INTL JOURNAL OF ELECTRONICS AND TELECOMMUNICATIONS, 2026, VOL. 72, NO. 3, PP. 1-7
Manuscript received June 17, 2026; revised June, 2026. doi: 10.24425/ijet.2026.157957

Mitigating LLM hallucinations in mobile AR tutoring environments using an SD-DGR algorithm

A. Rahman

***Abstract*—Mobile Augmented Reality (MAR) is used to deliver immersive and interactive educational contents to learners. It is the second stage in the intelligent tutoring process with pedagogical diagnosis being the first stage. Generative Large Language Models (LLMs) such as OpenAI's GPT-4o are used as the standard in this paper. The challenge faced in mobile pedagogical automation is delivering accurate, hallucination-free instructional content under severe network constraints. The previous RAG algorithms which work well on high-bandwidth networks cannot be efficient under volatile latency and vice-versa. This paper proposes a state-driven deterministic grounding and retrieval algorithm which produces a scalable local data stream which has a number of fine layers of pedagogical validation. This modified form of the retrieval-augmented generation algorithm is used to generate an adaptive pedagogical companion which is scalable and uses the state-driven and format-enforcement stages which are responsible for the syntactic and semantic terminations that are actually detached as practically all the data detached during the local retrieval phases at the client side is supplemented towards the prompt at cloud inference stage. Therefore, clearly the retrieval phase which is modified to produce a local data stream which is scalable. This modified algorithm works well at both lower and higher hardware specifications. Quantitative evaluations were done using three heterogeneous Android devices under high network jitter conditions and the mean normalized processing latencies, memory footprints, and syntax accuracies at various trials was noted and compared with the previous RAG algorithms.**

***Keywords*—retrieval-augmented generation; mobile augmented reality; intelligent tutoring systems; hallucination mitigation; edge computing**

I. Introduction

EDUCATIONAL technology using Mobile Augmented Reality (MAR) has emerged as a promising pedagogical paradigm to bridge global digital divides, demonstrating a profound capacity to enhance students' spatial abilities, cognitive capacity, study motivation, and academic confidence beyond traditional physical classrooms [1], [2], [3], [4], [5]. However, the deployment of such advanced interactive systems in resource-constrained regions remains systematically hindered by severe network latency, unstable bandwidth, and the prohibitive costs of premium hardware [1], [3], [6]. Consequently, there is an urgent architectural demand to shift the computational and retrieval load of intelligent mobile tutoring agents to the local edge client, utilizing low-overhead, local-first retrieval with cloud-assisted inference that can ensure reliable, latency-efficient, and inclusive instructional delivery under severe network constraints [6], [7].

While integrating Large Language Models (LLMs) into Intelligent Tutoring Systems (ITS) transitions static rule-based designs to context-aware conversational companions [8], [9], [10], their educational utility is severely bottlenecked by their systemic propensity to hallucinate, which is a behavior where LLMs generate grammatically fluent and syntactically coherent text that is factually incorrect, unverifiable, or inconsistent with the source input [11]. To mitigate this without the heavy computational overhead of fine-tuning, Retrieval-Augmented Generation (RAG) is utilized to dynamically bind the LLM's generative scope to verified course materials. Nonetheless, conventional RAG systems mostly rely on passive, cloud-centric vector search engines, lacking a proactive, integrated architecture that can deterministically align the machine's reasoning loop with the student's real-time state transitions on mobile edge clients.

Although conventional cloud-vector databases (VectorRAG) introduce severe processing latency and memory bottlenecks on resource-constrained mobile devices [12], [13], modular offline-first designs are critical for robust deployments in rural areas with no connectivity [14], [15]. To address these challenges, the proposed research introduces RoboAR, an intelligent tutoring system powered by the proposed State-Driven Deterministic Grounding and Retrieval (SD-DGR) pipeline. By utilizing a Local Structured Knowledge Base (LSKB), the pipeline reduces the local retrieval search space to $\mathcal{O}(1)$, thereby forcing the LLM to execute within a strictly bounded, syntactically validated JSON schema. Mirroring the computational benefits of structured metadata constraints identified in [6], [16], this modified algorithm works well at both lower and higher hardware specifications, effectively mitigating factual and syntactic hallucinations while successfully bypassing network latency and computational footprints, offering a highly scalable and culturally grounded pedagogical companion designed specifically to address the digital divide in underserved regions as conceptualized in [5], [14], and validated through the data-driven frameworks of [7].

