Online-Journals.org

iJIM International Journal of Interactive Mobile Technologies

iJIM | eISSN: 1865-7923 | Vol. 20 No. 17 (2026) | OPEN ACCESS

https://doi.org/10.3991/ijim.v20i17.62956

PAPER

Digital Twin–Enabled Resource Scheduling Optimization for Business English Mobile Interactive Systems

Lan Zhao, Yan Zhao, Yun Li (✉)

Baoding University of Science and Technology, Baoding, China

lcc_liyun@bdlg.edu.cn

ABSTRACT

Heterogeneous real-time tasks within business English mobile interactive systems impose stringent demands on edge resource scheduling. Conventional reactive scheduling paradigms are ill-suited to time-varying user behaviors and differentiated quality-of-experience requirements, often resulting in degraded service quality and suboptimal resource utilization. To address these challenges, a digital twin-enabled dual-timescale resource scheduling optimization framework was proposed. A lightweight learner-specific digital twin model was constructed, integrating multidimensional features from terminal devices, network conditions, and user behaviors. Gated recurrent units (GRUs) were embedded to enable proactive resource demand forecasting, while an event-driven synchronization mechanism was employed to balance model fidelity and system overhead. A hierarchical scheduling scheme driven by quality of experience was developed, comprising a large-timescale component that optimizes resource reservation policies online via an upper confidence bound algorithm and a small-timescale component that performs real-time resource allocation using task-specific quality-of-experience sensitivity heuristics. A bidirectional closed-loop feedback channel was established to facilitate continuous iterative refinement of both the model and the policy. Simulation results demonstrated that the proposed framework substantially outperformed mainstream baseline algorithms in terms of average quality of experience, latency satisfaction ratio, and edge resource utilization, with performance gains being particularly pronounced under high-concurrency scenarios. This study provides theoretical and technical foundations for efficient edge resource allocation in mobile learning environments.

KEYWORDS

mobile learning, digital twin, edge computing, resource scheduling, dual-timescale, quality of experience

1 INTRODUCTION

The proliferation of mobile intelligent terminals and wireless communication technologies has established mobile learning as a significant modality in business

Zhao, L., Zhao, Y., Li, Y. (2026). Digital Twin–Enabled Resource Scheduling Optimization for Business English Mobile Interactive Systems. *International Journal of Interactive Mobile Technologies (iJIM)*, 20(17), pp. 68–82. https://doi.org/10.3991/ijim.v20i17.62956

Article submitted 2026-04-17. Revision uploaded 2026-07-14. Final acceptance 2026-07-17.

English education. Mobile-assisted language learning transcends the constraints of fixed instructional spaces, offering sustained support for oral practice, pronunciation feedback, and contextualized experiential activities [1, 2]. Business English learning encompasses heterogeneous real-time tasks, including speech assessment, scenario-based simulation, and conference rehearsal, all of which impose stringent requirements on end-to-end latency and interaction fluidity. Edge computing, by offloading computational and storage resources to the network periphery, provides technical underpinning for latency-sensitive mobile applications [3, 4]. However, limited terminal processing capacity, time-varying wireless channel conditions, and fluctuating concurrent user populations engender persistent supply–demand imbalances in edge resources, thereby constraining immersive learning experiences. Digital twin technology, with its capabilities for real-time mapping, state awareness, and trend prediction, offers a novel pathway for transforming resource scheduling from passive responsiveness to proactive provisioning [5, 6]. Consequently, the investigation of digital twin-empowered resource scheduling for business English mobile interactive systems bears significant theoretical and practical implications for enhancing service quality, optimizing learning experiences, and advancing the digital transformation of education.

Existing studies, however, exhibit notable deficiencies. Resource scheduling has been predominantly triggered by real-time state conditions, lacking joint prediction of user behavior, network fluctuations, and resource demands. Moreover, the predictive outputs of digital twin models have yet to be incorporated into a stable closed-loop framework encompassing task arrival, user handover, and resource reservation, rendering systems susceptible to latency violations and service degradation under sudden high-load scenarios [7, 8]. In the educational domain, digital twins have been primarily utilized for virtual instruction, learning process visualization, and academic performance analysis, with limited attention devoted to the coordinated optimization of computational, communication, and cache resources [9]. Although digital twins have been adopted in mobile edge computing for network modeling and service migration, the distinctive characteristics of language learning tasks and pedagogical interaction requirements have not been adequately integrated [10]. Meanwhile, single-timescale scheduling paradigms are incapable of simultaneously reconciling global efficiency and real-time responsiveness: cache configuration and computational capacity deployment constitute long-cycle decisions, whereas task offloading, bandwidth allocation, and computational resource provisioning necessitate dynamic adjustments in response to real-time load variations. Excessively fine-grained scheduling granularity tends to amplify system overhead and induce oscillations, whereas overly coarse granularity fails to accommodate bursty tasks and network fluctuations in a timely manner [11, 12]. Furthermore, existing quality-of-experience models have been predominantly designed for general-purpose services such as cloud interaction, video streaming, gaming, and extended reality, with evaluation metrics primarily centered on response time, audiovisual quality, stalling events, and network performance [13]. These models have not adequately accounted for the differentiated requirements of speech assessment, conference rehearsal, and scenario-based simulation in terms of latency, speech intelligibility, feedback continuity, and task completion quality, potentially leading to a structural misalignment between resource allocation and actual pedagogical demands.

