

PAPER

A Generative AI-Enhanced Mobile Learning Framework for Adaptive Human–Machine Collaboration

Chen Zhang¹ ,
Zhongyuan Wu^{2,3}  

¹University of Birmingham,
Birmingham, United Kingdom

²Beijing Normal University,
Beijing, China

³Yanching Institute of
Technology, Langfang, China

wuzhongyuan77@126.com

ABSTRACT

Ubiquitous mobile learning has emerged as a central modality in higher education. While generative artificial intelligence offers significant potential for personalized learning, its deployment is constrained by two fundamental bottlenecks: the hard resource limitations of end-device computing and the inadequate adaptability of human–machine instructional roles. Existing systems predominantly rely on cloud-end binary scheduling and static role partitioning, which fail to simultaneously guarantee operational performance on mobile terminals and collaborative teaching effectiveness. To address these challenges, a three-tier progressive agent architecture was proposed. A cognition-driven progressive routing protocol was introduced to dynamically align computational resource levels with the cognitive load imposed by learning tasks. Local clustering pruning and cache gain mechanisms were further incorporated to optimize retrieval efficiency on the end-device side. Building upon this infrastructure, a cognition–collaboration dual-loop nested dynamic weighting model was formulated, in which technical architectural stability was incorporated as a decision variable within the framework. Smooth switching of the human–machine instructional roles was achieved through continuous weight computation, and an iterative closed loop with bidirectional coupling was established. This study provides a technical paradigm for the scalable deployment of generative artificial intelligence in mobile learning scenarios, addressing both engineering practicality and instructional adaptability.

KEYWORDS

mobile learning, generative artificial intelligence, agent architecture, human–machine collaboration, dynamic weighting, edge computing

1 INTRODUCTION

With the deep integration of mobile computing technologies and ubiquitous learning paradigms, mobile terminals have become the primary vehicle for fragmented and personalized learning among learners in higher education settings [1, 2]. The rapid evolution of generative artificial intelligence [3] has opened new pathways for overcoming the content supply limitations of traditional mobile learning [4] and for

Zhang, C., Wu, Z. (2026). A Generative AI-Enhanced Mobile Learning Framework for Adaptive Human–Machine Collaboration. *International Journal of Interactive Mobile Technologies (iJIM)*, 20(18), pp. 171–184. <https://doi.org/10.3991/ijim.v20i18.63066>

Article submitted 2026-06-12. Revision uploaded 2026-07-27. Final acceptance 2026-07-28.

© 2026 by the authors of this article. Published under CC-BY.

realizing highly adaptive intelligent learning support [5]. However, constrained by the hardware boundaries of mobile terminals—including computational power, memory, and energy consumption [6]—large language models cannot be efficiently deployed directly on end devices [7]. Cloud-based invocation modes, in turn, are susceptible to network conditions [8], rendering real-time interactivity and stability difficult to guarantee. Meanwhile, the design of existing mobile intelligent learning systems has been predominantly oriented toward technical performance metrics [9], with insufficient attention paid to the intrinsic regularities of pedagogical processes and the central role of human instructors. This has resulted in a pronounced misalignment between technological implementation and instructional needs. Therefore, the construction of a generative mobile learning environment that balances engineering efficiency with pedagogical appropriateness holds significant research importance and application value for promoting the large-scale deployment of intelligent educational technologies and for refining the theoretical framework of human–machine collaborative teaching.

In current mobile generative learning systems, computational resource scheduling has been predominantly implemented through a cloud–end binary architecture [10]. Scheduling decisions are typically based on physical metrics such as network latency and bandwidth, without establishing a precise mapping to the cognitive load characteristics of learning tasks [11]. This has resulted in low utilization of end-side computational resources, delayed responses to complex tasks, and redundant cloud computational power for simple tasks, such that a hierarchical scheduling system adapted to learning scenarios has yet to be formed. In the domain of human–machine collaborative teaching mechanisms, most studies have adhered to predefined static role assignment rules [12], which cannot be dynamically adjusted with respect to learners' cognitive states and task complexity, thus limiting the degree of adaptability. Moreover, collaborative decision models have generally incorporated only variables related to pedagogical agents [13], without including the operational stability of the underlying technical architecture as an evaluative dimension, leaving instructional continuity vulnerable to disruption by fluctuations in technical performance. At the level of system iteration logic, the technical architecture and teaching mechanisms have largely exhibited a unidirectional supportive relationship [14], with the vast amounts of behavioral data generated during instructional interaction lacking effective channels for targeted feedback to model optimization. Consequently, the system has been unable to achieve continuous autonomous evolution alongside the teaching process [15], and a virtuous cycle of synergistic gain between technology and pedagogy has proven difficult to establish.

To address the aforementioned research limitations, a construction methodology for a mobile interactive learning environment is proposed, which integrates a three-tier progressive agent architecture with a dual-loop nested weighting model. The core contributions of this work can be summarized in four aspects. First, a three-tier progressive agent architecture is proposed, incorporating a cognition-driven progressive routing protocol and a local cache gain optimization mechanism, through which dynamic matching between computational resource levels and cognitive load is achieved under the hard constraints of mobile terminal resources. Second, a cognition–collaboration dual-loop nested dynamic weighting model is constructed, in which the operational stability of the technical architecture is incorporated into the collaborative potential evaluation framework, and smooth human–machine role switching is realized through continuous weight computation with an inertia term. Third, an iterative closed loop with bidirectional coupling between the environment and the mechanisms is established, forming a complete chain encompassing computational scheduling, instructional decision-making, data feedback, and model iteration. Fourth, systematic validation is conducted across three

dimensions—engineering performance, mechanism effectiveness, and instructional outcomes—through five sets of multi-dimensional controlled experiments.

