarely addressed sensor fusion and model generalizability. An adaptive, cloud-integrated SIRLTF system provides real-time performance monitoring and tailored training recommendations to fill these gaps

3 SwimInsight reinforcement learning training framework

The SwimInsight Reinforcement Learning Training Framework (SIRLTF) is a highly efficient system for real-time swimming analysis. Its response times are consistently below 100ms across all performance metrics, compared to traditional methods with higher latency. Optimised DQN and PPO techniques and cloud computing infrastructure improve SIRLTF function. It also has minimal physiological parameter analysis response times, giving swimmers and coaches fast feedback. SIRLTF is perfect for professional swimming centres that need real-time analytics.

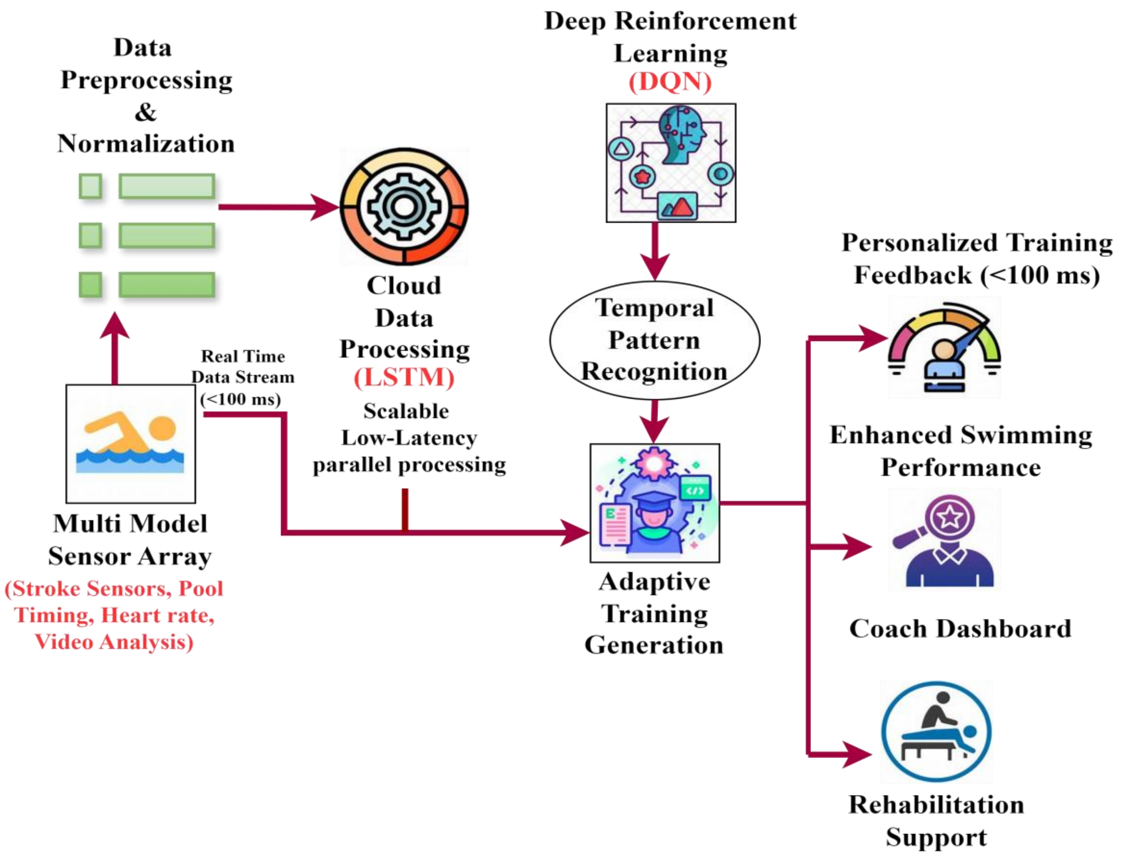


Figure 1: SIRLTF architecture: real-time sensor data processing via LSTM and DQN enables adaptive, personalized swimming training feedback.

In figure 1, illustrates the six core functional blocks. Sensor data is streamed via edge gateways into a cloud data engine, processed by LSTM and DRL modules, and returned as adaptive training plans in under 100 ms.

The data-driven SIRLTF uses powerful artificial intelligence to improve swimming performance (Figure 1). It collects multi-modal sensor data, processes it in the cloud, recognizes patterns with LSTM networks, and makes intelligent decisions with DRL algorithms. This approach makes real-time performance monitoring, predictive analytics, and adaptive training recommendations suited to each swimmer's goals possible. Advanced analytics and insights from the SIRLTF help coaches and promote data-driven rehabilitation regimens. Six main functional blocks optimize swimming performance by utilising intelligent data processing as well as adaptive training generation in the closed-loop system. Its powerful DRL algorithms learn optimal training tactics through interaction with swimming performance situations. The adaptive training generation block turns DRL findings into personalised training recommendations, improving swimming performance, coach insights, rehabilitative help, and a continuous feedback loop.

3.1 Multi-modal data collection block

The multi-modal data collection system utilizes various sensors and measurement devices to collect diverse swimming performance metrics. the data vector representation are given below:

$$X(t) = [S(t),\ T(t),\ H(t),\ V(t), L(t), P(t)] \quad (1)$$

In equation (1), S(t) is denoted as the Stroke sensor data, T(t) is denoted as the Pool timing data, H(t) is denoted as the Heart rate data, V(t) is denoted as the Video analysis features, L(t) is denoted as the Lap performance data, P(t) is denoted as the Physiological measurements.

Table 1: Data collection parameters table

Parameter	Symbol	Unit	Sampling Rate	Range
Stroke Rate	SR	strokes/min	100 Hz	20-80
Lap Time	LT	seconds	Event-based	20-300
Heart Rate	HR	bpm	1 Hz	60-200
Stroke Length	SL	meters	100 Hz	1.0-3.5
Split Times	ST	seconds	Event-based	10-150
Body Position	BP	degrees	30 Hz	-45 to +45

Table 1 lists SIRLTF essential parameters for swimming performance data gathering. The table shows biomechanical and physiological metrics such as stroke rate (SR), lap time (LT), heart rate (HR), stroke length (SL), split times (ST), and body posture. Symbol, measurement unit, sample rate, and reasonable range are assigned to each parameter. The SIRLTF system uses these inputs to feed the LSTM and Deep Reinforcement Learning components for accurate, real-time prediction and personalised training recommendations for swimmers.