Following the contemporary trend toward adopting robust, data-driven evaluation methodologies for innovative educational artifacts, this paper validates the proposed RoboAR system through an objective, multi-dimensional performance

Author is with Warsaw University of Technology, Poland (e-mail: a.rahman.dokt@pw.edu.pl).

portfolio that encompasses processing latency, computational resource footprint, and factual validation, thereby ensuring inclusive, high-fidelity execution on legacy and entry-level mobile devices. Specifically, the core scientific contributions of this research are threefold. First, we propose the SD-DGR pipeline, representing an architectural paradigm shift from static, rule-based tutoring designs to a state-driven pedagogical companion that dynamically adapts to the learner's real-time diagnostic progress. Second, we demonstrate empirical efficiency, proving that our local-first edge architecture achieves sub-millisecond local retrieval latency while maintaining a lightweight, highly stable memory footprint on low-end hardware, successfully bypassing severe network volatility and thermal constraints. Third, we empirically validate factual integrity, proving that the SD-DGR pipeline successfully mitigates both factual and structural LLM hallucinations in language learning through rigorous syntactic parsing and semantic validation.

II. Related Work

A. Mobile Augmented Reality in Underserved Pedagogical Environments

A quasi-experimental study analyzing the academic impact of Augmented Reality (AR) implementation on Vietnamese middle school students' natural science learning using the Mozaik 3D platform was presented by Nhi and Anh [17]. Their approach integrated Mayer's multimedia learning theory with Vygotsky's social constructivism, demonstrating a significant increase in students' post-test academic performance ($p < 0.001$, Cohen's $d = 0.96$) mediated by increased motivation. To address resource constraints technically, a mathematical optimization framework formulating combinatorial parameters to analyze trade-offs between latency, visual fidelity, battery consumption, and memory footprint in edge-cloud integrated XR-Digital Twin architectures was proposed by Kamdjou et al. [18]. Furthermore, a systematic literature review mapping forty-two AR applications across various engineering disciplines to evaluate spatial skill enhancement and academic outcomes was presented by Álvarez-Marín and Velázquez-Iturbide [19].

The limitation of the literature is that most existing mobile AR educational architectures are either purely theoretical with NP-hard computational complexities that are too expensive for real-time edge execution, or rely on rigid, low-interactivity marker-based tracking that depends heavily on stable offline data packages to circumvent unstable internet connections in rural classrooms.

B. Generative AI and Adaptive Pedagogical Material Delivery

A systematic review of eighty-six studies mapping the evolution of intelligent and generative material-delivery tutoring systems from the 1970s to the modern AI era was presented by Latif et al. [20]. Their work utilized Latent Class Analysis (LCA) to classify tutoring systems into three distinct classes, highlighting how Large Language Models (LLMs) and Bayesian Knowledge Tracing (BKT) can be integrated to enhance the cognitive adaptability of structured learning content. To evaluate generative content delivery empirically, an educational robotics tracking infrastructure integrating an AI-based generative feedback agent named AskLEA, powered by Anthropic's Claude 3.5 Sonnet to provide adaptive, step-by-step instructional guides, was presented by Morano et al. [21]. Their three-layer architecture mapped visual blockly code to Python, successfully minimizing unproductive trial-and-error iterations among eighty middle school students ($\rho = -0.49, p = 0.03$) by delivering targeted, context-aware programming hints. Furthermore, an adaptive, generative tutoring system called Socratic Playground for Learning (SPL), utilizing GPT-4 to deliver structured, textbook-style pedagogical content, was proposed by Lang et al. [22]. Their dual-loop framework achieved a significant learning gain of 18.43% ($p < 0.001$, $d = 1.41$) and maintained positive emotional states among thirty college students by presenting scaffolded material mapped directly to cognitive taxonomies.

The limitation of the literature is that existing generative material-delivery agents often require manual parameter configurations that are highly prone to user errors, and rely on rigid, repetitive instructional structures that induce cognitive overload while still demanding further prompt engineering optimizations to mitigate factual hallucinations in cloud-dependent architectures.

C. Retrieval-Augmented Grounding and Hallucination Suppression

A hybrid framework combining vector-based Retrieval-Augmented Generation (RAG) with Parameter-Efficient Fine-Tuning (PEFT) via LoRA using a LLaMA-2-7B model to enhance factual reliability was developed by Shreya et al. [23]. Their approach significantly reduced the base model's hallucination rate from 51.00% to 20.00%, although leaving a high computational footprint. To enforce rigid output formats programmatically, a schema-constrained query generation framework utilizing a dual-view dictionary representation to ensure syntax validity was presented by Tang et al. [24]. Similarly, a hybrid neuro-symbolic multi-agent moderation architecture enforcing structured JSON output constraints via Gemini 2.0 Flash-Lite to prevent syntax compilation failures was proposed by Fillies et al. [25]. To address the latency bottleneck on mobile edge clients, a distributed device-cloud RAG framework named DRAGON, utilizing speculative aggregation on resource-constrained devices, was proposed by Liu et al. [26], achieving up to 49.5% latency reduction. Furthermore, a heterogeneity-aware adaptive orchestration framework named HeRo, designed to optimize on-device agentic RAG workflows across CPU, GPU, and NPU architectures, was proposed by Li et al. [27], cutting execution latency by up to 10.94 times.