2 THREE-TIER AGENT PROGRESSION AND HUMAN–MACHINE DUAL-LOOP COLLABORATION METHOD

2.1 Overall design paradigm and coupled closed-loop logic

Design-based research is adopted as the overarching paradigm, following a reverse design logic in which technology empowers instruction and instruction, in turn, feeds back to technology. A unified operational closed loop is constructed, encompassing seven core stages: perception, retrieval, generation, evaluation, allocation, execution, and feedback. The traditional physical layering approach of cloud–edge–end is abandoned, and the computational supply chain of generative artificial intelligence is topologically aligned with the human–machine role allocation chain of the instructional process. The three-tier progressive agent architecture serves as the physical supporting substrate for the learning environment, while the cognition–collaboration dual-loop nested dynamic weighting mechanism functions as the control core of the instructional process. These two components form a deeply coupled organic whole, jointly supporting adaptive intelligent learning interactions in mobile scenarios.

The cognitive offloading depth of the three-tier progressive agent architecture directly determines the task complexity eigenvalues in the collaborative weighting model, corresponding to the activation states of different agent tiers. Conversely, the real-time human–machine role allocation results output by the collaborative weighting model guide the newly generated data from interactive processes to flow directionally to the corresponding agent tiers for incremental fine-tuning. Computational scheduling outcomes provide core inputs for instructional decision-making, while instructional process data provide iterative directions for architectural optimization. Driven bidirectionally by these two forces, a complete logical closed loop is formed, enabling the system to continuously optimize computational resource allocation efficiency and instructional collaborative adaptability under the resource constraints of mobile terminals.

2.2 Three-tier progressive agent architecture and cognitive routing mechanism

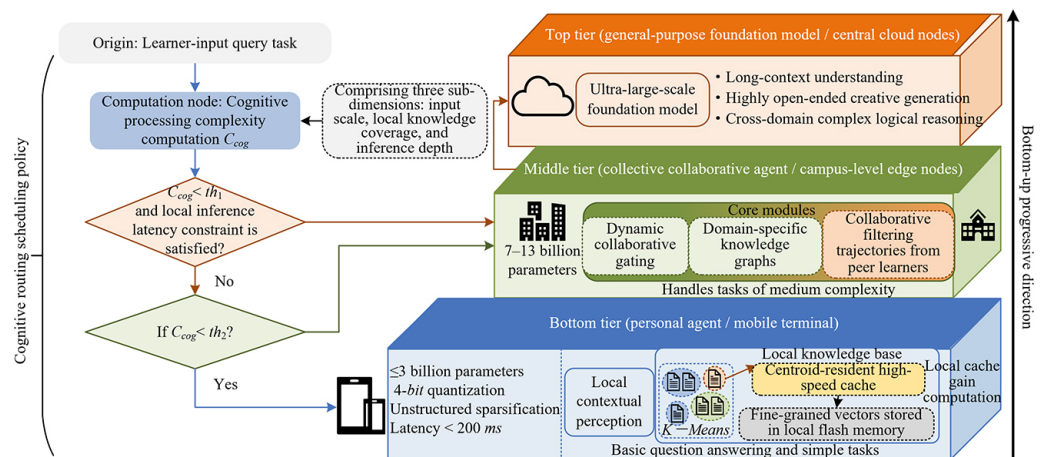


Fig. 1. Schematic diagram of the three-tier progressive agent architecture and cognitive routing mechanism

To alleviate the core conflict between the hardware resource constraints of mobile terminals and the high computational demands of large language models, a three-tier progressive intelligent computational network is constructed. From bottom to top, the three tiers are designated as the personal agent, the collective collaborative agent, and the general-purpose foundation model, with each tier dynamically activated according to the cognitive characteristics of the task, thereby forming a gradient-based computational supply system. Figure 1 presents a schematic diagram of the three-tier progressive agent architecture and cognitive routing mechanism. The personal agent resides on the learner's mobile terminal, where a lightweight foundation model with no more than 3 billion parameters is deployed using 4-bit weight quantization and unstructured sparsification techniques, with single-round inference latency controlled within 200 ms. This tier is responsible for local contextual perception, basic question answering, and retrieval-generation functions for simple tasks. The collective collaborative agent is deployed at campus-level edge computing nodes, capable of hosting medium-scale models with seven billion to 13 billion parameters. A dynamic collaborative gating mechanism is embedded within this tier, integrating domain-specific knowledge graphs and collaborative filtering trajectories from peer learners, and is responsible for handling tasks of medium complexity that exceed the capacity of the end-device tier. The general-purpose foundation model is deployed at central cloud nodes and is invoked only for tasks requiring long-context understanding, highly open-ended creative generation, and cross-domain complex logical reasoning, serving as the ultimate cognitive arbitration unit for the entire architecture.