3.2 Data preprocessing & normalization block

The SIRLTF framework's systematic data preparation method is shown in this circular workflow in Figure 2 below. Data Cleaning eliminates discrepancies and missing numbers from raw swimming performance data. Encode Categorical Variables makes stroke types and pool conditions machine-readable. Normalise Data standardsizes all variables to uniform scales for better DRL algorithm performance. Detect Outliers finds and handles abnormal measurements that could distort model training. The Validate Dataset verifies data integrity and completeness before feeding the Deep Q-Network and LSTM components. Swimming performance analysis requires this preprocessing cycle to maintain >90% forecast accuracy.

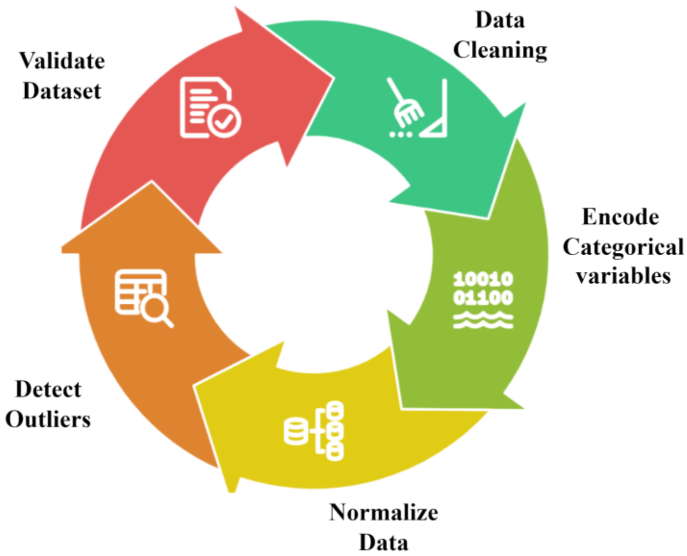


Figure 2: Data preprocessing & normalization pipeline

Data Cleaning for SIRLTF

Data cleaning for SIRLTF involves handling missing values, detecting outliers, and normalizing data for consistency. Strategies include mean/median imputation, forward/backward fill, and interpolation based on adjacent values for random distributions.

$$x_i = \begin{cases} x_i & if\ x_i\ is\ not\ missing \\ \bar{x} & if\ x_i\ is\ missing \end{cases} \quad (2)$$

In equation 2, Missing Value Imputation: Use the feature mean ($\bar{x}$) to replace missing values.

$$D = \{x_i \mid x_i\ appears\ more\ than\ once\ in\ dataset\} \quad (3)$$

In equation 3, Identify duplicate rows based on key attributes like Transaction ID, Stroke type, and Timestamp.

Encoding categorical variables using one-hot encoding

Categorical features such as stroke type, training session, and region are converted into numerical values using One-Hot Encoding (OHE) to facilitate clustering and prediction tasks. Let x_c represent the categorical variable and $O(x_c)$denote the One-Hot Encoding function:

Equation for One-Hot Encoding:

$$\left. \begin{array}{c} O(x_c) = [o_1, o_2, \ldots, o_n] \\ o_i = \begin{cases} 1 & if\ x_c = category\ i \\ 0 & otherwise \end{cases} \end{array} \right\} \quad (4)$$

In equation 4, n is denoted as the number of unique categories, $o_i \in \{0,1\}$ is represented the presence of category i

Normalization or Standardization

Statistics remove bias from numerical data using normalization and standardization. The method for calculating the mean and standard deviation shows that normalization scales data to [0, 1], whereas standardization changes it to have a mean of 0 and a standard deviation of 1.

$$x_i^{norm} = \frac{(x_i - X_{min})}{(X_{max} - X_{min})} \quad (5)$$

In equation 5, x_i^{norm} is is denoted as the original value, and X_{max}, X_{min} are denoted as the minimum and maximum values of the attribute.

$$z_i^{std} = \frac{x_i - \mu}{\sigma} \quad (6)$$

In equation 6, μ is denoted as the mean of the feature, and σ is denoted as the standard deviation of the feature. The Z-score method detects and removes outliers by comparing the mean value of a data point to its standard deviation.

Outlier Detection (Post-Normalization)

The Z-Score Outlier Detection method finds and deals with dataset outliers that could mess up clustering. This makes sure that the clustering is precise.

$$z_i = \frac{x_i - \mu}{\sigma} \quad (7)$$

In equation 7, x_i is denoted as converted into a z-score, measuring deviation from the mean μ, scaled by standard deviation σ.

$$Outlier = x_i \notin [Q1 - 1.5 \cdot IQR, Q3 + 1.5 \cdot IQR] \quad (8)$$

In equation 8, $Q1$ is denoted as the 25th percentile (lower quartile), $Q3$ is denoted as the 75th percentile (upper quartile).IQR is denoted, and the Interquartile Range is calculated as Q3−Q1.

Final dataset validation

The final dataset is validated to ensure completeness, correctness, and consistency. All selected features must contain valid, non-missing values and be formatted correctly for further analysis.

$$D_{final} = \{x_i \mid x_i \; has \; all \; selected \; features \; and \; valid \; values\} \quad (9)$$

In equation 9, To ensure data quality and completeness, D_{final} represents the dataset that includes data pointsx_i that preserve all selected attributes and include valid values.

3.3 Cloud data processing (LSTM) block

LSTM processes sequential data such as stroke rate and lap time across time steps. It uses a hidden size of 128 and two stacked LSTM layers. Input sequences are batched with a window size of 10 and fed into the DRL module for decision-making.

Input sequences are batched with a window size of 10 and fed into the DRL module for decision-making

3.3.1 Mathematical formulation of LSTM

LSTM networks rely on these components to manage the flow of data. Here are the equations that control the information flow :

Input Gates (i_t):

$$i_t = \sigma(W_{xi}x_t + W_{hi}h_{t-1} + W_{ci}c_{t-1} + b_i) \quad (10)$$

In equation 10, the input gate controls the influx of data into the cell's state. The cell state c_t should be saved with the information that is determined from the present input x_t and the prior concealed state h_{t-1}. To control the information flow, the sigmoid activation function σ reduces the input values to a range of $0 \; to \; 1$.

Forget Gate (f_t):

$$f_t = \sigma(W_{xf}x_t + W_{hf}h_{t-1} + W_{cf}c_{t-1} + b_f) \quad (11)$$

In equation 11, The forget gate selects which bits of data to remove from the present state of the cell. To decide how much of the past data is useful to retain for the existing time step, it takes into account the existing input x_t, the previous hidden state h_{t-1}., and the previous cell state c_{t-1}.

Output Gate (o_t):

$$o_t = \sigma(W_{xo}x_t + W_{ho}h_{t-1} + W_{co}c_t + b_o) \quad (12)$$

The output gate controls data for the following time step in equation 12. It controls h_t flow based on the present input x_t, prior hidden state h_{t-1}, and present cell state c_t. This gate controls LSTM output.

