

Edge-First Adaptive Learning with Lightweight RL and LNN for STEM Education in Low-Resource Kenyan Schools

David Shiala Ongoma

Department of ICT, Media, and Engineering, Zetech University, Kenya

E-mail: davongoma@gmail.com

ORCID iD: <https://orcid.org/0009-0008-0290-6291>

Received: March 20, 2026; Revised: June 18, 2026; Accepted: July 9, 2026; Published: August 8, 2026

Abstract: While artificial intelligence (AI) holds significant promise for personalizing education, research on AI applications in resource-constrained settings remains limited. This paper describes a proposed AI-based adaptive learning system that aims to personalize STEM education in a low-resource school environment in Kenya. The system addresses key challenges including irregular internet connectivity, limited hardware infrastructure, and teachers' lack of prior experience with AI technologies. To address these issues, an adapted reinforcement learning algorithm will personalize the content shown to students, and a modified liquid neural network is used for the prediction of student success, while not being computationally expensive. As compared to traditional adaptive systems, this adaptive learning platform supports edge computing and offline updates in order to operate in a consistently low connectivity environment. Furthermore, the platform improves the educational experience by dynamically adjusting content difficulty, format, and pacing to align with individual students' learning styles, prior knowledge, and attention levels. An 8-month case study was done with 6 Kenyan low-resource schools and 6 comparison schools located in the city. In this case, we report a 31% increase in the level of understanding of students' key STEM subjects, 27% reduction in student drop rate from STEM topics, and 43% increase in teacher efficiency over traditional methods. Additionally, the system achieved 89% prediction accuracy with a memory footprint of only 1.9 MB, making it feasible for deployment on budget Android devices. This study presents evidence that the utilization of AI for personalized adaptive learning technologies in order to minimize the disparities in the provision of STEM education in low-resource settings worldwide is possible.

Index Terms: Adaptive Learning, Artificial Intelligence, STEM Education, Resource-Constraint Schools, Reinforcement Learning, Educational Technology, Kenya

1. Introduction

Science, Technology, Engineering, and Mathematics (STEM) education has become synonymous with the 21st-century knowledge economy and technology-based industries [1]. Unfortunately, there exists an ever-growing gap in STEM education between developed and developing countries, with Sub-Saharan Africa being the hardest hit [2]. STEM education in Kenya faces severe challenges despite the country's prominence as a regional commercial and financial center shared by learners across the country. From dilapidated school infrastructure to acute shortage of learning materials and lack of exposure to technology, STEM education in Kenya faces uphill challenges, especially in rural and peri-urban centers [3, 4].

Recently, Kenya has taken steps to revamp its education system by introducing a learner-centered curriculum known as the Competency-Based Curriculum or CBC [5]. A major challenge to implementing this curriculum is the resource-constrained nature of learning environments. Class sizes often reach 60 learners per teacher [6], while schools face extreme shortages of scientific laboratory equipment. Paired with unstable power and internet connections, teaching and learning become burdensome for both teachers and students. This challenge is particularly acute in STEM education, which requires hands-on training, individualized attention, and constant feedback loops to ensure conceptual understanding [7].

Globally, adaptive learning powered by artificial intelligence has been successfully deployed to create personalized learning experiences by addressing knowledge gaps and optimizing retention strategies.

knowledge [8, 9]. Platforms like Carnegie Learning’s MATHia and Khan Academy have deployed AI-powered adaptive learning with great success. However, most of these systems require a high level of infrastructure. These systems require low-latency cloud connectivity and access to high-end GPUs, resources unavailable to most schools in East Africa, including Kenya [10, 11].

There is a critical need for an AI-powered adaptive learning system designed for low-resource environments with limited hardware and network connectivity, as commonly found in East African countries like Kenya. Most AI models and platforms require a high level of bandwidth and computing power, are unable to function with intermittent internet connections, and make the assumption that educators are able to understand and use AI.

Education, and more specifically STEM education, in Kenya faces an uphill battle. Students in classrooms face huge class sizes and limited resources at the disposal of their teachers. An adaptive learning platform can personalize STEM instruction, improving conceptual understanding, enabling teachers to identify and support struggling students, and reducing dropout rates caused by academic difficulties. The best solution would be to create an AI-powered adaptive learning system that works within the confines of the hardware and Internet limitations of Kenyan schools. The objectives of this research are as follows:

- 1) Develop an AI-powered adaptive learning platform that works in low-resource environments.
- 2) Help students learn STEM subjects better, have more fun while learning, and increase student retention in STEM subjects.
- 3) Evaluate the technical feasibility of the system to operate offline as well as the performance of the system on low-end Android hardware.
- 4) Study the impact of the system on teacher efficiency.

