LLM-Augmented Real-Time Assessment and Personalized Feedback in Instructional Systems

Tianlun Yang[1*], Zuyao Wang[1], Anhai Yao[2], Georgios Kapogiannis[3], Byung-Gyoo Kang[4]

[1]NingboTech University, Ningbo, ZJ 315100, China
[2]Zhejiang Sci-Tech University, Hangzhou, ZJ 310018, China
[3]University of Warwick, Coventry CV4 7AL, UK
[4]University of Nottingham Ningbo China, Ningbo, ZJ 315100, China

*tianlun.yang@nbt.edu.cn

Abstract. This study confronts the latency and generic nature of conventional pedagogical feedback by architecting and deploying an LLM-driven intelligent instructional system. Following a literature review of extant scholarship, this study first delineated the technical affordances that enable deep integration of large language models with educational contexts. This study then engineered a production-ready B/S architecture in which a lightweight front end orchestrates learner interaction while a scalable back end securely queries the model's official API to perform real-time semantic parsing and adaptive feedback synthesis. The system's kernel is a personalized, real-time feedback loop that continuously calibrates instructional scaffolds to individual cognitive profiles. Empirical assessment within authentic classroom settings demonstrates the system's instructional efficacy.

Keywords: LLM, Real-Time Feedback, Intelligent Instructional System.

1 Introduction

Education underpins societal continuity and civilizational transmission; however, conventional instruction, which centered on instructor exposition and delayed manual grading, deprives learners of prompt, precise feedback, thereby weakening retention and overall efficacy. Although successive instructional innovations and information systems have been introduced, their contextual misalignment with contemporary higher-education realities perpetuates a research gap that demands a fundamentally new approach.

The ongoing intelligence era has witnessed the accelerated commercialization of large-scale artificial-intelligence models whose versatile capabilities promise transformative impact across domains. Within education, the tight integration of such models with information architectures offers the prospect of real-time, data-driven responses to individual learners, yielding measurable gains in academic quality. This study first chart the shortcomings of current pedagogy, then fuse informatics and AI

A. T. Patanasorn et al. (eds.), *Proceedings of the 3rd International Conference on Educational Development and Social Sciences (EDSS 2026)*, Advances in Social Science, Education and Humanities Research 1010,
https://doi.org/10.2991/978-2-38476-569-0_10

into a prototype smart-teaching system, deploy it in class, and offer a replicable roadmap for future work.

2 Literature Reviews

2.1 Impact from Real-Time Feedback to Learning Performance

Real-time feedback exogenously delivers precise correctness within milliseconds, offloading metacognitive monitoring and error detection from working memory and thus reducing overall cognitive load [1]. By interrupting the consolidation of incorrect representations, the signal prevents their transfer to long-term memory, suppresses automatized maladaptive schemata, and reallocates attention to the encoding and restructuring of accurate rules, thereby optimizing learning outcomes [2].

The contingency between response and outcome, made salient by instantaneous confirmation, triggers dopaminergic reward circuits, generates immediate certainty and competence, and strengthens the behavior-outcome association [3]. This positive affect elevates intrinsic motivation, increases practice frequency and task persistence [6], and initiates a self-reinforcing cycle of performance, feedback, and re-engagement that continuously elevates learning investment and achievement [4].

By embedding error identification, strategic adjustment, and re-practice within a single temporal episode, real-time feedback compresses the error-awareness-correction loop into a high-frequency, micro-step cycle [5]. The resultant trajectory remains anchored in the zone of proximal development, minimizes redundant exploration costs, and yields maximal performance gains per unit of time, thereby exponentially increasing practice efficiency [6].

2.2 Impact from Information System to Real-Time Feedback

The platform continuously ingests semantically tagged telemetry generated by embedded scripts and lightweight probes, capturing every login, click, response, and resource switch at millisecond granularity [7]. These raw events are streamed through a stateless processor that cleanses, temporally aligns, and vectorizes them before committing the resulting multidimensional vectors to a learning data warehouse [8]. The pipeline materializes an immutable, real-time evidentiary chain that operationalizes instantaneous diagnosis and feedback without additional instrumentation [9].