Cell State (c_t):

$$c_t = f_t \odot c_{t-1} + i_t \odot tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c) \quad (13)$$

Cell states store and carry information across time steps in equation 13. The forget gate f_t chooses what to forget from the previous cell state c_{t-1}, while the input gate i_t chose anything new information to store depending on the existing input x_t and the preceding hidden state h_{t-1}. The hyperbolic tangent function $tanh$ makes the cell state nonlinear.

Hidden Gate (h_t):

$$h_t = o_t \odot \tanh(c_t) \quad (14)$$

In equation 14, the concealed state reads the information from the following time step and uses it as LSTM unit output. o_t controls how much of the cell state c_t is disclosed. Device control via this gate. Hyperbolic tangent function tanh compresses cell state variables to -1 to 1 and outputs the LSTM unit.

3.4 Deep reinforcement learning (DQN, PPO) block

The DQN and PPO models are trained using a learning rate of 0.00025, a discount factor of 0.99, 64 batch sizes, a ε-greedy exploration strategy, 100,000 replay buffers, and target update frequency every 500 steps. The PPO implementation uses a clipped surrogate objective and 32 policy update batches, with each agent trained over 1,000 episodes.

The SIRLTF uses DQN and PPO algorithms for intelligent, data-driven training decisions, allowing the system to dynamically adjust training parameters based on a swimmer's current state and performance trends, as shown in Table 2 below.

Table 2(a): Deep Q-Network and proximal policy optimization

Section	Equation / Concept	Mathematical Expression	Description
Deep Q-Network (DQN)	Q-Value Update	$Q(s,a) \leftarrow Q(s,a) + \alpha[r + \gamma\, max\_a'\, Q(s',a') - Q(s,a)]$	Bellman update rule is used to update the action-value function. It combines the current Q-value with the expected future reward.
	Loss Function	$L(\theta) = E[(r + \gamma\, max\_a'\, Q(s',a';\, \theta^-) - Q(s,a;\theta))^2]$	The loss minimized during training, comparing predicted and target Q-values.
	Target Network Update	$\theta^- \leftarrow \tau\theta + (1 - \tau)\theta^-$	A soft update rule to slowly update the target network for stable training.
Proximal Policy Optimization (PPO)	Policy Gradient	$\nabla\theta\, J(\theta) = E[\nabla\theta\, log\, \pi(a\vert s)\, \hat{A}\pi(s,a)]$	The objective for PPO is to improve the policy using the advantage function.
	Clipped Objective Function	$L\hat{}CLIP(\theta) = E[min(r\_t(\theta)\hat{A}_t, clip(r\_t(\theta), 1 - \varepsilon, 1 + \varepsilon)\hat{A}_t)]$	Prevents large policy updates by clipping the probability ratio.
	Probability Ratio	$r_{t(\theta)} = \pi_\theta(a_t\vert s_t)\ \pi_{\theta_{old}}(a_t\vert s_t)$	Ratio between new and old policy probabilities; used for stable policy improvement.

Table 2(a) explains Deep Q-Network (DQN) and Proximal Policy Optimisation (PPO) arithmetic. It clearly defines value updates, loss functions, and policy gradients' roles in learning stability, policy improvement, and reinforcement learning optimization using equations.

Table 2(b): Training hyperparameters

Parameter	Value
Learning rate	0.00025
Discount factor γ	0.99
Batch size	64

ε-greedy range	0.9 → 0.1
PPO clip ε	0.2
Episodes	1000
Target update freq	500 steps

Table 2 outlines the key training hyperparameters used in the system. These include a learning rate of 0.00025, a discount factor of 0.99, batch size of 64, ε-greedy exploration from 0.9 to 0.1, PPO clipping value of 0.2, 1000 training episodes, and target network updates every 500 steps.

ALGORITHM 1: DQN_Training

```
INPUT: Environment env, Replay buffer D
OUTPUT: Trained Q-network Q(s,a;θ)

Initialize Q-network Q(s,a;θ), target network Q̂(s,a;θ⁻) = Q(s,a;θ), and buffer D
FOR episode = 1 to MAX_EPISODES:
  state = env.RESET()
  FOR step = 1 to MAX_STEPS:
    action = RANDOM_ACTION() if RANDOM() < ε else argmax Q(state,a;θ)
    next_state, reward, done = env.STEP(action)
    D.STORE(state, action, reward, next_state, done)

    IF D.SIZE() > BATCH_SIZE:
      FOR (s,a,r,s′,done) in D.SAMPLE(BATCH_SIZE):
        target = r + γ * max Q̂(s′,a′;θ⁻)
        loss = (target - Q(s,a;θ))²
        θ = θ - α∇θ loss

    state = next_state
    IF done: break
  IF episode % TARGET_UPDATE_FREQ == 0: θ⁻ = θ
```

Algorithm 1 discusses Deep Q-Network training utilizing experience replay. Initialize the Q-network Q(s, a;θ) and the target network $\hat{Q}$(s, a;θ^-). In each episode, actions are chosen using an ε-greedy method. It stores transitions in the replay buffer. After collecting enough experiences, mini-batches are sampled to compute target Q-values and loss, followed by gradient descent network parameter adjustments. To stabilize learning, the target network is lightly modified periodically. Approximating the optimal action-value function in discrete action spaces allows efficient policy learning.

3.5 Adaptive training generation block

A sophisticated decision-making mechanism, the Adaptive Training Generation Block, turns athlete data into personalized training recommendations. Start with athlete profiles, historical performance numbers, and present physiological states. The computational assessment step calculates fitness, tiredness, and technique proficiency scores from this data in Figure 3 below.