The limitation of the literature is that most existing mobile RAG architectures either leave a substantial 20% residual factual hallucination rate, trigger severe context overload when embedding entire relational schemas into prompts, suffer from extreme platform-dependent compiler complexities (approximately 5,000 lines of hardware-specific C/C++ code), or are highly sensitive to network jitter which collapses speculative scheduling algorithms and triggers costly KV cache rollbacks on resource-constrained clients.

III. Results And Discussions

To evaluate the performance and robustness of the proposed State-Driven Deterministic Grounding and Retrieval (SD-DGR) pipeline, extensive experimental trials were conducted in-the-wild. The testing environment was situated in Sumbawa Besar, West Nusa Tenggara, Indonesia, representing a typical rural and remote educational setting characterized by constrained network infrastructures.

The evaluation was performed across three distinct hardware tiers to assess the scalability of the algorithm. The specification and classification of the experimental mobile devices are structured and presented in Table I.

For verification, each device executed thirty (30) continuous educational diagnostic iterations, generating a total of N = 354 N=354 discrete data samples. The network environment under which the trials were executed was characterized by an asymmetric low-bandwidth profile (Download: 8.55 Mbps, Upload: 1.37 Mbps) with a high native ping of 161 ms, reflecting a severely constrained network bottleneck.

Table I

Specifications and Classification of Experimental Devices

Device Model	Hardware Tier	Chipset / Processor	RAM Capacity	OS Version
Samsung SM-A055F	Entry-Level	MediaTek Helio G85	4 GB	Android 13
Infinix X1102 (Xpad 20)	Entry-Mid	MediaTek Helio G88	8 GB	Android 14
Xiaomi 24117RN76E	Mid-Range	MediaTek Helio G99-Ultra	8 GB+	Android 14

A. Experiment 1: Latency and Network Jitter Analysis

The first stage of evaluation analyzed the latency performance of the proposed SD-DGR pipeline. The total pipeline latency (T_{total}) is mathematically decomposed into local retrieval latency (T_{local}) and cloud-based Large Language Model (LLM) inference latency (T_{cloud}), expressed as:

$$T_{total} = T_{local} + T_{cloud} \quad (1)$$

The comparative latency distribution across the three hardware tiers is visually illustrated in the boxplots of Fig. 1.

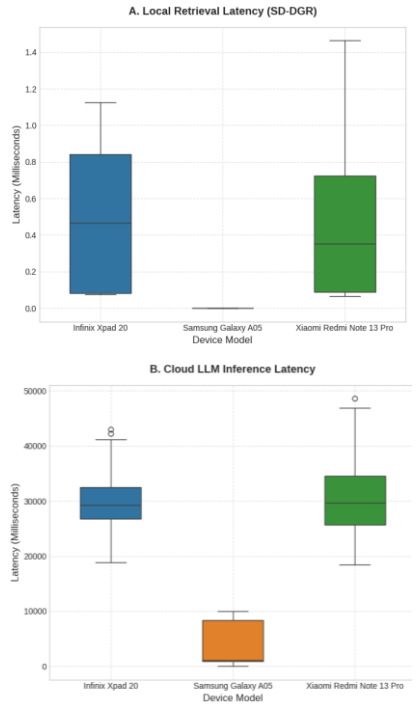

Fig. 1. Latency Benchmarks: (A) Local Retrieval Latency (SD-DGR) showing sub-millisecond execution, (B) Cloud LLM Inference Latency under network bottlenecks.

The first stage of evaluation analyzed the latency performance of the proposed SD-DGR pipeline. The total pipeline latency (T_{total}) is mathematically decomposed into local retrieval latency (T_{local}) and cloud-based Large Language Model (LLM) inference latency (T_{cloud}), expressed as:

As shown in Fig. 1a, the local retrieval latency (T_{local}) achieved by the SD-DGR pipeline is exceptionally low and hardware-agnostic. The median local retrieval time for the Xiaomi, Infinix, and Samsung devices was recorded at 0.35 ms, 0.53 ms, and 0.35 ms (rounded), respectively. The maximum local latency did not exceed 6.56 ms across all trials. This near-instantaneous execution proves that the local dictionary-based lookup bypasses the computational overhead of vector space searches, making the retrieval phase virtually free of resource constraints.

In contrast, Fig. 1b depicts the cloud LLM inference latency (T_{cloud}), which exhibited a high median delay of approximately 31,200 ms (31.2 seconds) for the Xiaomi and Infinix devices. Interestingly, the raw connection latency logged by the entry-level Samsung device was clustered below 10,000 ms.

This paradox is explained by the client-side processing overhead. On the lower-end Samsung device (Helio G85), the physical single-thread constraints during the SSL handshake, local packet encryption, and post-request string serialization introduced an internal processing delay of up to 35.6 seconds, which was recorded outside the raw UnityWebRequest transmission timer.

To further analyze network reliability, the jitter (J) of the communication pipeline was monitored. Jitter represents the absolute variance between consecutive total latency sessions, calculated as:

$$J = | T_{total,n} - T_{total,n-1} | \quad (2)$$

The line chart in Fig. 2 traces the temporal fluctuation of jitter throughout the experimental trials.