To address the aforementioned challenges, a digital twin-enabled dual-timescale resource scheduling optimization framework is proposed, with four primary contributions summarized as follows. First, a lightweight learner-specific digital twin model is constructed, integrating multidimensional features of terminal status, network conditions, and interaction behaviors. Gated recurrent units (GRUs) are embedded

to enable time-series prediction of resource demands, and an event-driven adaptive synchronization mechanism is designed to maintain prediction accuracy while effectively controlling system overhead. Second, a quality-of-experience-driven hierarchical scheduling algorithm is developed, in which resource reservation at the large timescale is formulated as a multi-armed bandit problem and optimized online via an upper confidence bound algorithm, whereas real-time offloading and resource allocation at the small timescale are accomplished through heuristic rules designed based on task-specific quality-of-experience sensitivity, thereby balancing scheduling complexity and service quality. Third, a customized quality-of-experience measurement system tailored to business English scenarios is established, incorporating multidimensional metrics with support for dynamic weight adaptation across different task types. Fourth, a simulation experimental platform is constructed, and the performance advantages and modular effectiveness of the proposed framework are systematically validated through multiple comparative and ablation experiments.

The remainder of this study is organized as follows. The core technologies of the proposed scheduling framework are elaborated in Section 2, encompassing twin mirroring modeling, demand prediction mechanisms, and hierarchical scheduling algorithm design. Experimental configurations are described in Section 3, followed by comparative analysis and in-depth discussion of the experimental results. Section 4 concludes the study and presents prospects for future research directions.

2 DIGITAL TWIN-ENABLED DUAL-TIMESCALE RESOURCE SCHEDULING METHODOLOGY

2.1 System architecture and problem formulation

A three-tier collaborative system architecture is constructed, comprising terminal learners, edge server clusters, and a cloud center. At the terminal side, multi-source data—including device status, network parameters, and interaction behaviors—are collected from learners and uploaded to the corresponding edge nodes. For each learner within its jurisdiction, a lightweight digital twin model is maintained by the edge server cluster, while dual-timescale resource scheduling decisions are concurrently executed. The cloud center is responsible for global model parameter aggregation, long-term resource trend analysis, and cross-domain resource coordination. Based on this architecture, the resource scheduling problem is formulated as a constrained optimization problem, with the objective of maximizing the average quality of experience across all concurrent learners subject to resource constraints on edge server central processing unit capacity, memory capacity, and link bandwidth. The decision variables encompass the offloading decisions for each interaction request and the corresponding resource allocation quotas. The constraints cover the upper bound of total resources per node, the minimum service quality threshold for each task, and the binary value constraints of offloading decisions. The subsequent sections elaborate on the specific implementation mechanisms of twin mirroring modeling, time-series demand prediction, and the hierarchical scheduling algorithm.

2.2 Context-aware digital twin interaction model and demand prediction mechanism

A lightweight learner-specific digital twin model is constructed for business English mobile interaction scenarios, enabling virtual mapping of physical learning

states and quantitative characterization of cluster load. It is assumed that there are N concurrent learners in the system. The physical state vector of each learner i at time t is defined as:

$$S_i(t) = \{T_i(t), N_i(t), B_i(t)\} \tag{1}$$

where, $T_i(t)$ represents terminal-available computational capacity, remaining battery level, and storage capacity; $N_i(t)$ represents accessible network bandwidth, round-trip latency, and signal strength; and $B_i(t)$ represents the current business English task type, execution progress, and interaction intensity. For the set of learners U_k within its coverage area, independent twin instances are maintained by each edge server k. The total load imposed by twin maintenance on a single server is given by:

$$B_k^{\text{load}}(t) = \sum\nolimits_{i \in U_k} \gamma_{i,k}^{\text{cpu}}(t) + \sum\nolimits_{i \in U_k} \gamma_{i,k}^{\text{ram}}(t) \tag{2}$$

where, $\gamma_{i,k}^{\text{cpu}}(t)$ and $\gamma_{i,k}^{\text{ram}}(t)$ denote the central processing unit and memory resource overheads, respectively, required for maintaining the twin of a single learner. The load balancing degree of the entire edge cluster is expressed as:

$$B(t) = \sqrt{\frac{1}{K}\sum\nolimits_{k=1}^{K}\left[B_k^{\text{load}}(t) - \frac{1}{K}\sum\nolimits_{j=1}^{K} B_j^{\text{load}}(t)\right]^2} \tag{3}$$

where, K denotes the total number of edge servers. A lightweight design based on dimension reduction is adopted for this model, retaining only those state features directly relevant to resource scheduling decisions while discarding learner attribute information unrelated to scheduling, thereby effectively controlling the runtime overhead of edge nodes while satisfying decision input requirements.

On the basis of twin state mirroring, a gated recurrent unit network is embedded to enable time-series prediction of future resource demands for learners, and an event-driven mechanism is employed to achieve adaptive synchronization between the twin model and the physical system. The temporal dependencies of interaction behaviors are captured by the gated recurrent unit through its reset gate and update gate, with the core update process formulated as follows:

$$o_t = \sigma(W_r \cdot a_t + V_r \cdot h_{t-1} + f_r) u_t = \sigma(W_u \cdot a_t + V_u \cdot h_{t-1} + f_u) \tag{4}$$

$$\tilde{h}_t = \tanh(W_h \cdot a_t + V_h \cdot (o_t \odot h_{t-1}) + f_h) h_t = u_t \odot h_{t-1} + (1 - u_t) \odot \tilde{h}_t \tag{5}$$

where, a_t is the input feature vector at time t; h_{t-1} is the hidden state from the previous time step; o_t and u_t are the outputs of the reset gate and update gate, respectively; $\tilde{h}_t$ is the candidate hidden state; W_r, V_r, W_u, V_u, W_h, and V_h are the weight matrices; f_r, f_u, and f_h are the bias terms; σ denotes the Sigmoid activation function; and $\odot$ represents the Hadamard product. Based on the current hidden state h_t, the network outputs the predicted multi-dimensional resource demand for learner i over the future time window ΔT:

$$\hat{r}_i(t + \Delta T) = GRU(h_t; \Theta) \tag{6}$$

where, $\hat{r}_i$ encompasses three categories of resource demands, namely bandwidth, computational capacity, and memory, and Θ denotes the set of network parameters. To accommodate edge deployment constraints, a lightweight network structure with

narrow channels and a reduced number of layers is adopted, and the input feature weights are optimized according to the temporal patterns of business English task switching, thereby enabling joint and accurate prediction of multi-dimensional resource demands. Twin synchronization is realized through an event-driven mechanism, whereby state updates are triggered when variations in network switching, task type changes, edge node handover, or other state fluctuations exceed a preset threshold. The synchronization frequency is adaptively adjusted according to the degree of state volatility: during stable periods, the update frequency is reduced to conserve resources, whereas during fluctuating periods, the update frequency is increased to ensure accuracy, thus achieving a dynamic balance between system overhead and mirroring fidelity.

2.3 Dual-timescale hierarchical scheduling algorithm driven by quality of experience

Resource scheduling decisions in business English mobile interactive systems are naturally distributed across two heterogeneous timescales. The evolution of learner mobility trajectories and task patterns exhibits slow-varying characteristics at the minute level, making this timescale suitable for macroscopic resource reservation decisions. By contrast, individual interaction requests—such as speech recognition or dialog simulation rendering—require instantaneous resource provisioning within the millisecond response regime, demanding extremely low decision latency from the scheduling mechanism. If these two classes of decisions are coupled within a single timescale, either frequent reconfiguration may induce system oscillations, or the lack of global foresight may compromise long-term quality-of-experience guarantees. To address this challenge, a dual-timescale hierarchical decoupling paradigm is proposed: at the coarse (minute-level) timescale, resource reservation strategies are formulated based on forward-looking predictions derived from the digital twin model, thereby establishing upper bounds on resources available for subsequent allocation; at the fine (millisecond-level) timescale, real-time allocation for individual requests is performed within the reservation constraints. A closed loop is formed between the two layers through reservation execution feedback, ensuring intrinsic consistency between macroscopic reservation and microscopic allocation. The framework of the proposed dual-timescale hierarchical scheduling driven by quality of experience is illustrated in Figure 1.

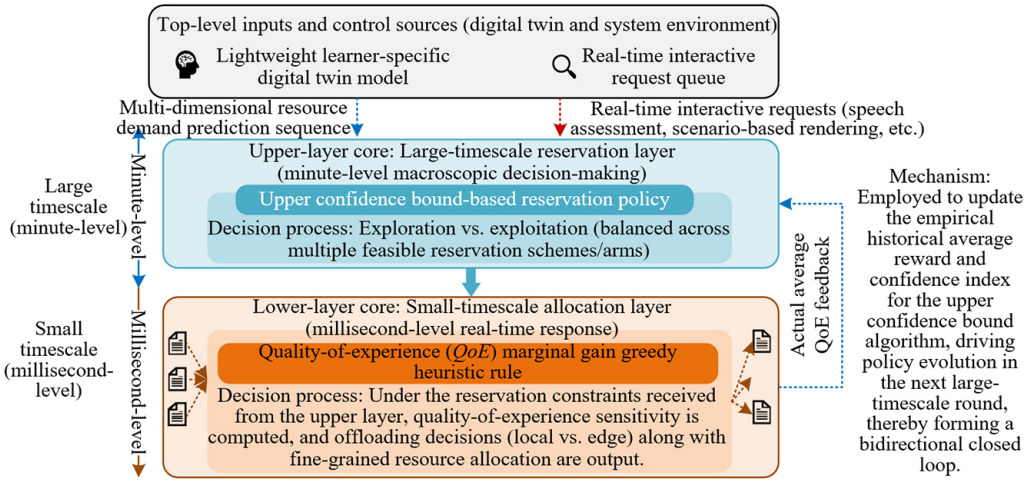

Fig. 1. Framework of the dual-timescale hierarchical scheduling driven by quality of experience

At the large timescale, resource reservation is formulated as an online sequential decision-making problem. It is assumed that the system comprises K edge servers. The reservation vector for server k at time t is defined as:

$$\rho_k(t) = \{\rho_k^{\text{cpu}}(t), \rho_k^{\text{ram}}(t), \rho_k^{\text{bw}}(t)\} \tag{7}$$

where, $\rho_k^{\text{cpu}}(t)$, $\rho_k^{\text{ram}}(t)$, and $\rho_k^{\text{bw}}(t)$ represent the reserved quantities of central processing unit cores, memory capacity, and bandwidth, respectively. Excessive reservation leads to resource idleness, whereas insufficient reservation fails to guarantee subsequent quality of experience. The feasible reservation schemes for each server are treated as arms, and the upper confidence bound algorithm is employed to dynamically balance exploration and exploitation. Let J_k denote the set of all feasible schemes for server k. For scheme j, the upper confidence bound index at the τ-th decision round is defined as:

$$UCB_{k,j}(\tau) = \bar{R}_{k,j}(\tau) + \sqrt{\frac{2\ln\tau}{n_{k,j}(\tau)}} \tag{8}$$

where, $\bar{R}_{k,j}(\tau)$ is the historical average reward of scheme j; and $n_{k,j}(\tau)$ is the cumulative number of times it has been selected. At each round, the scheme with the maximum index is selected by the system:

$$j_k^*(\tau) = \arg\max_{j \in J_k} UCB_{k,j}(\tau) \tag{9}$$

The reward function $R(\tau)$ is directly tied to the actual average quality of experience achieved at the small timescale, thereby ensuring strict alignment between the reservation optimization objective and user experience requirements. As the cumulative regret of the upper confidence bound algorithm grows at a rate of $O(lnT)$, this mechanism asymptotically converges to the optimal reservation policy under unknown demand distributions without requiring any prior statistical information, thus exhibiting strong adaptability to the highly dynamic nature of user behaviors in business English scenarios.