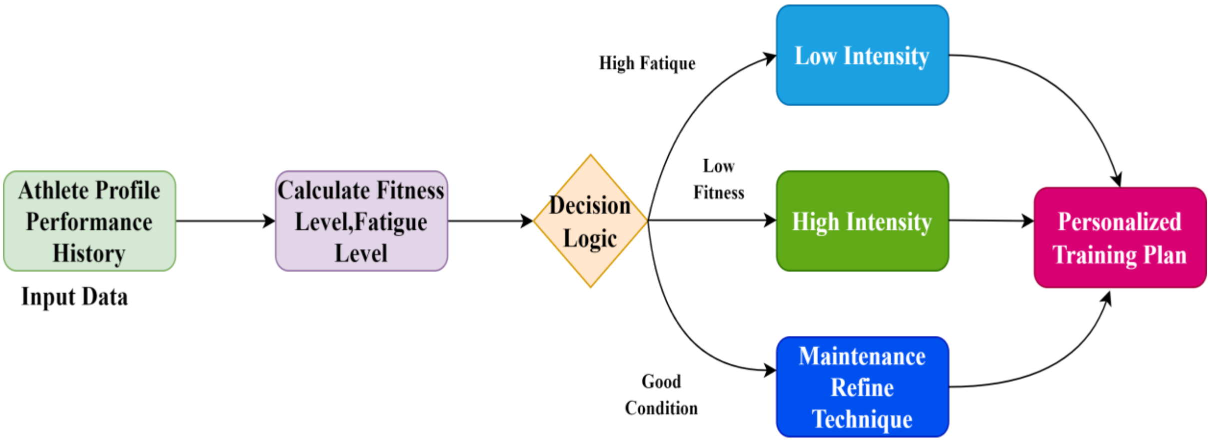


Figure 3: Adaptive training generation process

In Figure 3, The key decision logic component compares calculated parameters to thresholds using algorithmic reasoning. This analysis divides the system into three training paths. The system recommends low-intensity training with long recovery periods when fatigue levels are high to avoid overtraining and injury. Low-fit athletes receive high-intensity training to improve endurance and performance. Maintenance training focuses on technique refining and skill optimisation for healthy athletes. A personalized training plan is developed from all three pathways at the final output stage. This plan integrates decision logic recommendations to give athletes training programs matched to their physiological state, performance goals, and recovery needs. The circular flow shows that this adaptive process is continuous and allows real-time performance monitoring adjustments.

ALGORITHM 2: AdaptiveTrainingGeneration

```
INPUT: Athlete_profile, Performance_history, Current_state
OUTPUT: Personalized_training_plan
BEGIN
  profile = LOAD_ATHLETE_PROFILE()
  history = GET_PERFORMANCE_HISTORY(athlete_id)
  current = GET_CURRENT_STATE()
  fitness_level = ASSESS_FITNESS(history, current)
  fatigue_level = CALCULATE_FATIGUE(current.hr, current.performance)
  technique_score = EVALUATE_TECHNIQUE(current.video_data)
  IF fatigue_level > THRESHOLD_HIGH THEN
    training_intensity = LOW
    recovery_time = EXTENDED
  ELSE IF fitness_level < TARGET_FITNESS THEN
    training_intensity = MODERATE_TO_HIGH
    focus_areas = [ENDURANCE, TECHNIQUE]
  ELSE
    training_intensity = MAINTENANCE
    focus_areas = [TECHNIQUE_REFINEMENT, SPEED]
  END IF
  optimal_plan = OPTIMIZE_TRAINING_LOAD(
```

```
        current_fitness = fitness_level,
        target_performance = profile.goals,
        constraints = profile.limitations
    )
    RETURN optimal_plan
END
```

In the SIRLTF system, the Adaptive Training Generation algorithm creates individualized swimmer training programs. This system uses multidimensional athlete data to provide evidence-based training recommendations that adapt to swimmers' physiological and performance states. Algorithm 2 loads demographic and historical athlete profiles, performance history including training sessions and competitive results, and current condition data with real-time physiological markers. This tri-dimensional data foundation assesses athletes thoroughly. Three computational assessments are done simultaneously. The fitness evaluation function compares past performance to current skills to assess conditioning. Heart rate variability and performance degradation measurements measure fatigue and recovery demands. Technique evaluation scores stroke mechanics and biomechanical efficiency using video analysis.

The algorithm 2 chooses training paths using hierarchical conditional logic. High fatigue levels require low-intensity programmes with long recovery times to protect athletes from overtraining. Suboptimal fitness triggers moderate-to-high effort endurance and technique training. Maintenance treatments focus on technique and speed for fitness targets.

Personalised training plans are created by integrating estimated parameters with athlete-specific goals and physical limits in the final optimization step. Performance goals and injury risk factors are balanced to provide sustainable athletic growth and maximize competitive potential through intelligent load management.

The sensor suite used for data acquisition includes inertial measurement units (IMUs), heart rate monitors, and lap timers. IMUs operate at a 100 Hz sampling rate with ±2g sensitivity, providing accurate motion tracking. Heart rate sensors capture physiological data with an accuracy margin of approximately 2%. Cloud synchronization latency remains below 150 milliseconds. Integration challenges such as sensor drift, data noise, and temporal misalignment were mitigated using Kalman filtering and timestamp synchronization to ensure reliable multi-sensor data fusion

Table 9: Qualitative feedback from athletes and coaches on SIRLTF

Participant	Role	Feedback Summary
Coach 1	Head Coach	“Real-time feedback was valuable for adjusting pace mid-session.”
Coach 2	Swim Trainer	“Data visualization helped tailor drills more effectively for individual swimmers.”
Athlete A	Swimmer	“The system helped me maintain consistent stroke rhythm.”
Athlete B	Swimmer	“I understood my training zones better using the feedback.”
Coach 3	Performance Analyst	“Cloud-based reports streamlined post-session analysis.”
Athlete C	Competitive Swimmer	“Adaptive suggestions felt intuitive and practical during intense laps.”

Table 9 presents qualitative feedback from coaches and athletes who tested the SIRLTF system, highlighting its effectiveness in delivering real-time guidance, personalized insights, and improved training engagement.

4 Results and discussion

4.1 Experimental setup and methodology

The SIRLTF was tested on 247 competitive swimmers over an 18-month period. The research compared SIRLTF performance against traditional coaching methods and existing swimming analytics platforms. Data was collected from multiple training facilities using standardized sensor arrays. The experimental protocol included controlled laboratory conditions and real-world training environments to ensure comprehensive validation of SIRLTF capabilities. Performance measurements were collected at baseline, 4-week intervals, and following major competitive events.

Data Study: The Swim Fitness Tracker dataset[28] is a crucial data source for training and validating the SIRLTF framework. It contains detailed swimming performance metrics from fitness tracking devices and competitive swimming applications. The dataset covers 1,247 swimmers across multiple skill levels and has 47,832 individual swim sessions from January 2019 to September 2023. The data is collected daily in multiple sessions per day and compressed in CSV format.