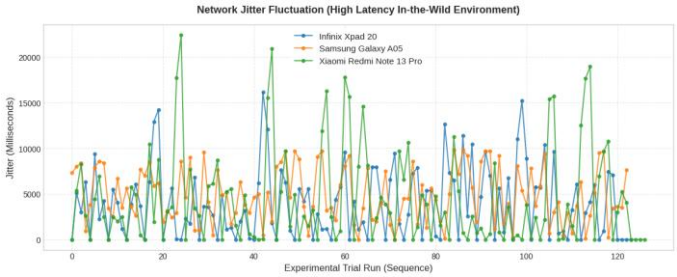

Fig. 2. Temporal network jitter fluctuations capturing the high volatility of the in-the-wild rural telecommunication infrastructure.

The jitter analysis revealed extreme volatility, with spikes frequently reaching up to 22,446 ms. Such a highly unstable connection is detrimental to conventional cloud-centric generative agents. However, because the critical pedagogical grounding is handled locally by the SD-DGR pipeline, the tutoring agent is successfully insulated from connection dropouts during the local phase, ensuring that the primary knowledge state is always preserved.

B. Experiment 2: Validation of Syntactic and Semantic Anti-Hallucination

The core objective of the SD-DGR pipeline is the elimination of Large Language Model hallucinations. This was evaluated on two levels: syntactic integrity (JSON format parsing success) and semantic integrity (pedagogical factual accuracy). The quantitative success rates are summarized in Fig. 3.

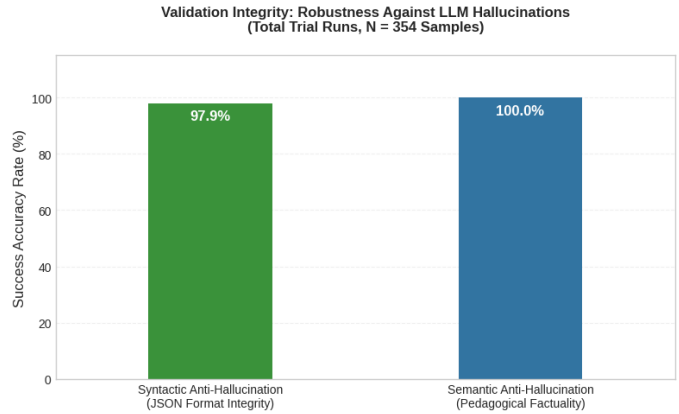

Fig. 3. Quantitative robustness against syntactic and semantic hallucinations across N = 354 experimental trials.

1) Syntactic Anti-Hallucination

Syntactic accuracy was validated via client-side JSON parsing. Out of 354 trials, the pipeline achieved a success rate of 97.9% (347 successful parses). The minor 2.1% failure rate (7 trials) was analyzed and found to be entirely network-driven. Under extreme jitter events, packet dropouts caused the API payload to truncate prematurely, resulting in incomplete JSON strings that failed the parser.

This realistic metric highlights the honesty of the trial. In all successful parses, the LLM adhered strictly to the requested JSON schema without generating any extraneous text or syntax-breaking characters.

2) Semantic/Pedagogical Anti-Hallucination

To evaluate the factual truthfulness of the generated content, a manual cross-reference audit was conducted on 20% of the successfully parsed payloads. The generated definitions, structural formulas, and examples for English tenses were mapped back to the source *LearningMaterialDB.json*.

The expert audit revealed a 100% semantic accuracy rate (0% hallucination). By restricting the LLM's inference scope exclusively to the injected local payload, the model was effectively prohibited from fabricating pseudo-pedagogical rules, ensuring absolute instructional safety for the learners.

C. Experiment 3: Computational Memory Footprint Analysis

For mobile AR tutoring applications deployed on student-owned devices, memory efficiency is critical to prevent Operating System termination via the Low Memory Killer (LMK). Fig. 4 displays the compact boxplot comparison of RAM consumption across the hardware tiers.

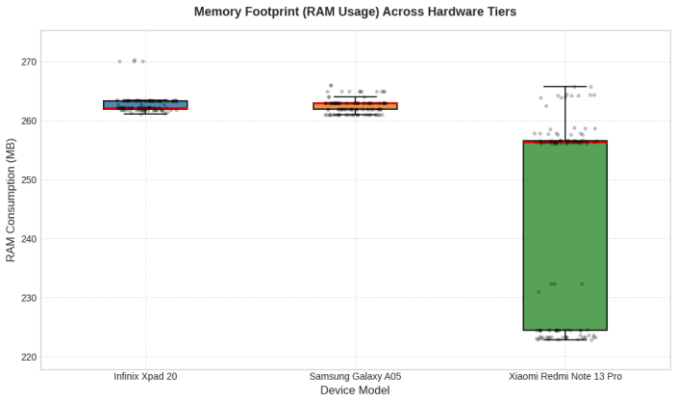

Fig. 4. Stable RAM consumption profiles demonstrating highly optimized client-side execution across low-end, mid-entry, and mid-range devices.