To address the problems of limited mobile terminal memory capacity and the excessive cost of full-scale vector index deployment, a retrieval optimization strategy combining centroid-resident caching with on-demand generation is adopted at the personal agent tier. The document chunk set in the local knowledge base is clustered into K semantic clusters via the K-Means algorithm. Only the centroid vectors of each cluster are permanently cached in the terminal's high-speed cache, while the fine-grained document embedding vectors within each cluster are stored in local flash memory, thereby substantially reducing permanent memory occupancy. When a query request is processed, the system first performs coarse-grained filtering on the resident centroid set using the query vector, identifying the top t most relevant semantic clusters. Embedding vectors are then generated on demand for document chunks within the selected clusters, and fine-grained retrieval matching is executed. To further optimize cache configuration, a local cache gain function is defined to quantitatively evaluate the retention value of each cluster:

$$Gain_{PIA}(C_k) = \alpha \cdot f_k(t) \cdot L_{gen}(C_k) - \beta \cdot Mem(C_k) \quad (1)$$

where, C_k denotes the k -th semantic cluster; $f_k(t)$ denotes the query access frequency of that cluster within time window t ; $L_{gen}(C_k)$ denotes the on-demand generation time overhead for document embeddings within the cluster; $Mem(C_k)$ denotes the estimated memory space occupied by the cluster centroid and index information; and α and β are the time benefit coefficient and space cost coefficient, respectively. Subject to the total local memory constraint $\sum Mem(C_k) \leq M_{local}$, a greedy algorithm is employed to preferentially retain semantic clusters with the highest gain values, achieving a Pareto-optimal trade-off between retrieval speed and memory occupancy.

Unlike the scheduling logic in traditional cloud-end collaborative architectures, which relies on physical metrics such as network latency and bandwidth, a cognition-driven

progressive routing protocol is designed, in which task cognitive complexity serves as the core criterion for precise matching of computational tiers. To this end, the cognitive processing complexity at the Personal Agent tier is defined for a given query, integrating three dimensions: input scale, local knowledge coverage, and inference depth:

$$C_{PIA}(x) = \lambda_1 \cdot TokenLen(x) + \lambda_2 \cdot (1 - Covearge_{local}(x)) + \lambda_3 \cdot ReasonStep(x) \quad (2)$$

where x denotes the current learning query, $TokenLen(x)$ denotes the token length of the query text, $Covearge_{local}(x)$ denotes the semantic coverage of the query by the local knowledge base, $ReasonStep(x)$ denotes the theoretical minimum number of inference steps required to complete the task corresponding to the query, and λ_1 , λ_2 , and λ_3 are the weight coefficients for each dimension, calibrated through preliminary experimental data. Based on the computed cognitive complexity, routing decisions are made through a two-level threshold mechanism:

$$Route(x) = \begin{cases} PIA, C_{PIA}(x) \leq \theta_{pia} \cap Latency_{local} \leq 300ms \\ CSI, \theta_{pia} < C_{PIA}(x) \leq \theta_{csi} \\ GFM, C_{PIA}(x) > \theta_{csi} \end{cases} \quad (3)$$

where, θ_{pia} and θ_{csi} denote the cognitive load thresholds for the personal agent and the collective collaborative agent, respectively. An additional local inference latency constraint is imposed at the Personal Agent tier to ensure real-time responsiveness for end-side interactions. Through this routing mechanism, computational resource consumption is transformed into quantifiable cognitive load characteristics, enabling each mobile interaction to be matched to the agent tier with the optimal comprehensive cost, while balancing runtime performance and terminal energy consumption.

The three agent tiers are not operated as independently stacked modules; rather, they form an organically collaborative computational network through a unified routing protocol. The tier-switching process remains transparent to the learner, thereby ensuring continuity of the interaction experience. Furthermore, the cognitive complexity metrics output by the routing computation are directly fed as core input variables into the subsequent human–machine collaborative weighting model, achieving a natural articulation between underlying computational scheduling and upper-level instructional decision-making, and providing the data foundation for deep coupling between the architecture and the mechanism.

2.3 Cognition–collaboration dual-loop nested dynamic weighting allocation model

The inner loop performs real-time quantification of learners' cognitive states and task characteristics, providing core cognitive-dimensional inputs for human–machine collaborative decision-making. Based on item response theory, latent capability parameters of learners on each knowledge point are parameterized. Through Bayesian filtering methods, behavioral data from question-answering interactions—including accuracy, response latency, and revision frequency—are incorporated to iteratively update the posterior distributions of capability parameters, thereby enabling dynamic tracking of cognitive levels. Figure 2 presents a schematic diagram of the cognition–collaboration dual-loop nested dynamic weighting allocation model. For the cognitive load of learning tasks, a quantitative model is constructed

from three dimensions: knowledge dependency, inference depth, and openness attribute:

$$(T) = \gamma_1 K_d(T) + \gamma_2 R_d(T) + \gamma_3 (1 - o_p(T)) \quad (4)$$

where, $K_d(T)$ denotes the knowledge dependency of task T , characterizing the number and associative strength of prerequisite knowledge points required for task completion; $R_d(T)$ denotes the inference depth, corresponding to the number of logical reasoning layers required for task completion; $O_p(T)$ denotes the task openness, with values in the range $[0, 1]$, where higher values indicate weaker convergence of the task answer; and γ_1 , γ_2 , and γ_3 are the dimensional weights, calibrated by domain experts and optimized through preliminary experiments.