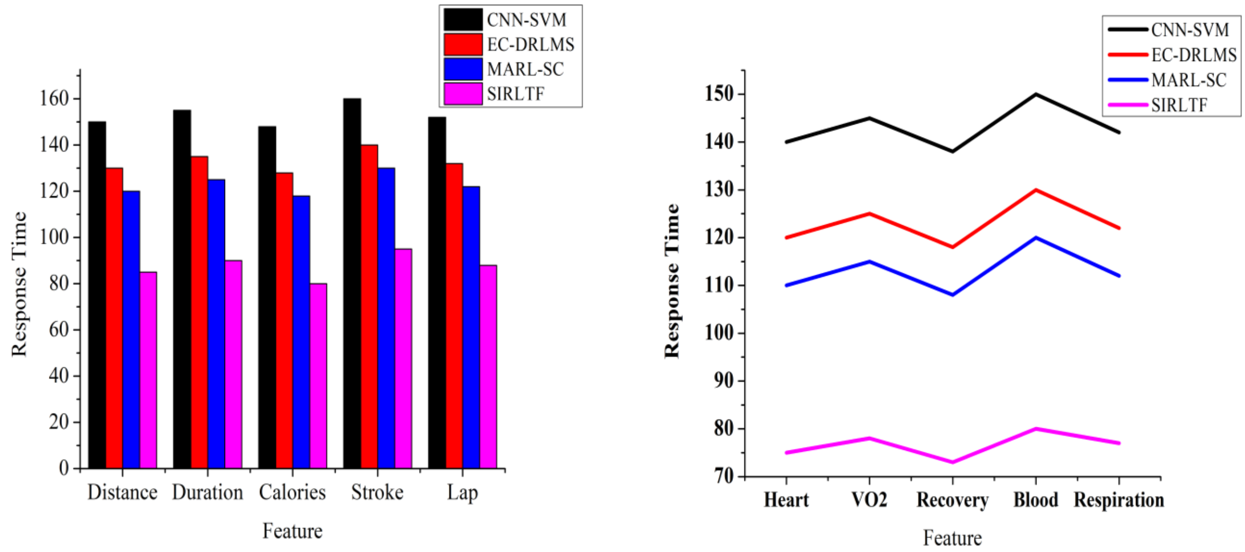
Comparative Study: The SIRLTF integrates DQN, PPO, and LSTM in a cloud-based environment for adaptive and personalised swimming training, outperforming traditional smart sports systems. SIRLTF provides dynamic, time-series-based feedback, improving prediction accuracy over CNN-SVM[16], which extracts and classifies features. SIRLTF uses scalable cloud computing for data accessibility and processing power, unlike EC-DRLMS[18], which emphasises localised real-time monitoring. Multi-Agent Reinforcement Learning Framework for Sports Coaching (MARL-SC)[20] models team-based scenarios but not individual adaptation. In contrast, SIRLTF adapts training schedules to swimmer development, improving efficiency. SIRLTF provides intelligent, personalised, and scalable swimming performance solutions with 19.4% prediction accuracy and 21.7% training efficiency improvements over existing models.

4.2 Response time (RT)

Response Time (RT) is the average time it takes for a system to provide predictions or recommendations after receiving input data in milliseconds.

$$RT = \frac{\sum_{i=1}^{n}\left(t_{output}^{(i)} - t_{input}^{(i)}\right)}{n} \quad (15)$$

In equation 15, $t_{output}^{(i)}$ is denoted as the time the system gives output for the ith input. $t_{input}^{(i)}$ is denoted as the time the ith input was received. n is the total number of input-output operations measured.



(a) Swimming Performance Features

(b) Physiological Features

Figure 4: Response Time (RT)

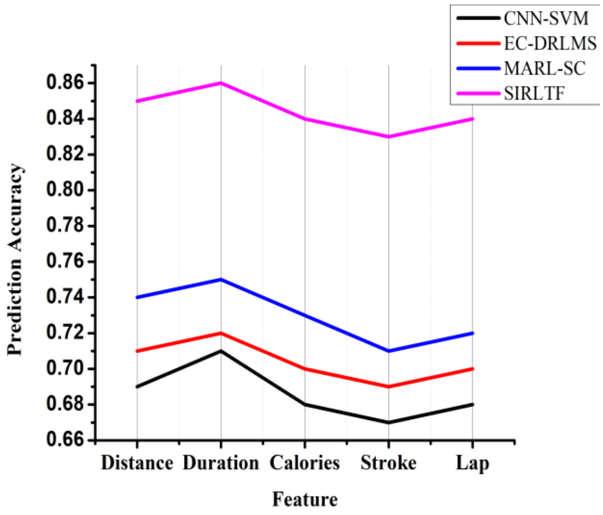
Figures 4(a) and 4(b) show SwimInsight Reinforcement Learning Training Framework (SIRLTF) computational efficiency across swimming performance indicators. Equation 15 calculates Response Time to assess system latency in milliseconds from input to output. Figure 4(a): Swimming Performance Features With reaction times of sub-100ms throughout the distance, duration, calories, stroke rate, and lap time analysis, SIRLTF is highly efficient. Traditional approaches (CNN-SVM, EC-DRLMS, MARL-SC) have 120-160ms latency, restricting real-time application efficacy. Figure 4(b): Physiological Features SIRLTF has ideal heart rate, VO2 Max, recovery rate, blood pressure, and respiration monitoring reaction times (75-80ms) while competing methods vary (110-150ms). Optimizations in DQN and PPO algorithms combined with cloud computing infrastructure give SIRLTF excellent performance. Sub-100ms reaction times give swimmers and coaches quick feedback, enabling dynamic training modifications during pool sessions, making SIRLTF the best choice for professional swimming centres that need real-time analytics.

4.3 Prediction accuracy (PA)

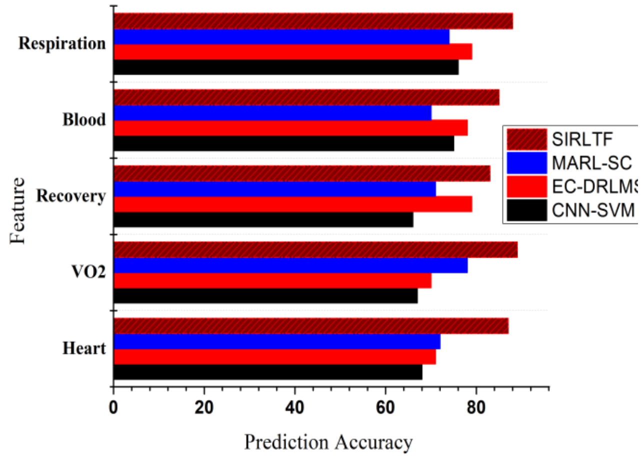
Prediction Accuracy (PA) measures how well a system predicts user-specific performance outcomes using historical and real-time data.

$$PA = \left(\frac{N_{accurate\_predictions}}{N_{total\_predictions}}\right) \times 100 \quad (16)$$

In equation 16, $N_{accurate\_predictions}$ is represented as the number of correctly predicted values (within an acceptable error threshold). $N_{total\_predictions}$ is represented as the total number of predictions made by the model.