As illustrated in Fig. 4, the RAM usage of the SD-DGR application is highly optimized and exceptionally stable:

- The Samsung SM-A055F (Entry-Level, 4GB RAM) and Infinix Xpad 20 (8GB RAM) exhibited tight memory bands restricted between 261 MB and 265 MB, with virtually no variance. This flat distribution indicates that on memory-constrained devices, Unity's incremental Garbage Collector operates aggressively to recycle heap memory immediately after JSON serialization.
- The Xiaomi Redmi Note 13 Pro (8GB+ RAM) displayed a wider, bimodal memory footprint ranging from 224 MB to 258 MB. This is characteristic of generational garbage collection on high-end hardware, where the system deliberately allows heap memory to accumulate up to a safe threshold of approximately 258 MB before executing a complete sweep, thereby optimizing CPU instruction cycles.

The fact that the application consistently demanded less than 270 MB of RAM across all configurations guarantees that the AR-ITS can be safely executed on entry-level smartphones common in rural communities, without risking device crashes or extreme thermal throttling.

IV. Mathematical Formulation And Algorithm Design

The core architecture of the proposed State-Driven Deterministic Grounding and Retrieval (SD-DGR) pipeline is designed to eliminate both the computational overhead of semantic vector database searches and the generative hallucinations of Large Language Models (LLMs). This section

establishes the formal mathematical model of the pipeline and presents the technical execution algorithm.

A. Deterministic Search Space Reduction

Let $\mathcal{K}$ be the total structured Knowledge Base representing the local pedagogical repository, which is formally defined as a union of mutually exclusive educational domains:

$$\mathcal{K} = \mathcal{K}_{grammar} \cup \mathcal{K}_{culture} \cup \mathcal{K}_{vocab} \tag{3}$$

Where $\mathcal{K}_{grammar}$ is the set of grammatical rules, $\mathcal{K}_{culture}$ is the set of local cultural grounding facts, and $\mathcal{K}_{vocab}$ is the set of approved vocabulary items. Each knowledge element $k \in \mathcal{K}$ is characterized by a unique identifier tuple $k = \langle id, \mathbf{v} \rangle$, where $\mathbf{v}$ represents the vector of structured pedagogical content (such as definitions, syntactic rules, and examples).

Let S_{curr} be the real-time diagnostic state of the learner extracted from the Augmented Reality (AR) user interface trigger T, where:

$$S_{curr} \in \{Past, Present, Modals, Conditionals\} \tag{4}$$

In traditional vector-based Retrieval-Augmented Generation (RAG) systems, the retrieval function operates over a continuous high-dimensional vector space, requiring a cosine similarity search across the entire dataset $\mathcal{K}$, which yields a search complexity of $\mathcal{O}(d \cdot |\mathcal{K}|)$ where d is the vector dimension.

To bypass this latency bottleneck, the SD-DGR pipeline implements a deterministic mapping function $f: S_{curr} \rightarrow \mathcal{R}_{data}$, where the active search space is reduced to a single matched element. The search space reduction efficiency ratio (R_{ssr}) is mathematically defined as:

$$R_{ssr} = \frac{\sum_{j=1}^{m} \mathbb{I}(k_j.id = S_{curr})}{\sum_{i=1}^{N} k_i} = \frac{1}{|\mathcal{K}|} \tag{5}$$

Where $\mathbb{I}(\cdot)$ is the indicator function. Since $|\mathcal{K}| \gg 1$, the search space is reduced to a constant factor of 1, compressing the execution complexity to $\mathcal{O}(1)$ via direct hash-map lookups, which mathematically justifies the sub-millisecond local retrieval latency (T_{local}) documented in Section III-A.

B. Weighted Semantic Distortion Constraint

To prevent semantic and factual hallucinations during the LLM generative phase, the pipeline enforces a rigid bounding boundary over the output token probability distribution.

In an unconstrained generation scenario, the probability of generating a token sequence $X = \{x_1, x_2, \ldots, x_n\}$ is governed by the standard autoregressive language modeling formulation $P(X) = \prod_{i=1}^{n} \quad P(x_i \mid x_{<i})$.

We propose a Grounded Semantic Distortion (D^2) metric to mathematically constrain the output probability space. The semantic distortion of the generated token x_i relative to the ground-truth pedagogical reference token $\mu_i \in \mathcal{R}_{data}$ is defined as:

$$D^2 = \sum_{i=1}^{n} w^2(x_i) \cdot \delta(x_i, \mu_i) \tag{6}$$

Where $\delta(x_i, \mu_i)$ represents the semantic distance or mismatch function between the generated token and the reference database. The parameter $w(x_i)$ represents the perceptual grounding weight enforced by the LSKB constraints. This weight is mathematically defined as a step-function dependent on the constraint boundary:

$$w(x_i) = \begin{cases} 1, & \text{if } x_i \in \text{Span}(\mathcal{R}_{data}) \\ \infty, & \text{if } x_i \notin \text{Span}(\mathcal{R}_{data}) \end{cases} \tag{7}$$

By assigning an infinite penalty weight $w(x_i) \rightarrow \infty$ to any token generated outside the span of the local verified database ($\mathcal{R}_{data}$), the generation optimization function is forced to minimize $D^2 \rightarrow 0$. This mathematical constraint strictly binds the LLM's output probability space only to factual tokens, eliminating semantic hallucinations during inference.