Concurrent advances in silicon throughput, algorithmic efficiency, and low-latency networking converge into a unified computational substrate [10]. Learning vectors migrate transparently between edge and cloud nodes, where GPU-accelerated models execute inference within a single Round-Trip Time (RTT), enabling decision-level interventions that improve both precision and latency by an order of magnitude over prior architectures [11].

Standardized endpoints expose recommendation, diagnosis, and interaction control as stateless micro-services [12]. Orchestrators compose these services into situational workflows that adapt to lecture, lab, or assessment contexts without code changes,

converting personalized immediate feedback from a custom engineering task into a declarative, portable, and horizontally scalable configuration artifact [13].

2.3 Impact from LLM to Information System

Amid proliferating digital sediment, corpora once considered peripheral, which is long-form texts and fragmentary utterances, are reclassified as organizational assets because they stream directly into large-scale pre-trained models [14]. Business requirements are now satisfied through natural-language descriptions that replace manual feature engineering and rule scripting, shrinking knowledge-injection lead times to hours [15]. As model capabilities surface behind standardized endpoints, infrastructural gravity migrates from relational databases and middleware to elastically scaled model services, establishing a new substrate for enterprise computation [16].

Natural language has become the default transport layer; users state intent conversationally and bypass multilevel graphical menus [17]. Functions are decomposed into reusable prompt-plugin micro-units that hot-plug at runtime, enabling on-the-fly recombination for volatile scenarios [18]. A unified gateway intercepts every call to enforce caching, rate-limiting, and security policies, delivering observable, governable guarantees without sacrificing latency [19].

Version control now spans three artifact classes, which are model weights, prompts, and plugins, requires a single source of truth to prevent skew [20]. Release gates augment latency Service-Level Agreements with credibility and factual-accuracy thresholds, expanding quality dimensions from speed to trust [21]. The system thereby exhibits a living phenotype of continuous learning and dynamic update; governance frameworks must internalize its probabilistic ontology and growth potential to remain coherent as the knowledge body drifts [22].

3 Research Design

The question of this research is how can the quality and efficiency of instruction be improved during the teaching process? The aim of this research is by applying an intelligent system that integrates a large language model to establish a real-time feedback mechanism and introduce personalized exercise methods to enhance students' learning motivation and ultimately improve the quality and efficiency of instruction.

The objective of this research are: a) to investigate the problems and limitations inherent in traditional instructional methods and processes; b) to explore the advantages of relevant artificial intelligence technologies in instruction within the intelligent era; c) to assess the deployment and application of large language models within information systems to improve instructional quality and efficiency.

This study adopts a mixed-methods approach. For research objective a and b, a literature review is conducted to analyze the problems and limitations of traditional instructional methods and processes. For research objective c, a design intervention is employed to develop an intelligent instructional system, and the application of this system is evaluated to assess its potential role and value.

4 Design Intervention

The system adheres to a Browser/Server (B/S) architecture. HTML and CSS render structure and style, while JavaScript governs interaction. Flask handles back-end logic and page coordination, MySQL stores and serves data, and Python mediates all database operations (see Fig. 1). Kimi, developed by Beijing Moonshot AI, provides the analytic engine; connectivity to this large language model is established through the official Kimi API.

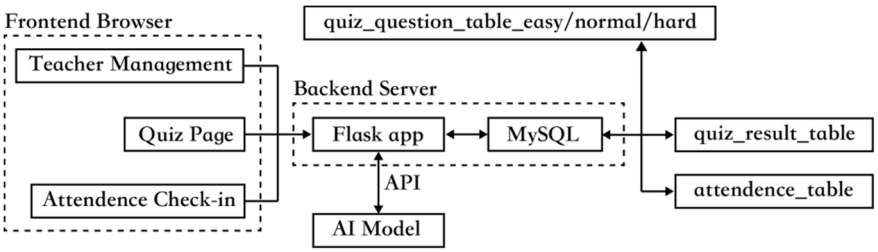

Fig. 1. Architecture of the Intelligent Instructional System

Three table categories populate the database: attendance records, individual quiz attempt records, and a bank of items stratified by difficulty. Students land on the homepage, click function buttons, and navigate to the corresponding interface (see Fig. 5). The prototype currently exposes only attendance and quiz modules; the instructor portal, also browser based, manages both.