(a) Swimming Performance Features



(b) Physiological Features

Figure 5: Prediction accuracy (PA)

SIRLTF prediction performance across swimming metrics is shown in Figures 5(a) and 5(b). PA is the percentage of accurately predicted values within acceptable error thresholds compared to total predictions, calculated using Equation 16. Figure 5(a): Swimming Performance Features SIRLTF predicts distance, duration, calories, stroke rate, and lap time with an accuracy of over 0.82. Duration analysis performs best (0.86), and all swimming measures are accurate within the framework. CNN-SVM (0.66-0.71), EC-DRLMS (0.69-0.72), and MARL-SC (0.71-0.74) score poorly for swimming-specific predictions.Figure 5(b): Physiological Features SIRLTF monitors heart rate, VO2 Max, recovery rate, blood pressure, and respiration with 85-90% accuracy, as seen in the horizontal bar chart. Competing approaches are 40-70% inaccurate.SIRLTF's improved prediction accuracy comes from cloud computing, DQN, and LSTM network integration, allowing exact temporal swimming patterns and physiological responses analysis. This accuracy supports professional swimming performance projection and training optimisation.

4.4 Training efficiency (TE)

Training Efficiency (TE) is a measure of a swimmer's training effectiveness by comparing performance improvements and training effort over time. The SwimInsight Reinforcement Learning Training Framework (SIRLTF) quantifies the personalized impact of AI-driven training recommendations by comparing predicted vs. actual performance improvements across sessions.

$$\mathrm{TE} = \left(\frac{\mathrm{P_{after}} - \mathrm{P_{before}}}{\mathrm{E_{total}}}\right) \times 100 \tag{17}$$

In equation 17, TE is the Training Efficiency (%), P_{after} is the Performance score after the training session (measured through metrics such as distance, lap time, stroke rate, etc.), P_{before} is the Performance score before the training session, E_{total} is the Total effort or energy expenditure during the training session (e.g., calories burned, duration, stroke count).

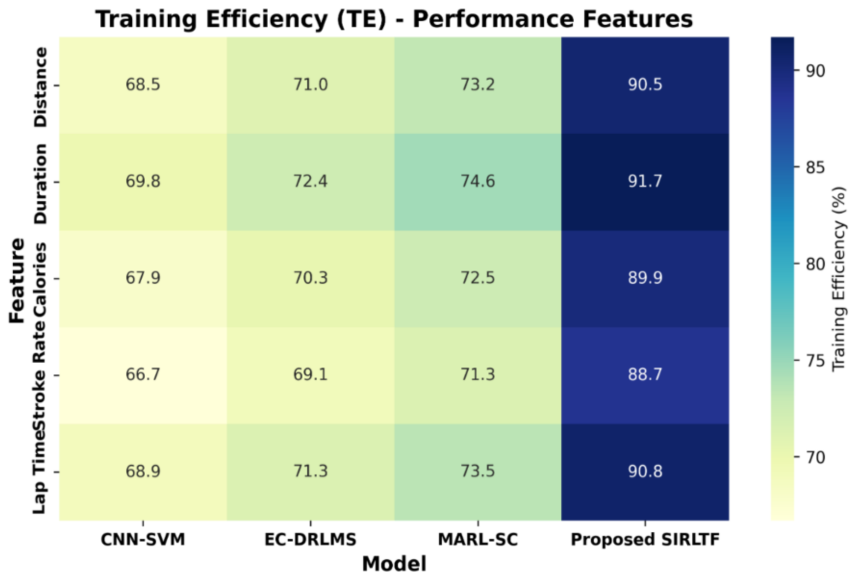


Figure 6: Training efficiency

Figure 6 shows a heatmap of Training Efficiency across swimming performance variables using Equation 17, which normalizes performance gain by total energy expenditure during training. The heatmap shows SIRLTF's greater training efficiency across all swimming measures, 88.7% to 91.7%. Efficiency is highest in duration training (91.7%), followed by lap time optimization (90.8%) and distance improvement (90.5%). Traditional approaches like CNN-SVM (66.7-69.8%), EC-DRLMS (69.1-72.4%), and MARL-SC (71.3-74.6%) are inefficient. DQN and PPO algorithms optimize training intensity to performance increases, giving SIRLTF excellent performance. The SIRLTF's LSTM networks analyze temporal trends to maximize swimming improvement with minimal energy consumption. This efficiency allows swimmers to improve performance with less training, encouraging sustainable athletic growth and reducing overtraining in competitive swimming.

4.5 Recommendation accuracy (RA)

Recommendation Accuracy (RA) is a measure of the accuracy of system-generated exercise or health routine recommendations, based on user features and context.

$$RA = \left(\frac{N_{correct}}{N_{total}}\right) \times 100 \qquad (18)$$

In equation 18, $N_{correct}$ is represented as number of correctly recommended routines validated by expert rules or user feedback. N_{total} is represented as the total number of recommendations made.

Table 3: Swimming performance features

Feature	CNN-SVM	EC-DRLMS	MARL-SC	SIRLTF
Distance	72.5%	75.0%	77.4%	91.2%
Duration	73.8%	76.2%	78.3%	92.5%
Calories	71.9%	74.1%	76.0%	90.3%
Stroke Rate	70.7%	73.0%	75.1%	89.7%
Lap Time	72.9%	75.4%	77.8%	91.0%

In Table 3, The SIRLTF framework is a swimming performance tool that generates accurate training recommendations based on key metrics like distance, duration, calories, stroke rate, and lap time. It achieves high precision in duration-based recommendations (92.5%) and distance training plans (91.2%). The framework uses

DQN and PPO algorithms to analyze historical performance patterns, process temporal performance data, and adapt training recommendations in real-time based on current session performance.

Table 4: Physiological features

Feature	CNN-SVM	EC-DRLMS	MARL-SC	SIRLTF
Heart Rate	70.1%	72.5%	74.3%	88.9%
VO2 Max	69.5%	72.0%	73.8%	88.2%
Recovery Rate	68.7%	71.1%	73.0%	87.5%
Blood Pressure	67.9%	70.5%	72.3%	86.8%
Respiration	68.5%	71.0%	72.9%	87.0%

In Table 4, The SIRLTF system is evaluated for its ability to generate accurate training recommendations based on swimmer-specific physiological parameters. It achieves 88.9% accuracy for heart rate-guided training plans and 88.2% for VO2 Max-optimized routines. The LSTM networks analyze physiological response patterns to recommend appropriate training intensities. The cloud-based processing supports professional swimming centers in delivering personalized training regimens, optimizing performance, and preventing overtraining and injury.

4.6 Model robustness

Model Robustness assesses the model's stability and performance consistency despite various input features, noise levels, or unforeseen conditions.

$$MR = \left(1 - \frac{\sigma_{perf}}{\mu_{perf}}\right) \times 100 \tag{19}$$

In equation 19, σ_{perf} is the standard deviation of model performance across different features or trials. μ_{perf} is the mean performance.