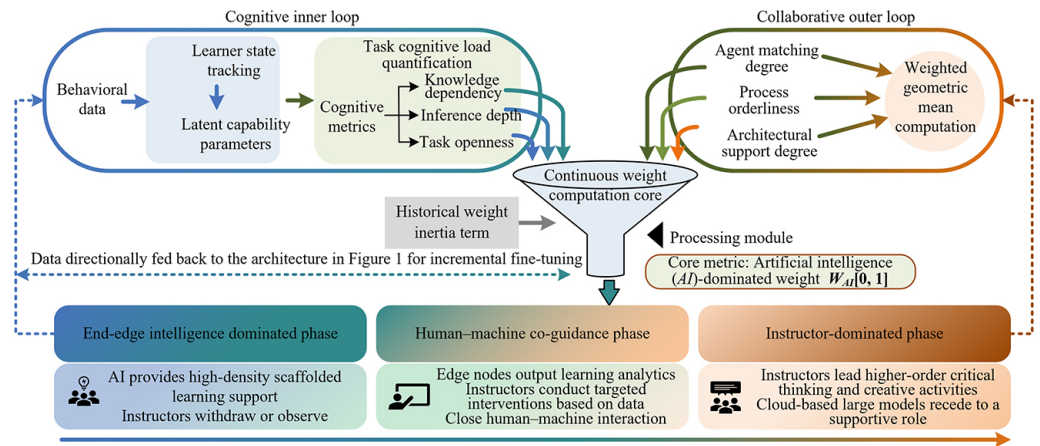


Fig. 2. Schematic diagram of the cognition–collaboration dual-loop nested dynamic weighting allocation model

The outer loop constructs a comprehensive collaborative potential evaluation system from the perspective of system operation, incorporating the operational stability of the underlying technical architecture as a constraint variable in collaborative decision-making. Traditional collaborative models have predominantly focused only on the capability matching of pedagogical agents, leaving collaborative effectiveness vulnerable to degradation caused by fluctuations in the technical environment. A weighted geometric mean formulation is adopted to compute the system's comprehensive collaborative potential, thereby reflecting the short-board constraint characteristics among dimensions:

$$\Phi_{total} = S_{subject}^{\alpha} \cdot S_{process}^{\beta} \cdot S_{TPAA}^{\gamma}, \quad \alpha + \beta + \gamma = 1 \quad (5)$$

where, $S_{subject}$ denotes the agent matching degree, measuring the degree of capability complementarity between the instructor's professional expertise and the currently activated agent tier; $S_{process}$ denotes the process orderliness, characterizing the semantic coherence and feedback immediacy of the current interaction sequence; S_{TPAA} denotes the architectural support degree, computed jointly from the smoothness of three-tier agent architecture tier switching and the anti-jitter capability of network transmission, reflecting the capacity of the underlying computational environment to support instructional collaboration; and α , β , and γ are the weight coefficients for the corresponding dimensions, determined through the analytic hierarchy process.

By fusing the outputs of the two loops, a continuous artificial intelligence-dominated weight computation model is constructed to achieve smooth dynamic allocation of human–machine instructional roles. To suppress frequent role switching caused by single-round decision fluctuations, a historical weight inertia term is introduced into the model. Through a sigmoid function, the multi-dimensional features are mapped to a weight interval of [0, 1]:

$$w_{AI}(\theta, \Psi, \Phi_{total}) = \sigma\left(\eta_1 \cdot (1 - \bar{\theta}) + \eta_2 \cdot \Psi(T) - \eta_3 \cdot \Phi_{total} + \eta_4 \cdot w_{AI}^{history}\right) \quad (6)$$

where, $\bar{\theta}$ denotes the learner’s average capability value on the knowledge points associated with the current task, $w_{AI}^{history}$ denotes the artificial intelligence-dominated weight from the previous interaction round, and η_1 through η_4 are the adjustment coefficients calibrated through preliminary experiments. The instructor-dominated weight is correspondingly given by $w_{Human} = 1 - w_{AI}$. Based on the weight values, the collaborative process is divided into three operational phases: when the weight exceeds 0.65, the end-side and edge intelligence agents dominate, providing high-density scaffolded learning support; when the weight falls within the interval [0.35, 0.65], the human–machine co-guidance phase is entered, in which the instructor conducts targeted interventions based on learning analytics output by edge nodes; when the weight falls below 0.35, the instructor-dominated phase is entered, in which the cloud-based model assumes only a retrieval-assistance function, and the instructor leads higher-order critical thinking and creative instructional activities.

This weighting model is deeply coupled with the three-tier progressive agent architecture. The task cognitive complexity output by the routing protocol serves directly as a core input to the weight computation, while the real-time operational status of the architecture synchronously updates the architectural support degree indicator within the collaborative potential. Conversely, the weight allocation results guide the interactive data to flow directionally back to the corresponding agent tiers, achieving bidirectional adaptation between instructional decision-making and technical architecture and ensuring the continuity and effectiveness of human–machine collaboration in mobile learning scenarios.

2.4 Iterative mechanism of bidirectional coupling between environment and mechanisms

Through a bidirectional coupling operational mechanism, the three-tier progressive agent architecture and the dual-loop nested weighting model form a unified iterative closed loop, enabling deep integration and co-evolution between the computational supply system and the instructional collaboration process. In the forward control path, the task cognitive processing complexity output by the cognition-driven progressive routing protocol is directly fed into the task complexity computation module of the dual-loop weighting model, completing a natural mapping from technical-side computational load to instructional-side role allocation. The higher the task’s cognitive load, the higher the agent tier activated by the system, with the artificial intelligence-dominated weight being correspondingly increased, thereby achieving topological alignment between computational supply intensity and human–machine responsibility boundaries and preventing, at the mechanistic level, the misalignment between technical capability and instructional demand.