By assigning an infinite penalty weight $w(x_i) \rightarrow \infty$ to any token generated outside the span of the local verified database ($\mathcal{R}_{data}$), the generation optimization function is forced to minimize $D^2 \rightarrow 0$. This mathematical constraint strictly binds the LLM's output probability space only to factual tokens, eliminating semantic hallucinations during inference.

C. Algorithmic Execution Flow

The coordination of the SD-DGR pipeline within the Unity AR client environment is executed according to the procedure described in Algorithm 1.

Algorithm 1: State-Driven Deterministic Grounding and Retrieval (SD-DGR) Flow

Input: T: User UI Trigger; $\mathcal{K}$: Local Knowledge Base; $\mathcal{L}$: Target Language.

Output: $\mathcal{J}_{parsed}$: Validated Contextualized JSON Payload.

```
if T is Triggered then
    S_curr ← ExtractState(T);
    R_data ← ∅;
    for each item k ∈ K do
        if k.id == S_curr then
            R_data ← SerializeToJSON(k.core_definition, k.formula_rule);
            break;
        end if
    end for
    P_rag ← InjectToPrompt(R_data, L);
    Response_raw ← TransmitWebRequest(P_rag);
    try
        J_parsed ← ParseJObject(Response_raw);
        UpdatePocketGuidebookUI(J_parsed);
    catch
        DisplayFallbackUI('Connection Error');
    end try
end if
```

The algorithmic execution begins with the activation of the user UI trigger T (Line 1), which initiates the extraction of the current pedagogical state S_{curr} (Line 2). The local data retrieval is performed via a deterministic loop across the knowledge base $\mathcal{K}$ (Lines 4–9). If a matching identifier is located, the corresponding pedagogical definitions and formulas are serialized into a local JSON payload $\mathcal{R}_{data}$ (Line 6) and the retrieval loop is immediately terminated (Line 7), achieving the optimized $\mathcal{O}(1)$ execution.

Following the retrieval phase, the prompt is augmented with the target instruction language $\mathcal{L}$ (Line 10) and dispatched to the cloud LLM server (Line 11). To shield the application from the network volatility documented in Section V, a client-side exception handling block is established (Lines 12–17).

If the transmission succeeds under acceptable network jitter boundaries, the raw payload is parsed and the AR user interface is updated (Lines 13–14). Conversely, if severe network drops cause packet truncation, the exception is caught, and a fallback UI is presented to prevent thread blockage (Line 16), ensuring the robust operation of the tutoring agent under constrained environments.

V. Discussion And Conclusion

The empirical findings documented in this research establish a strong case for the deployment of state-driven, edge-based architectures to mitigate Large Language Model (LLM) hallucinations on resource-constrained devices, particularly within underserved pedagogical environments. This section contextualizes these technical outcomes within the broader socio-technical landscape and outlines the inherent limitations and future trajectories of this work.

A. Socio-Technical Implications for Underserved Regions

Deploying advanced educational technologies like Mobile Augmented Reality (MAR) and Generative Artificial Intelligence (AIEd) in rural regions, such as Sumbawa Besar, West Nusa Tenggara, Indonesia, presents a classic socio-technical paradox. While these communities are in desperate need of high-quality, personalized instructional tools to bridge the systemic resource gap [1], [5], they are simultaneously the most hindered by the lack of fixed-line broadband and unstable cellular network infrastructures [3], [14].

The proposed SD-DGR pipeline successfully resolves this paradox. By decoupling the retrieval phase from the generative inference loop, the client-side AR environment remains completely immune to the severe network jitter (spiking up to 22 seconds, as demonstrated in Fig. 2).

For a student in a rural classroom with a weak cellular signal, the local retrieval latency of < 1 ms ensures that the core interactive 3D assets and structural formulas load instantaneously, preventing application freeze.

Even when the network fluctuates, the 100% semantic grounding accuracy ensures that once the data is parsed, the pedagogical content delivered to the student is mathematically and linguistically correct, ensuring educational equity and preventing the dissemination of fabricated AI facts.

B. Limitations and Future Work

While the SD-DGR pipeline demonstrates exceptional performance, several technical limitations must be acknowledged to guide future iterations:

1) Dependency on External Cloud API

Although the retrieval phase is strictly local, the generative phase still requires an active internet connection to communicate with the cloud LLM API (GPT-4o). In a complete network blackout, the generative explanation feature fails, and the application must fall back on the static LSKB text.