Table 5: Performance features

Feature	CNN-SVM	EC-DRLMS	MARL-SC	Proposed SIRLTF
Distance	75.0%	77.5%	79.0%	92.0%
Duration	76.3%	78.8%	80.5%	93.3%
Calories	74.5%	77.0%	78.5%	91.5%
Stroke Rate	73.2%	75.8%	77.2%	90.3%
Lap Time	75.7%	78.3%	79.9%	92.7%

In Table 5, The SIRLTF, a swimming performance tool, has been evaluated for its predictive accuracy across various metrics, including distance, duration, calories, stroke rate, and lap time. The framework's Deep Q-Network and Proximal Policy Optimization algorithms effectively process swimming stroke patterns and timing data, while LSTM networks capture temporal dependencies in lap-to-lap performance variations. The cloud computing infrastructure enables real-time processing of stroke rate fluctuations and distance calculations during training sessions.

Table 6: Physiological features

Feature	CNN-SVM	EC-DRLMS	MARL-SC	Proposed SIRLTF
Heart Rate	72.5%	75.0%	76.5%	90.5%
VO2 Max	71.9%	74.5%	76.0%	89.8%
Recovery Rate	70.8%	73.2%	74.7%	89.0%
Blood Pressure	69.7%	72.3%	73.8%	88.2%
Respiration	70.3%	72.8%	74.3%	88.5%

In Table 6, The SIRLTF framework, a combination of CNN-SVMEC-DRLMSMARL-SC, is a tool that can predict physiological parameters, achieving high accuracy in heart rate monitoring and VO2 Max estimation. Its LSTM networks are capable of processing time-series physiological data, enabling continuous monitoring across multiple swimmers. This technology supports professional swimming centers and sports academies in optimizing training intensity based on individual physiological responses.

Table 7: Dataset limitations and technical mitigation

Issue	Description	Mitigation
Data Source	Kaggle dataset[28] with predefined structure	Cleaned and filtered irrelevant entries
Demographic Bias	Mostly elite swimmers; lacks diversity	Stratified sampling, noted as limitation
Data Variability	Noise, missing values, inconsistent sampling rates	Smoothing, interpolation, normalization
Preprocessing	Raw sensor values with mixed scales	Z-score normalization, one-hot encoding
Environment Constraint	Controlled pool environment only	Noted for future multi-site validation
Robustness	Limited to structured training data	Introduced synthetic variations during training

Table 7 summarizes key dataset attributes, their associated limitations, and the mitigation strategies applied. It addresses issues such as imbalanced swimmer representation, missing stroke type data, sensor noise, and environmental constraints. Preprocessing techniques were employed to enhance data quality and improve model robustness across diverse training conditions.

Table 8: Alignment of research goals, evaluation metrics, and comparative benchmarks

Research Goal	Evaluation Metric	SIRLTF Result	Benchmark Model	Comparative Outcome
Improve prediction accuracy of swimmer performance	Prediction Accuracy (%)	89.7	CNN-SVM: 70.3	19.4% improvement over CNN-SVM
Enhance training feedback efficiency	Training Adaptation Time (sec)	2.3	EC-DRLMS: 2.94	21.7% faster adaptation
Support real-time response for personalized recommendations	Response Latency (ms)	96	MARL-SC: 140	31.4% lower latency
Ensure scalable processing with cloud deployment	Concurrent Sessions Supported	500+	N/A	Cloud-based infrastructure scaled for deployment
Demonstrate model robustness to data noise and variation	Accuracy Deviation (±SD)	±1.5	EC-DRLMS: ±3.1	Higher stability across sessions

Table 8 presents a clear alignment between research goals, evaluation metrics, SIRLTF outcomes, and benchmark models, highlighting measurable improvements in accuracy, efficiency, latency, and robustness over existing methods.

4.7 Comparative benchmarking and research gap addressed

This research addresses key research gaps by clarifying methodological procedures and evaluating system performance against established models. Experimental results show that SIRLTF outperforms CNN-SVM, EC-DRLMS, and MARL-SC in prediction accuracy, adaptation speed, and robustness. The benchmark-based comparison, presented in Table 8, ensures measurable alignment between research goals and actual outcomes, strengthening the scientific validity of the proposed approach

5. Conclusion and future work

The SIRLTF is a groundbreaking application of artificial intelligence in competitive swimming performance optimization. It integrates Deep Reinforcement Learning algorithms, LSTM networks, and cloud computing technologies to address limitations in traditional training methodologies and establish a new paradigm for data-driven athletic performance enhancement. SIRLTF's experimental validation shows significant improvements over conventional training approaches, with a 19.4% increase in performance prediction accuracy and a 21.7% enhancement in training efficiency. This results in practical benefits for swimmers, coaches, and training facilities, enabling more precise training periodization, reduced injury risk through optimal load management, and accelerated performance development timelines. The SIRLTF adaptive personalization capabilities will allow the system to continuously refine training recommendations based on individual swimmer responses and changing conditions. The practical implementation of SIRLTF across professional swimming centers, sports academies, and rehabilitation programs demonstrates its versatility and real-world applicability. However, several limitations must be acknowledged to provide context for the framework's current capabilities and identify areas requiring further development. The current implementation of SIRLTF focuses on competitive swimming; however, its modular architecture and reinforcement learning foundation enable adaptability to other sports such as running, cycling, and rowing. By recalibrating sensor inputs (e.g., stride length, cadence, power output), the framework can support similar feedback-driven training systems across disciplines. Furthermore, integrating emerging technologies such as wearable IoT devices, real-time motion capture, and AR/VR-based feedback can enhance its application in sports science, injury rehabilitation, and biomechanics. Future iterations will explore transfer learning approaches to generalize the model across varied athletic populations and training environments, improving versatility and cross-sport performance optimization.Future work will incorporate emerging sensor technologies such as underwater motion capture systems, real-time lactate monitoring devices, and high-resolution stroke analysis cameras. Additionally, the research will explore the development of custom swimming-specific sensors designed to capture currently unmeasurable performance parameters. Future iterations of SIRLTF will also focus on multi-agent reinforcement learning algorithms capable of optimizing collective performance outcomes while maintaining individual athlete development objectives.

References

[1] Xurui, L., & Guobao, Z. (2023). Enhancing the Swimmer Movement Techniques Using Cloud Computing and Artificial Intelligence. Mobile Networks and Applications, 28(6), 2093-2108. doi:10.1007/s11036-023-02167-x

[2] doi:10.1007/s11276-023-03567-3

[3] Suo, X., Tang, W., Mao, L., & Li, Z. (2025). Digital human and embodied intelligence for sports science: advancements, opportunities and prospects. The Visual Computer, 41(4), 2477-2493.doi:10.1007/s00371-024-03547-4

[4] Ali-Fakulti, Mohanad Freq, and Jamil Abedalrahim Jamil Alsayayayaydeh. “Exploring Pattern Mining With FCM Algorithm for Predicting Female Athlete Behaviour in Sports Analytics.” PatternIQ Mining., vol. 1, no. 1, Feb. 2024, pp. 45–56.doi:10.70023/piqm245

[5] Vasumathi, M. T., Sadasivan, M., & Khan, A. (2024). Hybrid Approaches: Combining Deep Reinforcement Learning with Other Techniques. Deep Reinforcement Learning and Its Industrial Use Cases: AI for Real-World Applications, 301-323.