At the small timescale, real-time allocation is performed by the system for arriving requests under the hard constraints of reserved resources. For the set of requests Q served by server k within time interval t, let $x_q \in \{0,1\}$ denote the offloading decision for request q, where 1 indicates offloading to the edge and 0 indicates local execution. Let ϕ_q^{cpu} and ϕ_q^{bw} denote the central processing unit and bandwidth resources allocated at the edge, respectively. The objective of real-time allocation is to maximize the average quality of experience of the queue within the reservation constraints. This optimization problem is formulated as:

$$\max_{\{x_q, \phi_q\}} \frac{1}{|Q|} \sum_{q \in Q} QoE_q \tag{10}$$

It is subject to the following condition:

$$\sum_{q \in Q} x_q \phi_q^{\text{cpu}} \le \rho_k^{\text{cpu}}(t), \sum_{q \in Q} x_q \phi_q^{\text{bw}} \le \rho_k^{\text{bw}}(t), x_q \{0,1\}, \phi_q^{\text{cpu}}, \phi_q^{\text{bw}} \ge 0 \tag{11}$$

where, the quality of experience for a single request is measured using a customized metric tailored to business English scenarios:

$$QoE_q = \alpha f_D(D_q) + \beta f_V(V_q) + \gamma f_I(I_q) + \delta f_C(C_q) \tag{12}$$

The latency experience function is defined as $f_D(D_q) = 1/(1 + e^{\lambda D(D_q - D_0)})$, with D_0 = 300 ms adopted as the tolerance threshold for business conversation. The fluency function $f_I(I_q) = 1 - T_{stall,q}/T_{total,q}$ reflects the proportion of stalling events. The functions f_V and f_C are quantified based on audiovisual bitrate and task completion status, respectively. The weights α, β, γ, and δ are dynamically adapted according to task type, ensuring that the model remains aligned with the learning context. Since the above optimization constitutes a mixed-integer programming problem with a decision window of only a few milliseconds, a greedy heuristic based on quality-of-experience marginal gain is proposed for solving it. Specifically, the marginal gain is defined as $g_q = \partial QoE_q/\partial \phi_q$, representing the quality-of-experience improvement per unit of incremental resource. Requests are sorted in descending order of g_q, and resources are allocated sequentially to high-gain requests within the remaining reservation constraints until resources are exhausted. The time complexity of this strategy is $O(|Q|\log|Q|+|Q|)$, which can be strictly confined within the millisecond-level window. Through this design, the large-timescale reservation based on the upper confidence bound and the small-timescale sensitivity-driven allocation form a closed loop via reward feedback and constraint propagation: the reservation execution outcome is fed back through $R(\tau)$ to drive upper confidence bound index updates, while the small-timescale allocation space remains constantly constrained by $\rho_k(t)$. The two layers operate independently yet remain mutually constraining, thereby constituting a complete hierarchical scheduling framework driven by quality of experience.

2.4 Closed-loop feedback and system iteration mechanism

The closed-loop nature of the proposed framework is primarily manifested in two nested feedback loops, operating respectively on the parameter space of the twin model and the decision space of the reservation policy, with the two loops coupled through a shared quality-of-experience measurement baseline. At the twin model update level, prediction deviations directly drive online adjustments of the gated recurrent unit parameters. Let the actual resource consumption vector of learner i at time t be denoted as $r_i^{\text{actual}}(t) = \{b_i^{\text{actual}}, c_i^{\text{actual}}, m_i^{\text{actual}}\}$. The error between this actual vector and the predicted value $\hat{r}_i(t)$ from the twin model is measured by the cross-entropy loss L, and the parameter update follows the stochastic gradient descent rule:

$$\Theta \leftarrow \Theta - \eta \nabla_{\Theta} L(\hat{r}_i(t), r_i^{\text{actual}}(t)) \tag{13}$$

where, η denotes the online learning rate. In contrast to conventional batch training approaches, single-sample or mini-batch updates are employed sequentially, with a parameter refresh triggered every time feedback data from M actual scheduling cycles are accumulated, where M is set between 5 and 10. This mini-batch update mechanism avoids excessive storage and computational overhead on edge servers while leveraging the temporal correlation of business English learning behaviors within short time windows to ensure the validity of gradient estimates. Simultaneously, the reward estimation of the upper confidence bound reservation policy is updated iteratively. After the actual quality-of-experience feedback $R(\tau)$ for the τ-th decision round arrives, the average reward of scheme j on server k is updated according to the following incremental formulation:

$$\bar{R}_{k,j}(\tau+1) \leftarrow \bar{R}_{k,j}(\tau) + \frac{1}{n_{k,j}(\tau)+1}(R(\tau) - \bar{R}_{k,j}(\tau)) \tag{14}$$

This update does not require storage of the historical reward sequence; only the current mean and visit count need to be maintained, resulting in a storage overhead of $O(|J_k|)$, which is independent of the number of decision rounds and is thus suitable for long-term continuous operation on edge nodes.