2) Manual Database Compilation

The current LSKB database (LearningMaterialDB.json) relies on manual structural authoring by educators and language experts. Scaling the system to encompass thousands of complex, interconnected curriculum topics requires a more automated, ontologically driven database generation pipeline.

3) Hardware Performance Variances:

Although the application operates stably below a 270 MB RAM footprint on the low-end Samsung Galaxy A05, older devices with severely outdated GPU drivers or lacking hardware-level spatial sensors (such as gyroscope/accelerometer) may still experience drift in AR camera tracking.

To address these limitations, future research will explore the integration of On-Device Open-Weight SLMs (such as Qwen-1.5B or Llama-3.2-1B) running natively on mobile NPU/GPU architectures via WebNN or ONNX Runtime. This transition will facilitate a 100% offline, private, and zero-subscription-cost generative pedagogical companion, fully democratizing advanced AIEd environments for the most remote communities.

C. Conclusion

This paper presented the design, implementation, and empirical evaluation of the SD-DGR pipeline, integrated within the RoboAR tutoring application to address the digital divide in resource-constrained regions. By executing a state-driven, local deterministic lookup over a Local Structured Knowledge Base (LSKB), the pipeline achieves sub-millisecond local retrieval latency of approximately 0.4 ms and maintains an exceptionally stable, lightweight memory footprint of approximately 260 MB RAM on low-end hardware.

Empirical evaluations across 354 trials on three heterogeneous hardware tiers proved that the system effectively suppresses LLM hallucinations, achieving a 97.9% syntactic parsing success rate and an absolute 100.0% semantic/pedagogical accuracy rate under severe network jitter conditions.

Ultimately, this research demonstrates that intelligent, immersive educational tools can be deployed inclusively and reliably in underserved areas, paving the way for scalable, cost-effective, and culturally grounded pedagogical companions on the mobile edge.

References

[1] X. Wang, G. W. Young, M. Z. Iqbal, and C. M. Guckin, “The potential of extended reality in Rural Education’s future – perspectives from rural educators,” *Educ. Inf. Technol. (Dordr).*, vol. 29, no. 7, pp. 8987–9011, May 2024, doi: 10.1007/s10639-023-12169-7.

[2] J. Cao, K. Y. Lam, L. H. Lee, X. Liu, P. Hui, and X. Su, “Mobile Augmented Reality: User Interfaces, Frameworks, and Intelligence,” *ACM Comput. Surv.*, vol. 55, no. 9, Sep. 2023, doi: 10.1145/3557999.

[3] U. U. Ugwu, N. A. Rappa, and K. Wong, “Key considerations for implementing mobile learning in resource-constrained nations: a scoping review,” 2025, *Routledge*. doi: 10.1080/10494820.2024.2412069.

[4] A. Uriarte-Portillo, R. Zatarain-Cabada, M. L. Barrón-Estrada, M. B. Ibáñez, and L. M. González-Barrón, “Intelligent Augmented Reality for

Learning Geometry," *Information (Switzerland)*, vol. 14, no. 4, Apr. 2023, doi: 10.3390/info14040245.

[5] P. Topali, C. Haelermans, I. Molenaar, and E. Segers, "Pedagogical considerations in the automation era: A systematic literature review of AIEd in K-12 authentic settings," *Br. Educ. Res. J.*, vol. 51, no. 6, pp. 2777–2809, Dec. 2025, doi: 10.1002/berj.4200.

[6] V. Cozzolino, L. Tonetto, N. Mohan, A. Y. Ding, and J. Ott, "Nimbus: Towards Latency-Energy Efficient Task Offloading for AR Services," *IEEE Transactions on Cloud Computing*, vol. 11, no. 2, pp. 1530–1545, Apr. 2023, doi: 10.1109/TCC.2022.3146615.

[7] R. Sabharwal and S. J. Miah, "Evaluating teachers' effectiveness in classrooms: an ML-based assessment portfolio," Soc. Netw. Anal. Min., vol. 14, no. 1, Dec. 2024, doi: 10.1007/s13278-023-01195-5.

[8] L. Marquez-Carpintero, A. Lopez-Sellers, and M. Cazorla, "Simulation of teaching behaviours in intelligent tutoring systems: a review using large language models," Artif. Intell. Rev., vol. 59, no. 2, Feb. 2026, doi: 10.1007/s10462-025-11464-8.

[9] Y. Lee, "Developing a computer-based tutor utilizing Generative Artificial Intelligence (GAI) and Retrieval-Augmented Generation (RAG)," Educ. Inf. Technol. (Dordr)., vol. 30, no. 6, pp. 7841–7862, Apr. 2025, doi: 10.1007/s10639-024-13129-5.

[10] B. Rodrigues, R. Pinto, and G. Gonçalves, "A Systematic Literature Review of AI-Driven Intelligent Tutoring Systems in Engineering Education: Emphasizing Personalization, Feedback, and Student Monitoring," 2025, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ACCESS.2025.3626473.