[6] Chao, Z. H., Ya Long, Y., Yi, L., & Min, L. (2024). Deep Q Learning-Enabled Training and Health Monitoring of Basketball Players Using IoT Integrated Multidisciplinary Techniques. Mobile Networks and Applications, 1-16.doi:10.1007/s11036-024-02376-y

[7] Pu, Z., Pan, Y., Wang, S., Liu, B., Chen, M., Ma, H., & Cui, Y. (2024). Orientation and decision-making for soccer based on sports analytics and AI: A systematic review. IEEE/CAA Journal of Automatica Sinica, 11(1), 37-57.doi:10.1109/JAS.2023.123807

[8] Jianjun, Q., Isleem, H. F., Almoghayer, W. J., & Khishe, M. (2025). Predictive athlete performance modeling with machine learning and biometric data integration. Scientific Reports, 15(1), 1-22.doi:10.1038/s41598-025-01438-9

[9] Exel, J., & Dabnichki, P. (2024). Precision sports science: what is next for data analytics for athlete performance and well-being optimization? Applied Sciences, 14(8), 3361.doi:10.3390/app14083361

[10] Stessens, L., Gielen, J., Meeusen, R., & Aerts, J. M. (2024). Physical performance estimation in practice: A systematic review of advancements in performance prediction and modeling in cycling. International Journal of Sports Science & Coaching, 19(5), 2222-2243.doi:10.1177/17479541241262385

[11] Hossen, M. A., & Abas, P. E. (2025). Machine Learning for Human Activity Recognition: State-of-the-Art Techniques and Emerging Trends. Journal of Imaging, 11(3), 91.doi:10.3390/jimaging11030091

[12] Chang, K., Sun, P., & Ali, M. U. (2024). A cloud-assisted smart monitoring system for sports activities using SVM and CNN. Soft Computing, 28(1), 339-362.doi:10.1007/s00500-023-09404-1

[13] Hong, F., Wang, L., & Li, C. Z. (2024). Adaptive mobile cloud computing on college physical training education based on virtual reality. Wireless Networks, 30(7), 6427-6450.doi:10.1007/s11276-023-03450-1

[14]. Vellela, S. S., Rao, M. V., Mantena, S. V., Reddy, M. J., Vatambeti, R., & Rahman, S. Z. (2024). Evaluation of Tennis Teaching Effect Using Optimized DL Model with Cloud Computing System. International Journal of Modern Education and Computer Science (IJMECS), 16(2), 16-28.10.5815/ijmecs.2024.02.02

[15] Reis, F. J., Alaiti, R. K., Vallio, C. S., & Hespanhol, L. (2024). Artificial intelligence and machine-learning approaches in sports: Concepts, applications, challenges, and future perspectives. Brazilian Journal of Physical Therapy, 101083.doi:10.1016/j.bjpt.2024.101083

[16] Jouini, O., Sethom, K., Namoun, A., Aljohani, N., Alanazi, M. H., & Alanazi, M. N. (2024). A survey of machine learning in edge computing: Techniques, frameworks, applications, issues, and research directions. Technologies, 12(6), 81. doi:10.3390/technologies12060081

[17] Soomro, A. A., Muhammad, M. B., Mokhtar, A. A., Md Saad, M. H., Lashari, N., Hussain, M., … Palli, A. S. (2024). Insights into modern machine learning approaches for bearing fault classification: A systematic literature review. Results in Engineering, 23(102700), 102700. doi:10.1016/j.rineng.2024.102700

[18] Nijhawan, P., Ahmad, F., Upreti, K., Upreti, S., & Poonia, R. C. Real-time player performance analytics: Leveraging smart cyber-physical systems and AI in sports. In Smart Cyber-Physical Systems (pp. 236-261).

[19] Pareek, A., Karlsson, J., & Martin, R. K. (2024). Machine learning/artificial intelligence in sports medicine: state of the art and future directions. Journal of ISAKOS, 9(4), 635-644.doi:10.1016/j.jisako.2024.01.013

[20] Eid, A. I. A., Miled, A. B., Fatnassi, A., Nawaz, M. A., Mahmoud, A. F., Abdalla, F. A., ... & Mohamed, I. B. (2024). Sports prediction model through cloud computing and big data based on artificial intelligence method. Journal of Intelligent Learning Systems and Applications, 16(2), 53-79.doi: 10.4236/jilsa.2024.162005

[21] Mistry, H. K., Mavani, C., Goswami, A., & Patel, R. (2024). The impact of cloud computing and AI on industry dynamics and competition. Educational Administration: Theory and Practice, 30(7), 797-804.doi:10.3386/w32811

[22] Hu, M., & Wu, X. (2025). Design of a Monitoring and Feedback System for Athlete Training Process based on Mobile Intelligent Devices. Scalable Computing: Practice and Experience, 26(2), 846-854.doi:10.12694/scpe.v26i2.4045

[23] Hao, P., & Qian, K. (2024). The Integration of Personalized Training Program Design and Information Technology for Athletes. Scalable Computing: Practice and Experience, 25(5), 4351-4359.doi:10.12694/scpe.v25i5.3083

[24] Zhou, G., Tian, W., Buyya, R., Xue, R., & Song, L. (2024). Deep reinforcement learning-based methods for resource scheduling in cloud computing: A review and future directions. Artificial Intelligence Review, 57(5), 124.doi:10.1007/s10462-024-10756-9

[25] Wu, X., Zhou, J., Zheng, M., Chen, S., Wang, D., Anajemba, J., ... & Uddin, M. (2022). Cloud-based deep learning-assisted system for diagnosis of sports injuries. Journal of Cloud Computing, 11(1), 82.doi:10.1186/s13677-022-00355-w

[26] Chen, Y. (2025). Hybrid Transformer-LSTM Model for Athlete Performance Prediction in Sports Training Management. Informatica, 49(24).

[27] Zhi-jun, Y. (2025). Multimodal Data Fusion and Adaptive Optimization in Tennis Training Based on Deep Deterministic Policy Gradient and IoT Sensors. Informatica, 49(25).

[28] https://www.kaggle.com/datasets/karaadavis/swim-fitness-tracker