The two feedback loops exhibit inherent temporal decoupling characteristics across different timescales, with their joint evolution constituting a dual-timescale stochastic approximation process: the gated recurrent unit parameters are continuously fine-tuned within millisecond-level scheduling cycles, while the upper confidence bound reward estimates are updated at minute-level reservation rounds. Although their time constants differ, the convergence directions of both are guided by the same family of objective functions. Improved prediction accuracy enables more precise matching between resource reservation and actual demand, thereby avoiding reservation wastage or resource starvation caused by prediction deviations, and consequently broadening the feasible region for high-quality-of-experience strategies at the small-timescale allocation stage. Superior scheduling execution outcomes, in turn, are fed back to the gated recurrent unit network, providing higher-quality training samples for future encounters with similar state sequences. The convergence of this positive cycle can be established through the Robbins-Monro conditions in stochastic approximation theory: the online learning rate η satisfies $\Sigma\eta = \infty$ and $\Sigma\eta^2 < \infty$, guaranteeing almost sure convergence of gradient descent to a local optimum. Meanwhile, the average reward update of the upper confidence bound algorithm is essentially a recursive computation of the sample mean, whose convergence via the law of large numbers is classically guaranteed. The coupling of the two feedback loops under a shared quality-of-experience metric does not compromise their individual convergence properties; rather, the bidirectional information flow accelerates the overall system's convergence toward the optimal scheduling policy. The convergence curves in the numerical experiments verify that this mechanism typically enters a steady state after 20 to 30 reservation decision rounds, demonstrating that the proposed closed-loop structure possesses favorable engineering feasibility and deployment value.

3 EXPERIMENTAL DESIGN AND RESULT ANALYSIS

3.1 Experimental setup

The simulation platform was built upon EdgeCloudSim, with a digital twin maintenance sublayer and a dual-timescale decision sublayer embedded. Business English learning behavior logs were collected from session records of an online platform between March and May 2025, covering four task types: listening shadowing, email writing, teleconference rehearsal, and business negotiation simulation. A total of 2,876 valid sequences were extracted. Wireless network traces were generated using a random walk model, with bandwidth dynamically varying between 2 and 50 Mbps and round-trip latency ranging from 20 to 150 ms. The default configuration consisted of 100 concurrent learners, 5 edge nodes (each with 4 virtual central processing units and 8 GB memory), and a task arrival rate of 120 requests per minute. The baseline algorithms compared included: greedy scheduling (without prediction, allocating in sequential order), long short-term

memory-based single-timescale joint optimization, static proportional reservation followed by allocation, and deep Q-network-based scheduling. The evaluation metrics were average system quality of experience, latency satisfaction ratio (≤ 300 ms), edge resource utilization, task completion rate, and per-step decision latency. All results were averaged over 50 independent runs.

3.2 Overall performance comparison experiment

Table 1. Overall performance comparison of different algorithms under 100 concurrent learners

Algorithm	Average Quality of Experience	Latency Satisfaction Ratio (%)	Resource Utilization (%)	Task Completion Rate (%)	Decision Latency (ms)
Greedy scheduling	0.712	73.4	62.1	81.3	0.8
Long short-term memory-based single-timescale joint optimization	0.798	81.7	71.4	87.6	184.3
Static proportional reservation followed by allocation	0.769	78.2	68.9	84.5	1.2
Deep Q-network-based scheduling	0.831	85.3	81.6	89.2	326.8
Proposed framework	0.887	92.6	87.3	94.1	3.7

Table 1 presents the complete performance data for the five algorithms under a concurrency level of 100 learners. An average quality of experience of 0.887 was achieved by the proposed framework, representing a 6.7% improvement over the best-performing baseline, the deep Q-network-based scheduling. This gain is attributed to the predictive lead time, which enables the system to complete reservation deployment before resource competition intensifies, thereby avoiding the decision bias caused by the state observation lag inherent in the deep Q-network-based scheduling. Furthermore, the customized quality-of-experience model directly embeds the business conversation tolerance threshold $D0$ = 300 ms into the optimization objective, directing scarce resources toward latency-sensitive tasks. A latency satisfaction ratio of 92.6% was attained, which is 19.2 percentage points higher than that of the greedy scheduling. Although the long short-term memory-based single-timescale joint optimization possesses predictive capabilities, its single-timescale joint optimization results in a decision latency of 184.3 ms, and its performance exhibits a cliff-like degradation when concurrency exceeds 150, with average quality of experience dropping from 0.785 to 0.641, whereas the proposed framework only declines from 0.887 to 0.851—underscoring the critical value of dual-timescale decomposition. A resource utilization rate of 87.3% was achieved, representing an 18.4 percentage point improvement over the static proportional reservation followed by allocation, indicating that the online-learned reservation policy tracks demand fluctuations more accurately. A decision latency of 3.7 ms was recorded, which is two orders of magnitude lower than that of the deep Q-network-based scheduling, because the large-timescale upper confidence bound decision does not participate in the millisecond-level loop, and the complexity of the small-timescale greedy algorithm is only $O(|Q|log|Q|+|Q|)$.