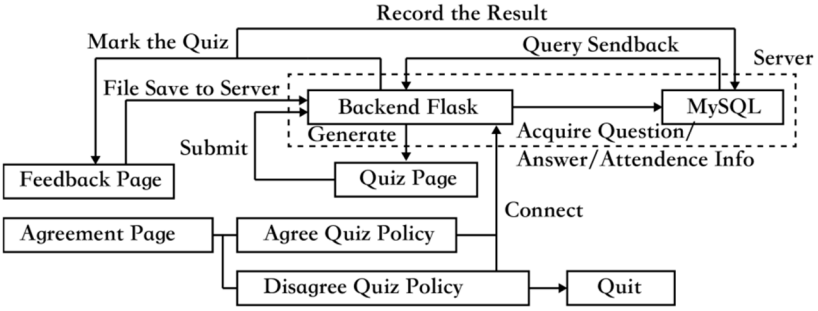

Fig. 2. Quiz-Taking Flowchart

The kernel is an real-time feedback loop for quiz outcomes (see Fig. 2). Part one automates grading via conventional information processing. All current items are multiple choice to guarantee precision. The server extracts returned answers, retrieves the keyed response from MySQL, and compares the two strings (see Code 1). Matches accrue points to a running total.

Code 1 Auto-Grading Core

```
data = request.get_json()
answer1 = data["question1"]
```

```
ques_ans = {}
conn, cursor = loginSQL_dict()
for question_num in ques_num:
cursor.execute(
f"SELECT * FROM quiz_questions WHERE id = %s LIMIT 1",
        (question_num,)
    )
  row = cursor.fetchone()
  if row:
    ques_ans[f"question_{question_num}"] = row['Answer']
  conn.commit()
cursor.close()
conn.close()

SCORE = 0
if answer1 == ques_ans[f'question_{ques_num[0]}']:
  SCORE = SCORE + score_of_this_question
```

Upon completion, the server merges keyed and student answers into a single HTML page returned instantly. Learners review, summarize, and download this file for later reference; the same page is archived for subsequent LLM analysis (see Code 2).

Code 2 Quiz-Result Push-Back Logic

```
<a href="/{file_name}">Download the Report</a>
file_path = os.path.join(f"result_{quiz_num}_quiz", file_name)
with open(file_path, 'w', encoding='utf-8') as f:
  f.write(html2)

@app.route('/<filename>')
def download_result(filename):
  folder = os.path.join(app.root_path, f"result_{quiz_num}_quiz")
  try:
    return send_from_directory(folder, filename, as_attachment=True)
  except FileNotFoundError:
    abort(404)
```

Because mastery varies, item difficulty must adapt. After demographic entry the server queries prior quiz scores from MySQL, estimates current ability, and selects an appropriate difficulty tier (see Fig. 3).

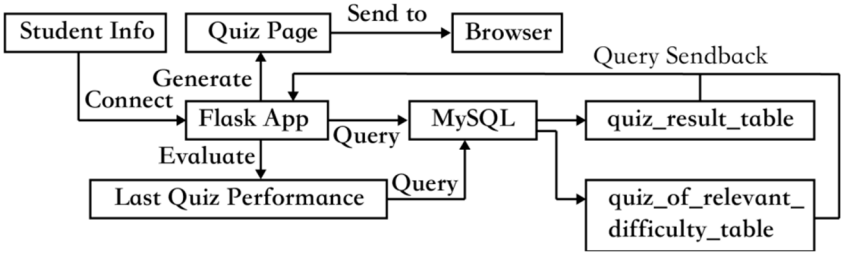

Fig. 3. Quiz-Question Generation Pipeline

The prototype uses the most recent score as the sole selector (see Code 3). Relevant items are retrieved, rendered into HTML, and delivered for attempt.