In the reverse data path, the full set of interaction data generated during the instructional collaboration process is directionally fed back, based on the attribute characteristics of the current weight phase, to the corresponding agent tiers for incremental optimization. Content correction and knowledge supplementation data generated during the instructor-dominated phase are used to iteratively update the local knowledge base and retrieval index of the personal agent, enhancing end-side coverage accuracy for higher-order instructional content. Large-scale learning interaction trajectories accumulated during the artificial intelligence-dominated phase are used to update the collaborative filtering matrix and domain-specific knowledge graph associations of the collective collaborative agent, strengthening the contextual adaptability of the middle tier for tasks of medium complexity. Cross-domain creative instructional cases are deposited into the fine-tuning dataset of the cloud-based foundation model, supporting the continuous upgrade of top-tier cognitive capabilities. This bidirectionally driven iterative mode enables the system to continuously self-optimize alongside the instructional process, forming a complete operational chain encompassing computational scheduling, instructional decision-making, data feedback, and model iteration.

3 EXPERIMENTAL DESIGN AND RESULTS ANALYSIS

3.1 Experimental environment and dataset description

The experimental hardware environment covered a three-tier architecture comprising end, edge, and cloud layers. Two mainstream mobile devices were selected for the terminal tier: a mid-range model equipped with a Snapdragon 7 Gen 1 processor and 8 GB of random-access memory and a flagship model equipped with a Snapdragon 8 Gen 2 processor and 16 GB of random-access memory, in order to verify architectural compatibility across different hardware configurations. The edge nodes were deployed in a campus data center, configured with dual Intel Xeon Gold 6330 processors, 128 GB of memory, and a single A10 accelerator card, hosting the collective collaborative agent module. The cloud tier was implemented using public cloud graphics processing unit instances, deploying the general-purpose foundation model. In terms of model configuration, an open-source language model with 3 billion parameters and INT4 weight quantization was adopted for the personal agent tier; a 13-billion-parameter subject-specific fine-tuned model was deployed at the collective collaborative agent tier; and a 70-billion-parameter general-purpose large language model was deployed at the general-purpose foundation tier.

The experimental dataset comprised two parts: a local knowledge base and a test task set. The local knowledge base was constructed from core instructional materials of a university English reading and writing course, which, after chunking, yielded a total of 1,200 document chunks with an average length of 256 words per chunk. The test task set was collaboratively constructed by three subject-matter instructors holding associate professor or higher academic ranks, consisting of 180 test items. Based on Bloom's taxonomy of cognitive objectives, the items were classified into three complexity levels—low, medium, and high—with 60 items per level, corresponding respectively to the cognitive requirements of remember/understand, apply/analyze, and evaluate/create. A total of 126 students from three parallel undergraduate classes of the 2024 cohort at a university were recruited as participants for the instructional experiment. All participants had prior experience with routine mobile learning. The experimental protocol was approved by the institutional ethics review board, and informed consent was obtained from all participants.

3.2 Mobile terminal performance benchmark testing of the three-tier progressive agent architecture

This set of experiments was designed to validate the runtime performance and adaptability of the three-tier progressive agent architecture under mobile terminal resource constraints. A mid-range mobile device was selected as the test platform. Four configurations were deployed for comparative testing: a purely end-side local model, a purely cloud-based invocation mode, a traditional cloud–end binary architecture, and the proposed three-tier progressive agent architecture. Batch testing was executed for tasks at three distinct complexity levels, and core performance metrics were compared across configurations. The experimental results are presented in Figure 3.

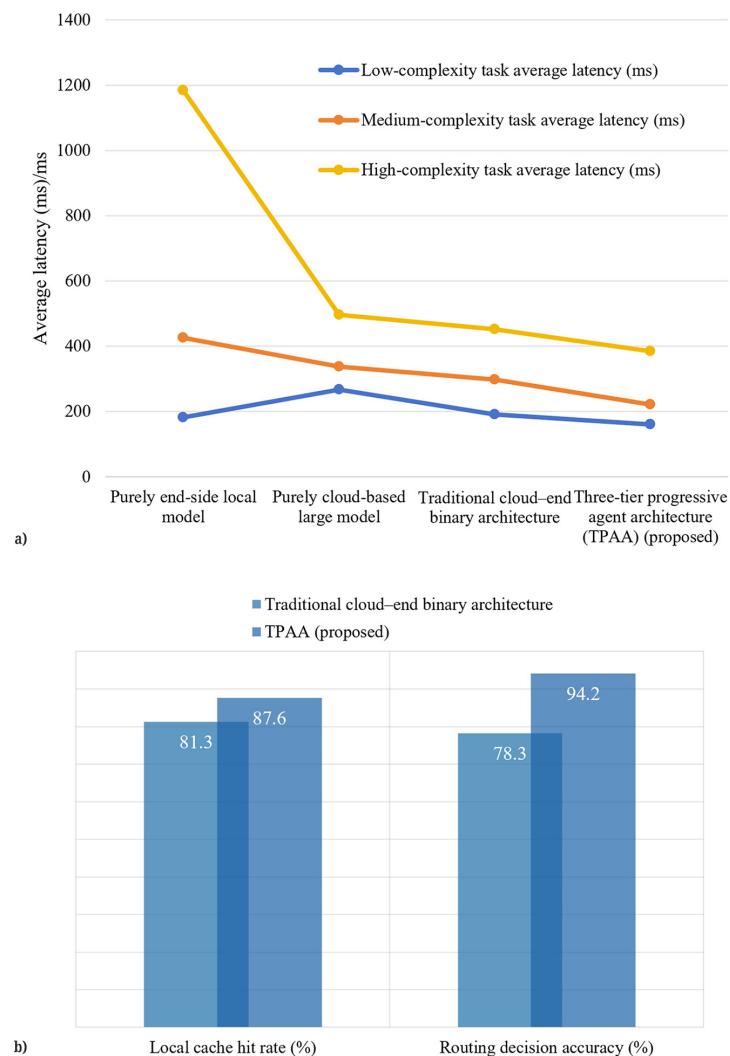


Fig. 3. Mobile terminal performance comparison across different architectures

From the test results, the three-tier progressive agent architecture maintained superior latency performance across tasks of varying complexity. For low-complexity tasks, the local cache optimization strategy resulted in retrieval and inference latency lower than that of the purely end-side full-deployment mode. For medium- and high-complexity tasks, the tiered progressive scheduling effectively prevented the sharp latency increase caused by insufficient end-side computational capacity while simultaneously reducing network transmission overhead compared to the purely