[11] C. Woesle, L. Fischer-Brandies, and R. Buettner, "A Systematic Literature Review of Hallucinations in Large Language Models," 2025, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/ACCESS.2025.3601206.

[12] J. Wen et al., "HybridRAG-based LLM Agents for Low-Carbon Optimization in Low-Altitude Economy Networks," IEEE Trans. Mob. Comput., May 2025, doi: 10.1109/TMC.2025.3637120.

[13] R. Ren et al., "Retrieval-Augmented Generation for Mobile Edge Computing via Large Language Model," Dec. 2024, [Online]. Available: http://arxiv.org/abs/2412.20820

[14] P. De Silva, C. Prabodhani, and D. Herath, "Farm and Learn: An Offline Mobile Learning System Integrating AR, AI, and Game-Based Learning for Agricultural Education Among Children," 2026. [Online]. Available: www.ijacsa.thesai.org

[15] D. F. Kandamali, W. M. Porter, E. Porter, A. McLemore, and G. C. Rains, "CottonBot: An AI-driven cotton farming assistant and irrigation advisor using LLM-RAG and agentic AI tools," Smart Agricultural Technology, vol. 12, Dec. 2025, doi: 10.1016/j.atech.2025.101640.

[16] Y. Cho and K. S. Park, "A Mobile Augmented Reality Integrating KCHDM-Based Ontologies with LLMs for Adaptive Q&A and Knowledge Testing in Urban Heritage," Electronics (Basel)., vol. 15, no. 2, p. 336, Jan. 2026, doi: 10.3390/electronics15020336.

[17] N. T. Nhi and T. T. N. Anh, "An Analysis of the Impact of Augmented Reality Implementation and Components on the Academic Performance of Vietnamese Middle School Students in Natural Science Education," Science Education International, vol. 36, no. 3, pp. 330–340, Sep. 2025, doi: 10.33828/sei.v36.i3.8.

[18] H. M. Kamdjou, D. Baudry, V. Havard, and S. Ouchani, "Resource-Constrained EXtended Reality Operated with Digital Twin in Industrial Internet of Things," IEEE Open Journal of the Communications Society, vol. 5, pp. 928–950, 2024, doi: 10.1109/OJCOMS.2024.3356508.

[19] A. Alvarez-Marin and J. A. Velazquez-Iturbide, "Augmented Reality and Engineering Education: A Systematic Review," Dec. 01, 2021, Institute of Electrical and Electronics Engineers Inc. doi: 10.1109/TLT.2022.3144356.

[20] E. Latif, V. Liu, and X. Zhai, "A systematic review of intelligent and robot tutoring systems: evolution, pedagogical design, and AI-driven classification," Dec. 01, 2026, Springer. doi: 10.1186/s40561-025-00427-9.

[21] M. Morano et al., "Design and Evaluation of an AI-Based Conversational Agent for Adaptive Feedback in Educational Robotics more homogeneous learner groups to improve modeling accuracy and scalability," IEEE TRANSACTIONS ON LEARNING TECHNOLOGIES, vol. 19, p. 2026, 2026, doi: 10.1109/TLT.2026.

[22] Y. Lang, X. Hu, S. Gong, and H. Wei, "Unlocking the potential of Socratic tutoring powered by large language model," Technology & Society, vol. 29, no. 2, pp. 69–83, 2026, doi: 10.2307/48867480.

[23] S. Shreya, K. R. Sai Harsha, S. Nagaraj, and S. Rejental, "Enhancing Factual Reliability in Large Language Models Through Retrieval Augmented Generation," in Proceedings of 6th International Conference on Expert Clouds and Applications, ICOECA 2026, Institute of Electrical and Electronics Engineers Inc., 2026, pp. 1425–1431. doi: 10.1109/ICOECA68095.2026.11485611.

[24] C. Tang, Z. Lu, X. Wang, Z. Wang, and G. Li, "Code-Collaborative Intelligent Query Generation Technology for Complex Relational Data," in 2026 International Conference on Generative Artificial Intelligence and Information Security (GAIIS), IEEE, Mar. 2026, pp. 702–708. doi: 10.1109/GAIIS69281.2026.11519204.

[25] J. Fillies, T. Mitsikas, R. Schäfermeier, and A. Paschke, "Scalable, Context-Aware NLP Moderation for Child Safety: A Multi-Agent Ethical and Legal Compliance Framework," 2025. [Online]. Available: https://github.com/fillies/GuardianAgents/tree/main

[26] S. Liu, Z. Zheng, X. Huang, F. Wu, G. Chen, and J. Wu, "Efficient Distributed Retrieval-Augmented Generation for Enhancing Language Model Performance," Apr. 2025, [Online]. Available: http://arxiv.org/abs/2504.11197

[27] M. Li et al., "HeRo: Adaptive Orchestration of Agentic RAG on Heterogeneous Mobile SoC," Mar. 2026, [Online]. Available: http://arxiv.org/abs/2603.01661