Code 3 Difficulty-Picker Logic

```
difficulty = "normal"
conn, cursor = loginSQL()
cursor.execute(
    f"SELECT * FROM class_quiz_result WHERE ID = %s",
    (student_ID,)
)
row = cursor.fetchone()
last_score_col = f"{last_quiz}_quiz_Score"
last_score = row[last_score_col]  # to know the score of last quiz
if last_score < 60:
    difficulty = "easy"
if last_score >= 90:
    difficulty = "hard"
```

The returned markdown report is written to file and presented to the student, yielding granular diagnostic insight and enabling timely strategic adjustment; instructors gain comparable visibility for pedagogical calibration (see Fig. 4).

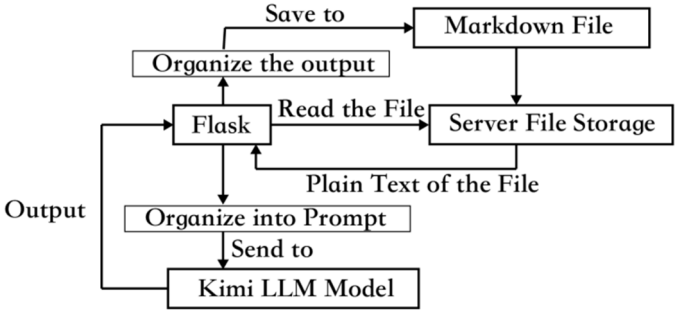

Fig. 4. LLM-Driven Quiz Evaluation Integration

Part two injects the large language model. The HTML answer sheet is read, embedded into a formatted prompt (see Code 4), and submitted to the Kimi K2 endpoint.

Code 4 LLM Gateway Connection Logic

```
client = OpenAI(
api_key="api_key",
base_url="https://api.moonshot.cn/v1",
)
file_object = client.files.create(file=Path(f"./result_{quiz_num}_quiz/{file}"), pur-
pose="file-extract")
file_content = client.files.content(file_id=file_object.id).text
message = [{"role": "system",
            "content": f"You serve as a teaching assistant tasked with analyzing the stu-
dent's examination script, which is provided as {file_content}. Produce feedback valuable
to both the student and the instructor, structured under the following headings:##1. Overall
Performance Evaluation ## 2. Areas Mastered by the Student ## 3. Areas Requiring
```

```
Improvement ## 4. Learning Recommendations for the Student. Each section must contain plain text only; tables, bullet points, code blocks, extra line breaks, introductory phrases, and concluding remarks are prohibited.
"}]
completion = client.chat.completions.create(
model="kimi-k2-0905-preview",
messages=message,
temperature=0.6,
    )
result = completion.choices[0].message.content  with open(f"./{quiz_num}_quiz_result_analysis/{file}_quiz_{quiz_num}_result_analysis.md", "w", encoding="utf-8") as f:
    f.write(result)
```

The interface is styled to signal youthful energy (see Fig. 5). Pure HTML and CSS suffice; students open any browser, enter the URL, and operate the system without additional software. Because no sensitive data are handled, plain-text transmission is retained for efficiency; SSL is not deployed.