cloud-based mode, with high-complexity task latency reduced by approximately 22.5%. End-side peak memory usage was controlled within 800 MB, representing a 57.5% reduction compared to the purely end-side full-deployment mode, thereby satisfying the memory constraints of mainstream mid-range mobile devices. The local cache hit rate reached 87.6%, preserving high retrieval efficiency despite substantial memory footprint reduction, validating the effectiveness of the clustering pruning and cache gain mechanisms. The routing decision accuracy reached 94.2%, enabling optimal computational tier matching for the vast majority of tasks. Further robustness testing on lower-configuration mobile devices demonstrated that the architecture maintained stable basic functionality, with only high-complexity task latency exhibiting an acceptable increase, indicating good hardware adaptability.

3.3 Validation of the cognition-driven progressive routing protocol

This set of experiments was designed to validate the advantages of the cognition-driven progressive routing strategy in terms of scheduling rationality and comprehensive user experience. Based on the three-tier progressive agent architecture, four scheduling strategies—random routing, fixed-threshold routing, latency-driven routing, and cognition-driven progressive routing—were employed. The complete test task set was used to run the system continuously for 6 hours, and four metrics were recorded: task response accuracy, tier-switching frequency, end-side energy consumption per task, and user experience rating. The experimental results are presented in Table 1.

Table 1. Performance and user experience comparison across different routing strategies

Routing Strategy	Task Response Accuracy (%)	Tier-Switching Frequency (times per 100 rounds)	End-Side Energy Consumption per Task (mAh)	User Experience Subjective Rating (out of 5)
Random Routing	62.1	128	3.2	2.8
Fixed-Threshold Routing	79.5	76	2.7	3.5
Latency-Driven Routing	83.7	69	2.4	3.8
Cognition-Driven Progressive Routing (Proposed)	94.2	32	1.9	4.6

The results demonstrated that the task response accuracy of the cognition-driven progressive routing was significantly higher than that of the other three strategies, indicating that the cognitive-complexity-based scheduling dimension enabled more precise matching of the computational support tier required by each task, thereby preventing response quality degradation caused by insufficient computational capacity. The tier-switching frequency was only 32 times per 100 rounds, substantially lower than that of the other compared strategies, indicating that the routing mechanism effectively reduced unnecessary tier transitions and maintained the continuity and stability of the interaction process. The end-side energy consumption per task was the lowest among all strategies, benefiting from the rational computational offloading strategy, which reduced both ineffective end-side computation and battery drain caused by frequent network transmission. The user experience subjective rating reached 4.6, reflecting high learner recognition of the interaction fluency and response quality under this scheduling mode. Further sensitivity testing of the routing threshold parameters within a $\pm 15\%$ range revealed that all metric fluctuations remained

below 5%, indicating that the mechanism possessed good parameter robustness and could accommodate task characteristic variations across different subject domains.

3.4 Decision consistency validation of the dual-loop collaborative weighting model

This set of experiments was designed to validate the decision accuracy and smoothness of the dual-loop nested weighting model and to examine its adaptability in instructional scenarios. Five subject-matter instructors with over five years of frontline teaching experience were invited to independently annotate human–machine role assignments for 20 typical mobile learning scenarios covering different cognitive levels and task types. The majority annotation results were adopted as the decision benchmark. Weight computations were performed using three models—the static weighting model, the single-cognitive-loop model, and the complete dual-loop nested model—and decision consistency and fluctuation characteristics were compared. The experimental results are presented in Table 2.

Table 2. Decision effectiveness comparison across different weighting models

Weighting Model	Agreement with Expert Annotation (%)	Weight Oscillation Frequency (times per 20 rounds)	Scenario Adaptability Mean Score (out of 5)
Static Weighting Model	65.0	0	3.2
Single-Cognitive-Loop Model	82.5	11	3.9
Dual-Loop Nested Model (Proposed)	92.0	3	4.7

The experimental results demonstrated that the dual-loop nested model achieved 92.0% agreement with expert annotations, representing a 9.5 percentage point improvement over the single-cognitive-loop model. This indicates that the introduction of the outer-loop collaborative potential effectively corrected the deviations inherent in purely cognitive-dimensional decision-making, enabling comprehensive consideration of factors such as instructor capability matching, process coherence, and technical architecture stability, thereby making role allocation more consistent with actual instructional regularities. The weight oscillation frequency was only 3 times, substantially lower than that of the single-cognitive-loop model, indicating that the combined effect of the historical inertia term and the architectural stability indicator effectively suppressed high-frequency weight fluctuations, preventing disruption to the learning process caused by frequent instructional role switching and ensuring instructional continuity. The scenario adaptability score was the highest among all models, validating the model's generalizability and pedagogical appropriateness across different learning scenarios.