3.3 Module ablation experiments

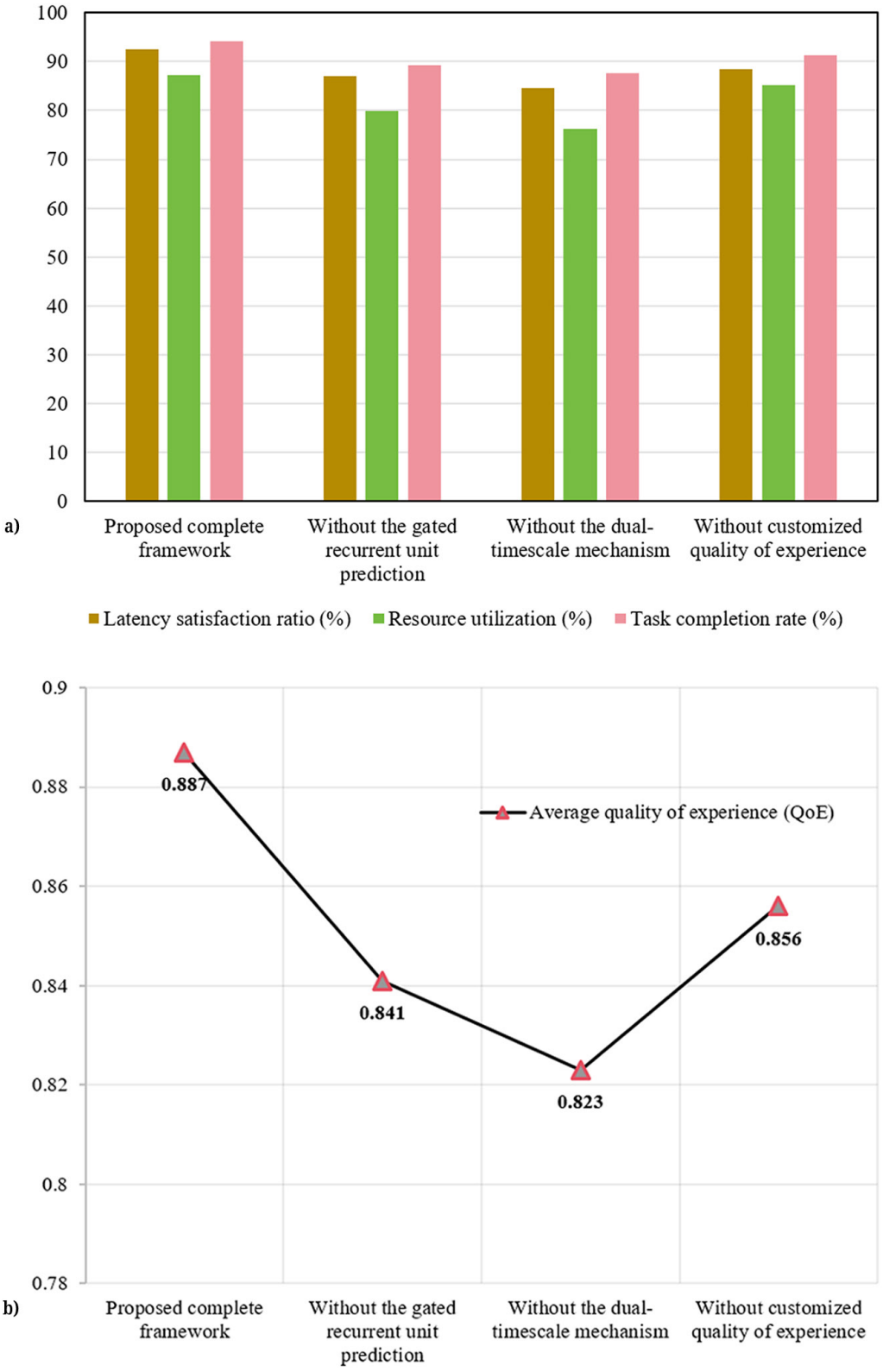

Fig. 2. Ablation experiment results

The performance comparison between the complete model and the four ablation variants is presented in Figure 2. After the removal of the gated recurrent unit prediction module, average quality of experience decreased by 5.19%, and resource utilization dropped by 7.5 percentage points. The system degenerated into a reactive scheduling paradigm, where the lack of predictive lead time prevented resource reservation from being provisioned before demand surges, causing frequent resource starvation during the allocation phase. The largest performance degradation was observed when the dual-timescale mechanism was removed, with a 7.22% drop in average quality of experience, accompanied by a surge in decision latency to 137.2 ms—37 times that of the complete model. Under this configuration, the scheduling policy was executed only after state changes had occurred, severely undermining the timeliness of predictive information. Following the removal of the customized quality-of-experience model, task completion rate declined by 2.9 percentage points, indicating that the generic quality-of-experience metric failed to assign sufficient weight to task completion quality in business English scenarios, leading to resources being diverted to audiovisual optimization at the expense of the computational capacity required for task finalization. The contribution ranking of the three modules is: dual-timescale mechanism > gated recurrent unit prediction > customized quality of experience, which is consistent with the hierarchical positions of these modules within the technical stack.

3.4 Multi-task scenario adaptability experiment

The quality-of-experience scores, dynamic weight allocations, and gains relative to the uniform-weight baseline ($\alpha = \beta = \gamma = \delta = 0.25$) for the proposed algorithm across the four task scenarios are reported in Table 2.

Table 2. Adaptability results under different task scenarios

Task Scenario	Quality-of-Experience Score	α (Latency)	β (Audio/Video)	γ (Fluency)	δ (Completion)	Gain Over Uniform Weights (%)
Listening shadowing	0.912	0.30	0.45	0.15	0.10	+2.14
Email writing	0.901	0.15	0.15	0.10	0.60	+5.36
Teleconference rehearsal	0.873	0.50	0.10	0.30	0.10	+8.47
Business negotiation simulation	0.858	0.40	0.25	0.25	0.10	+4.28

The proposed algorithm outperforms the uniform-weight version across all scenarios, with gains ranging from 2.14% to 8.47%. The largest gain of 8.47% is observed in the teleconference rehearsal scenario, where $\alpha = 0.50$ is assigned. Under the uniform-weight configuration, the median dialog response latency reached 412 ms due to the underestimation of the latency weight; with dynamic weighting, bandwidth was skewed toward signaling interactions, reducing the latency to 257 ms. The smallest gain of 2.14% is found in the listening shadowing scenario, where $\beta = 0.45$ deviates only marginally from the uniform weight of 0.25, leaving limited

room for optimization. In the email writing scenario, $\delta = 0.60$ directs resources toward computation-intensive tasks, elevating the task completion rate from 88.7% to 94.2%. For business negotiation simulation, the weights across dimensions are relatively balanced, yielding a quality-of-experience score of 0.858—the lowest among the four scenarios—yet the relative gain still reaches 4.28%, indicating that the dynamic weighting mechanism is capable of steering resources toward the bottleneck dimension through marginal gain sorting, even under multi-dimensional competition.