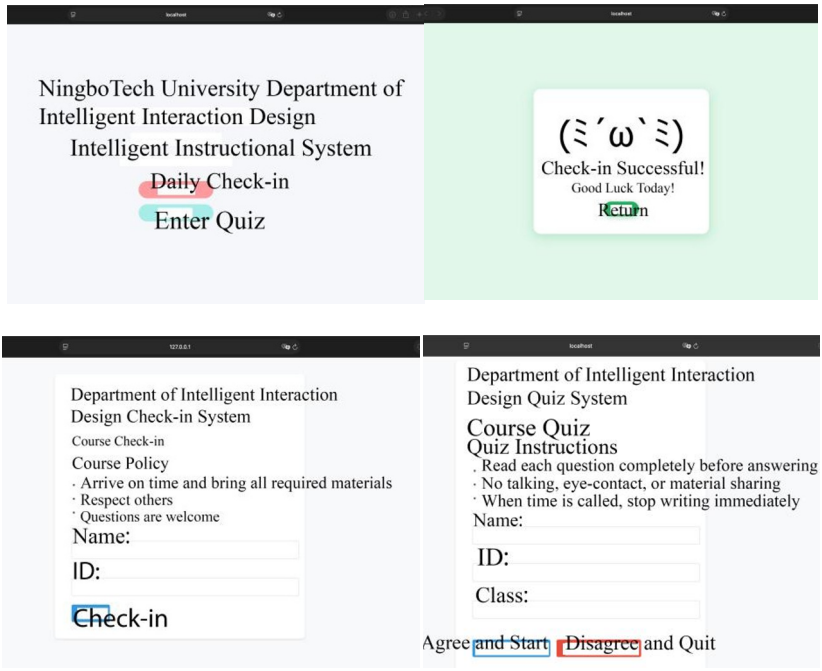

Fig. 5. System UI Layout Specification

The system prototype has been tested in real classroom settings throughout the semester. As shown in (see Fig. 6), students used laptops and accessed the quiz via the internal network through their browsers during class.

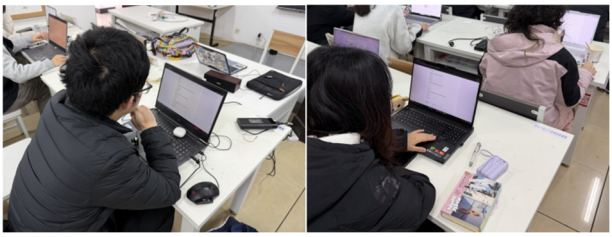

Fig. 6. In-Class Live-Quiz Session Monitor

5 Discussion

The literature review delineates the pedagogical role of real-time feedback and identifies the intelligent information system as its enabling vehicle: the system returns instantaneous quiz diagnoses that reveal residual knowledge gaps. The kernel of the proposed system is the LLM-driven real-time feedback mechanism (see Fig. 7).

Once the front end relays quiz responses to the back-end server, automatic scoring is executed and the LLM is invoked via API to generate an evaluation report; students iteratively exploit this loop to elevate learning quality. The UI design additionally conveys youthful vitality, fostering student affinity.

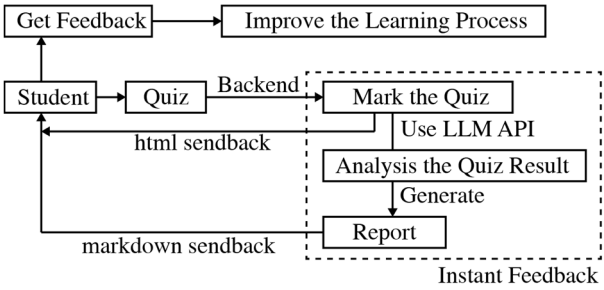

Fig. 7. Impact to Student Learning Process

Relative to traditional instruction, the following Table 1 gives measurable advantages of the LLM-Infused system. Attendance registration shifts from error-prone verbal roll call to a web-based self-service interface that is both rapid and accurate. Quiz administration migrates from paper scripts and manual grading to automated scoring and LLM-authored evaluation reports, yielding superior speed and precision. Archival storage transitions from vulnerable paper records to a digital repository that supports instantaneous, exact retrieval through search.

Table 1. Edge of the LLM-Infused System

Traditional Teaching	LLM-Integrated Information System	Advantages
Attendance taking	Web-based self-check-in	Convenience and Accuracy

Paper quiz	Online quiz	High accessibility & efficiency
Manual grading	Auto-grading by LLM	Efficiency & consistency
Handwritten feedback	LLM-generated feedback	Timeliness & precision
Physical archiving	Server-side digital archive	Secure long-term preservation

6 Conclusion

The proposed method embeds a large language model within an information system to operationalize instantaneous feedback during instruction. Students receive real-time feedback that enable them to re-calibrate their learning trajectories without delay; instructors gain granular visibility into individual progress, thereby increasing both instructional efficacy and operational efficiency. Consequently, the approach is positioned for scalable deployment in contemporary higher education and furnishes a conceptual benchmark for future pedagogical advancement.

7 Limitation and Future Research

The limitation of this study is that the system's current status is a prototype under validation, whose learning affordances are restricted to in-class quizzes and their analytics; furthermore, deployment has occurred in only one course within a single class, leaving the remainder of the program untouched.

Future research will systematically expand the learning repertoire by incorporating note-taking and post-class assignments, employ LLM technology to track learner states and deliver real-time feedback, and extend implementation to every course in the program, thereby establishing empirical evidence of the system's contribution to instructional quality and efficiency.