3.5 Quasi-Experimental study on mobile learning instructional effectiveness

This set of experiments was designed to validate the actual improvement in mobile learning effectiveness afforded by the proposed complete framework.

A quasi-experimental design was adopted, in which the 126 participants were divided into three groups by natural class, with 42 students in each group. The experimental group used the mobile learning system constructed in this study for a one-semester course; Control Group 1 used a traditional mobile learning platform equipped with a static artificial intelligence question-answering function; Control Group 2 used a purely instructor-led online tutoring mode. Standardized academic achievement tests were administered to all groups before and after the experiment. Upon completion of the experiment, learning engagement scales and learning satisfaction questionnaires were distributed. The experimental results are presented in Figure 4.

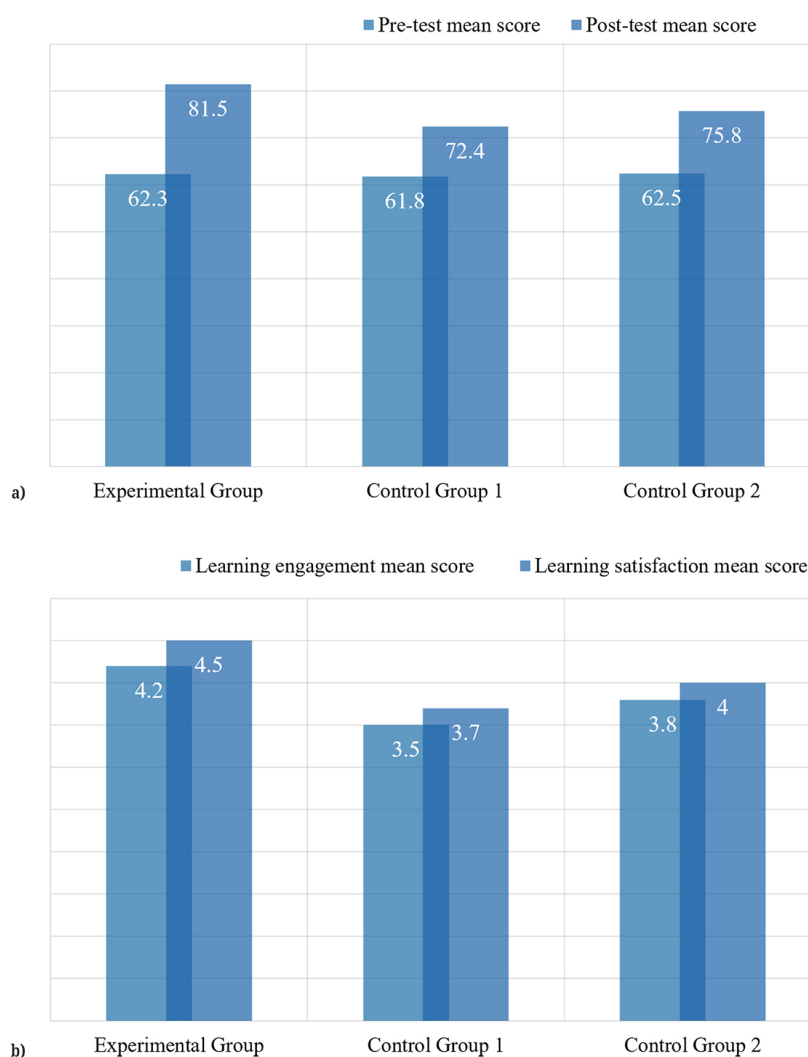


Fig. 4. Comparison of learning outcomes and subjective perceptions across groups

No statistically significant differences were observed in the pre-test scores among the three groups, indicating good between-group comparability. Post-test scores of the experimental group were significantly higher than those of both control groups, with a Hedges' g effect size of 1.24, representing a large effect magnitude and demonstrating that the proposed framework produced a highly significant improvement in learning outcomes. Learning engagement and satisfaction scores were also the highest among all groups, indicating that the dynamically adaptive human-machine collaboration mode better matched learners' individualized needs, stimulated

learning participation willingness, and enhanced the overall mobile learning experience. Compared to the purely instructor-led tutoring mode, the experimental group still maintained a clear advantage in learning outcomes, demonstrating that the immediate and personalized scaffolded support provided by the intelligent agents effectively compensated for the lack of targeting inherent in uniform tutoring modes, forming complementary advantages with instructor-led instruction.

4 CONCLUSION AND FUTURE WORK

To address the core challenges of terminal resource constraints and inadequate adaptability of human–machine instructional role assignment encountered when deploying generative artificial intelligence in mobile learning scenarios, a three-tier progressive agent computational architecture and a cognition–collaboration dual-loop nested weighting allocation model were constructed, with a bidirectionally coupled iterative closed loop formed through forward control mapping and reverse data feedback. A cognition-driven progressive routing protocol was designed, in which task cognitive complexity served as the core criterion for computational scheduling. Combined with end-side clustering pruning and cache gain optimization mechanisms, precise matching of computational resources across different tiers was achieved under the hard memory constraints of mobile terminals, effectively reducing inference latency and terminal energy consumption. In the dual-loop nested weighting model, the operational stability of the underlying technical architecture was incorporated into the collaborative potential evaluation framework. Smooth switching of human–machine instructional roles was realized through continuous weight computation incorporating a historical inertia term, thereby preventing technical fluctuations from disrupting the instructional process. Results from multiple sets of controlled experiments validated the engineering feasibility of the architecture and the instructional effectiveness of the mechanisms, confirming that the human–machine collaboration mode produced significant synergistic gains. A systematic technical pathway and empirical support are thereby provided for the construction and upgrading of mobile intelligent learning environments.