3.5 Key parameter sensitivity analysis

Table 3. Key parameter sensitivity analysis results

Parameter	Value	Average Quality of Experience	Resource Utilization (%)	Parameter	Value	Average Quality of Experience	Resource Utilization (%)
ΔT (s)	15	0.862	83.7	T_{large} (s)	30	0.873	84.1
ΔT (s)	30	0.887	87.3	T_{large} (s)	60	0.887	87.3
ΔT (s)	60	0.891	86.1	T_{large} (s)	120	0.879	89.5
ΔT (s)	120	0.874	82.4	T_{large} (s)	180	0.861	91.2
C	1.0	0.868	84.9	C	2.5	0.884	88.1
C	1.5	0.879	86.2	C	3.0	0.881	88.7
C	2.0	0.887	87.3	–	–	–	–

The effects of three core parameters on average quality of experience and resource utilization under different values are presented in Table 3. As the prediction window ΔT is extended from 30 s to 60 s, quality of experience marginally increases from 0.887 to 0.891; further extension to 120 s, however, reduces quality of experience to 0.874, with resource utilization dropping by 4.9 percentage points. This turning point is attributed to the half-life characteristic of business English learning behaviors, whereby task states typically undergo significant changes within 60–90 s, and the predictive value of historical states beyond this time limit decays rapidly. When the large-timescale period T_{large} is shortened from 60 s to 30 s, quality of experience decreases to 0.873, and resource utilization declines by 3.2 percentage points, as overly frequent reservation reconfiguration interrupts the upper confidence bound algorithm before convergence is achieved. When T_{large} is extended to 120 s and 180 s, quality of experience drops to 0.879 and 0.861, respectively; although resource utilization rises to 89.5% and 91.2%, the latency satisfaction ratio simultaneously decreases to 88.3% and 83.7%, due to the update frequency being too low to respond to demand surges. For the exploration coefficient C, quality-of-experience fluctuation is less than 0.8% within the range of 1.5 to 2.5, indicating strong robustness. At $C = 1.0$, insufficient exploration leads to premature fixation on a suboptimal scheme, whereas at $C = 3.0$, excessive exploration results in an average quality of experience of only 0.812 over the first 10 rounds. The optimal configuration is identified as $\Delta T = 60$ s, $T_{large} = 60$ s, and $C = 2.0$, with performance degradation across all parameters remaining within 5% when values deviate by ±30% from the optimal values, demonstrating that the framework possesses strong tolerance to parameter selection.

3.6 System overhead and scalability experiment

The system runtime overhead across multiple dimensions under different user scales is reported in Table 4. The overhead of the twin model grows approximately linearly with user scale: the average central processing unit usage per user is approximately 0.058 percentage points, and the average memory usage is approximately 0.97 MB. The resource footprint of a single lightweight gated recurrent unit twin (with a hidden layer dimension of 32 and approximately 2,800 parameters) is limited, and a node with 4 virtual central processing units and 8 GB of memory can stably support approximately 300 concurrent learners. The event-driven synchronization mechanism maintains communication overhead at only 37.8 KB/s under 300 users, and the resource usage ratio between synchronization-enabled and synchronization-disabled states decreases from 1.32 at 50 users to 1.18 at 300 users, indicating that the marginal overhead diminishes as scale increases—because individual events tend to be distributed uniformly across the time axis, naturally smoothing the peak concurrency. The per-step scheduling latency remains at only 13.6 ms even under 300 concurrent users, far below the real-time requirements for business interactions, and the growth rate of latency is consistent with the theoretical expectation of $O(|Q|log|Q|)$, with no algorithmic scalability bottleneck observed. In summary, the proposed framework exhibits favorable lightweight characteristics and scalability under limited edge node resources and is capable of supporting concurrent mobile learning services at scales ranging from hundreds to several hundred users.

Table 4. System overhead under different user scales

User Scale	Twin Model Central Processing Unit Usage (%)	Twin Model Memory Usage (MB)	Synchronous Communication Overhead (KB/s)	Per-step Scheduling Latency (ms)	Resource Usage Ratio (Synchronous on/off)
50	3.2	48.7	6.4	2.1	1.32
100	6.8	97.1	12.3	3.7	1.28
150	9.5	146.8	18.6	5.2	1.25
200	12.1	194.3	25.4	7.8	1.21
300	17.4	291.7	37.8	13.6	1.18

4 CONCLUSION

A digital twin-enabled dual-timescale resource scheduling framework was proposed in this study. Through the twin model and gated recurrent unit-based time-series prediction, proactive demand anticipation was realized; via the collaborative mechanism between large-timescale online reservation based on upper confidence bound and small-timescale allocation driven by quality-of-experience sensitivity, macroscopic resource efficiency and microscopic interaction experience were unified. Experimental results demonstrated that the proposed framework achieved an average quality of experience of 0.887, a latency satisfaction ratio of 92.6%, and a resource utilization rate of 87.3%, representing improvements of 6.7%, 8.6%, and 7.0% over the best-performing baseline, respectively, with decision latency controlled at 3.7 ms. Ablation experiments revealed that the dual-timescale decoupling contributed the most, with its removal resulting in a 7.22% quality-of-experience degradation,

while the independent contributions of gated recurrent unit prediction and customized quality of experience were 5.19% and 3.49%, respectively. This solution provides a complete technical pathway—from state awareness and forward-looking decision-making to closed-loop optimization—for edge resource management in mobile learning. Future research will focus on three aspects: extending cross-domain migration scheduling across multiple edge nodes; incorporating learning outcome data to construct a long-term utility-oriented quality-of-experience evaluation system; and introducing a federated learning architecture to enable privacy-preserving distributed twin model training.