Acknowledgments

This study was funded by 2024 General Project of Ministry of Education Foundation on Humanities and Social Sciences: "Ethic Research of Generative AI Design" (Project Number: 24YJA760103).

References

1. Hattie, J., Timperley, H.: The power of feedback. Rev. Educ. Res. 77(1), 81-112 (2007).
2. Liu, S., Yu, G.: L2 learners' engagement with automated feedback: An eye-tracking study. Language Learning & Technology 26(2), 78-105 (2022).

3. Butler, A.C., Karpicke, J.D., Roediger, H.L.: Correcting a metacognitive error: Feedback increases retention of low-confidence correct responses. J. Exp. Psychol. Learn. Mem. Cogn. 34(4), 918-928 (2008).
4. Azevedo, R.: Why is externally-facilitated regulated learning more effective than self-regulated learning with hypermedia? Educational Technology Research and Development 56(1), 45-72 (2008).
5. Attali, Y.: Immediate feedback and opportunity to revise answers to open-ended questions. Educational and Psychological Measurement 70(1), 22-35 (2010).
6. Epstein, M.L.: Immediate feedback assessment technique promotes learning and corrects inaccurate first responses. The Psychological Record 52(2), 187-201 (2002).
7. Lorido-Botran, T.: A Review of Auto-scaling techniques for elastic applications in cloud environments. Journal of Grid Computing 12(1), 55-82 (2014).
8. Vajda, D.L.: Machine learning-based real-time anomaly detection using data pre-processing in the telemetry of server farms. Scientific Reports 14(1), 23288 (2024).
9. Meng, F.: Integrating sensor embeddings with variant transformer graph networks for enhanced anomaly detection in multi-source data. Mathematics 12(17), 2612 (2024).
10. Venkatesan, A.: The AI-driven future of real-time telemetry analytics in computer networks. Journal of Artificial Intelligence & Cloud Computing 3(1), 3-1 (2024).
11. Wang, Z.: GPU-accelerated machine learning inference as a service for computing in neutrino experiments. Computers & Geosciences 149, 104-115 (2021).
12. Tsakanikas, V.: An intelligent model for supporting edge migration for virtual function chains in next generation internet of things. Sci Rep 13(1), 1063 (2023).
13. Taleb, T.: Real-time service migration in edge networks: a survey. Journal of Sensor and Actuator Networks 14(4), 79-95 (2025).
14. Almeida, J.J.: The Per-Fide corpus: a new resource for corpus-based terminology, contrastive linguistics and translation studies. Working with Portuguese Corpora 1(1), 177-200 (2014).
15. Tatipamula, S.: The evolution of AI workflow automation: From rules to adaptive intelligence. International Journal on Science & Technology 16(2), 31-42 (2025) .
16. Ochuba, N.A.: Systematic Review of API Gateway Patterns for Scalable and Secure Application Architecture. Journal of Frontiers in Multidisciplinary Research 2(5), 99-110 (2021).
17. Shah, A.V.: Serverless architectures: Implications for distributed system design and implementation. International Journal of Science and Research 13(12), 1250-1253 (2024).
18. Alharbi, S.J., Moulahi, T.: API Security Testing: The Challenges of Security Testing for Restful APIs. International Journal of Innovative Science and Research Technology 8(5), 1485-1499 (2023).
19. J. Zhao, M. Yang, Y. Zhao, X. Hu, W. Zhou, H. Li: MCMARL: Parameterizing Value Function via Mixture of Categorical Distributions for Multi-Agent Reinforcement Learning. IEEE Trans. Games 16(3), 556-565 (2024).
20. Wisniewski, B., Zierer, K., Hattie, J.: The power of feedback revisited: A meta-analysis of educational feedback research. Front. Psychol. 10, 3087 (2019).
21. Schlag, R.: The role of feedback type and peer interaction on knowledge acquisition in a flipped classroom on social science research methods. Eur. Educ. Res. 7(2), 21-40 (2024).
22. Yulhendri, Hakim, L.: The influence of feedback learning on student engagement and student performance. Educ. Process Int. J. 17, e2025318 (2025).