5 REFERENCES

- [1] L. Li, “Virtualized and personalized financial management education enabled by mobile interactive technologies,” *International Journal of Interactive Mobile Technologies*, vol. 20, no. 8, pp. 103–117, 2026. <https://doi.org/10.3991/ijim.v20i08.61249>
- [2] J. Liu and T. Gao, “Multimodal perception-driven optimization of mobile interaction interfaces and supply chain efficiency,” *International Journal of Interactive Mobile Technologies*, vol. 20, no. 12, pp. 140–153, 2026. <https://doi.org/10.3991/ijim.v20i12.62255>
- [3] E. Erturk and L. Dissanayaka, “Exploring the operational potential of generative artificial intelligence in local government organizations,” *Acadlore Transactions on AI and Machine Learning*, vol. 4, no. 4, pp. 263–272, 2025. <https://doi.org/10.56578/ataiml040403>
- [4] H. Crompton, A. Edmett, N. Ichaporia, and D. Burke, “AI and English language teaching: Affordances and challenges,” *British Journal of Educational Technology*, vol. 55, no. 6, pp. 2503–2529, 2024. <https://doi.org/10.1111/bjet.13460>

- [5] I. Gligorea, M. Cioca, R. Oancea, A. Gorski, H. Gorski, and P. Tudorache, “Adaptive learning using artificial intelligence in e-learning: A literature review,” *Education Sciences*, vol. 13, no. 12, p. 1216, 2023. <https://doi.org/10.3390/educsci13121216>
- [6] Y. Abadade, A. Temouden, H. Bamoumen, N. Benamar, Y. Chtouki, and A. S. Hafid, “A comprehensive survey on TinyML,” *IEEE Access*, vol. 11, pp. 96892–96922, 2023. <https://doi.org/10.1109/ACCESS.2023.3294111>
- [7] O. Friha, M. A. Ferrag, B. Kantarci, B. Cakmak, A. Ozgun, and N. Ghoualmi-Zine, “LLM-based edge intelligence: A comprehensive survey on architectures, applications, security and trustworthiness,” *IEEE Open Journal of the Communications Society*, vol. 5, pp. 5799–5856, 2024. <https://doi.org/10.1109/OJCOMS.2024.3456549>
- [8] S. Deng, H. Zhao, W. Fang, J. Yin, S. Dustdar, and A. Y. Zomaya, “Edge intelligence: The confluence of edge computing and artificial intelligence,” *IEEE Internet of Things Journal*, vol. 7, no. 8, pp. 7457–7469, 2020. <https://doi.org/10.1109/JIOT.2020.2984887>
- [9] R. Alfredo, V. Echeverria, Y. Jin, L. Yan, Z. Swiecki, D. Gašević, and R. Martinez-Maldonado, “Human-centred learning analytics and AI in education: A systematic literature review,” *Computers and Education: Artificial Intelligence*, vol. 6, p. 100215, 2024. <https://doi.org/10.1016/j.caeai.2024.100215>
- [10] X. Zheng, W. Zhang, C. Hu, L. Zhu, and C. Zhang, “Cloud-edge-end collaborative inference in mobile networks: Challenges and solutions,” *IEEE Network*, vol. 39, no. 4, pp. 90–96, 2025. <https://doi.org/10.1109/MNET.2025.3533581>
- [11] B. Zhu, K. T. Chau, and N. A. M. Mokmin, “Optimizing cognitive load and learning adaptability with adaptive microlearning for in-service personnel,” *Scientific Reports*, vol. 14, no. 1, pp. 1–18, 2024. <https://doi.org/10.1038/s41598-024-77122-1>
- [12] G. Stockwell and Y. Wang, “Framing human-AI collaboration in language education,” *Technology in Language Teaching & Learning*, vol. 7, no. 4, p. 103824, 2025. <https://doi.org/10.29140/tltl.v7n4.103824>
- [13] P. Topali, A. Ortega-Arranz, M. J. Rodríguez-Triana, E. Er, M. Khalil, and G. Akçapınar, “Designing human-centered learning analytics and artificial intelligence in education solutions: A systematic literature review,” *Behaviour & Information Technology*, vol. 44, no. 5, pp. 1071–1098, 2024. <https://doi.org/10.1080/0144929X.2024.2345295>
- [14] N. S. Raj and V. G. Renumol, “An improved adaptive learning path recommendation model driven by real-time learning analytics,” *Journal of Computers in Education*, vol. 11, no. 1, pp. 121–148, 2022. <https://doi.org/10.1007/s40692-022-00250-y>
- [15] M. Delange *et al.*, “A continual learning survey: Defying forgetting in classification tasks,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 44, no. 7, pp. 3366–3385, 2021. <https://doi.org/10.1109/TPAMI.2021.3057446>

6 AUTHORS

Chen Zhang is a master’s student at the School of Education, University of Birmingham, majoring in educational leadership and management. During his undergraduate studies, he published one research paper (E-mail: zc13811555588@126.com).

Zhongyuan Wu is currently pursuing a PhD in education at the Faculty of Education, Beijing Normal University. He serves as the dean of the Yanching Institute of Technology, School of International Education, and holds the position of associate professor. His research interests lie in higher education management and international education. He has led and participated in five provincial and municipal research projects, edited and co-edited four books, and published over 10 scientific research papers (E-mail: wuzhongyuan77@126.com).