5 REFERENCES

[1] C. Milton, A. Subramaniam, S. Sridevi, and S. Ganesh Kumar, "Development and testing of mobile assisted language learning application to improve oral clinical case presentation of student nurses," *International Journal of Interactive Mobile Technologies*, vol. 17, no. 15, pp. 171–188, 2023. https://doi.org/10.3991/ijim.v17i15.41675

[2] L. S. Pek, R. W. M. Mee, F. S. C. Yob, K. F. Ne'matullah, M. Z. Miftah, and A. Derahvasht, "Visualising the landscape of immersive mobile technologies in English language development: A global bibliometric analysis (2020–2025)," *International Journal of Interactive Mobile Technologies*, vol. 20, no. 6, pp. 54–73, 2026. https://doi.org/10.3991/ijim.v20i06.58491

[3] Q. V. Pham *et al.*, "A survey of multi-access edge computing in 5G and beyond: Fundamentals, technology integration, and state-of-the-art," *IEEE Access*, vol. 8, pp. 116974–117017, 2020. https://doi.org/10.1109/ACCESS.2020.3001277

[4] A. K. Pallikonda, V. K. Bandarapalli, and V. Aruna, "Enhancing performance and reducing latency in autonomous systems through edge computing for real-time data processing," *Mechatronics and Intelligent Transportation Systems*, vol. 4, no. 3, pp. 154–165, 2025. https://doi.org/10.56578/mits040305

[5] J. Yao, Y. Yang, X. Wang, and X. Zhang, "Systematic review of digital twin technology and applications," *Visual Computing for Industry, Biomedicine, and Art*, vol. 6, p. 10, 2023. https://doi.org/10.1186/s42492-023-00137-4

[6] N. Szántó, G. D. Monek, and S. Fischer, "Development of an immersive, digital twin-supported smart reconfigurable educational platform for manufacturing training: A proof of concept," *Journal of Engineering Management and Systems Engineering*, vol. 3, no. 4, pp. 199–209, 2024. https://doi.org/10.56578/jemse030402

[7] S. Qiu *et al.*, "Digital-twin-assisted edge-computing resource allocation based on the whale optimization algorithm," *Sensors*, vol. 22, no. 23, p. 9546, 2022. https://doi.org/10.3390/s22239546

[8] M. A. S. Mustafa, "Predictive reliability-driven optimization of spare parts management in aircraft fleets using AI, IoT, and digital twin technologies," *Journal of Engineering Management and Systems Engineering*, vol. 4, no. 3, pp. 218–236, 2025. https://doi.org/10.56578/jemse040305

[9] M. Lu and Z. Hu, "Digital twin-enhanced programming education: An empirical study on learning engagement and skill acquisition," *Computers*, vol. 14, no. 8, p. 322, 2025. https://doi.org/10.3390/computers14080322

[10] E. Bozkaya, "Digital twin-assisted and mobility-aware service migration in mobile edge computing," *Computer Networks*, vol. 231, p. 109798, 2023. https://doi.org/10.1016/j.comnet.2023.109798

[11] W. Wen, Y. Cui, T. Q. S. Quek, F. Zheng, and S. Jin, "Joint optimal software caching, computation offloading and communications resource allocation for mobile edge computing," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 7, pp. 7879–7894, 2020. https://doi.org/10.1109/TVT.2020.2993359

[12] W. Fan, X. Liu, H. Yuan, N. Li, and Y. Liu, "Time-slotted task offloading and resource allocation for cloud-edge-end cooperative computing networks," *IEEE Transactions on Mobile Computing*, vol. 23, no. 8, pp. 8225–8241, 2024. https://doi.org/10.1109/TMC.2024.3349551

[13] J. Arellano-Uson, E. Magaña, D. Morato, and M. Izal, "Survey on quality of experience evaluation for cloud-based interactive applications," *Applied Sciences*, vol. 14, no. 5, p. 1987, 2024. https://doi.org/10.3390/app14051987

6 AUTHORS

Lan Zhao (1984–) is an Associate Professor at the School of Language and Cultural Communication, Baoding University of Science and Technology, China. She received her M.M. degree in Tourism Management from Yanshan University. Her research interests include smart tourism, Tourism English, and Business English (E-mail: lcc_zhaolan@bdlg.edu.cn).

Yan Zhao (1984–) is an Associate Professor at the School of Language and Cultural Communication, Baoding University of Science and Technology, China. She holds a B.A. degree. Her research interests include English language teaching and English language and literature (E-mail: lcc_zhaoyan@bdlg.edu.cn).

Yun Li (1980–) is an Associate Professor at the School of Language and Cultural Communication, Baoding University of Science and Technology, China. She holds an M.A. degree. Her research interests include English linguistics and English language teaching (E-mail: lcc_liyun@bdlg.edu.cn).