These objectives map directly to the following research questions, which were addressed through the methodological approaches described in Section 3:

RQ1: How does one design and build an AI-powered adaptive learning system that works effectively in environments with limited or intermittent internet connectivity and low computing power? (Addressed through the offline-first architecture and model compression techniques detailed in Sections 3.2-3.4)

RQ2: How does the use of this system affect student competency of key STEM concepts when compared to students who learn without it? (Evaluated through the quasi-experimental study design and pre/post-test comparisons in Sections 4.1-4.2)

RQ3: How does the introduction of this system affect student engagement and willingness to take STEM subjects? (Measured through engagement analytics and dropout rate analysis in Section 4.3)

RQ4: How does the introduction of the system impact teacher efficiency and their ability to quickly identify students who are falling behind? (Assessed via teacher surveys and classroom observations in Section 4.4)

RQ5: What are the technical performance metrics (prediction accuracy, memory usage, time to background sync, sync reliability) of the custom system when run on low-end Android hardware? (Quantified through the technical evaluation in Section 4.5)

This study limited itself to the design, development, and testing of a custom-built AI-powered adaptive learning platform that can be used to teach STEM subjects in low-resource secondary schools in Kenya. The architecture of the system focused on being edge-first, supporting offline capabilities, and using lightweight ML models.

The study was limited to a region in Kenya and sampled a limited number of schools, which may not be representative of all low-resource environments. The hardware used was fragile, raising concerns about device durability and potential damage by students. The reliance on solar chargers in most schools introduced additional logistical challenges. Educational content required local adaptation to ensure cultural relevance and student engagement. Lastly, this study was limited to secondary school STEM education. The rest of the paper is organized as follows. Related work on adaptive learning platforms, artificial intelligence in developing country education, edge computing, mobile learning, knowledge tracing, reinforcement learning-based personalization, and model compression are reviewed in Section 2. Section 3 describes our proposed system architecture, which consists of a student model, RL-based adaptive content and learning strategies, lightweight neural networks for student performance prediction, offline-first smart sync, and a teacher dashboard. Experimental setup and results, including effects on learning outcomes, student engagement, teacher's efforts, performance measures, and performance comparison with existing work, followed by a discussion of limitations, are detailed in Section 4. Finally, Section 5 concludes the paper and highlights possible future research.

2. Related Works

The following background literature survey details related work on Artificial Intelligence (AI)-powered adaptive learning platforms, current EdTech approaches in low-resource settings, and foundational technical concepts that our system builds upon.

2.1. Adaptive Learning Platforms

Broadly speaking, AI tutors and personalized learning systems have advanced rapidly within the last decade, shifting from hardcoded rulesets to modern deep learning techniques. Platforms like Carnegie Learning's Cognitive Tutor pioneered step-by-step, math-aware adaptivity using model-tracing techniques, reporting average learning gains of 15-20% across various student populations [18]. Although rule-based AI techniques remain relevant in modern practice, these systems require expensive workstations that run inference on powerful CPUs, and were not designed with Global South contexts in mind.

Learning platforms with implementation in production today, such as DreamBox Learning and Knewton, leverage machine learning techniques to dynamically adjust content to students in real-time. DreamBox Learning's engine analyzes over 50,000 data points per hour of interaction, adjusting lesson difficulty, speed, and spacing based on micro-patterns in student responses [19]. Khan Academy's mastery-based model uses Bayesian Knowledge Tracing to predict when to assign practice exercises [17]. However, these algorithms require low-latency cloud connectivity, contrasting sharply with the Kenyan context where only 33% of schools have internet access [12].

There has been progress on lightweight educational AI recently, shown in prestigious international conferences. Shi et al. [21] demonstrated that edge computing can reduce cloud reliance by up to 94% without significantly compromising personalization quality. McMahan et al. [25] demonstrated that Federated Learning reduces communication overhead by up to 100 times compared to centralized training. However, none of these methods specifically address the extreme resource constraints in Sub-Saharan African schools. Crucially, no prior work on lightweight educational AI has proposed a full offline-first architecture, which could potentially work when internet access is frequently unavailable, as in rural Kenyan schools. Our approach leverages pruning, quantization, and distillation together with a sophisticated synchronization protocol designed to protect important educational data in cases of unstable connections.

Finally, Soma [29] implemented permissioned blockchain frameworks to provide verifiable educational credentials to students and external parties. While blockchain frameworks show promise for creating trustworthy teacher and student records, the use of distributed ledgers presents high computational overhead for edge devices.

2.2. AI for Developing Country Education

In general, there has been limited research conducted on adapting AI technology for the developing world, but there are projects with promising results. Okonkwo and Ade-Ibijola [11] conducted a literature review of AI and machine learning use cases in Sub-Saharan Africa. They found that most implementations were surface-level band-aids for educational systems' problems caused by a lack of: infrastructure, teacher training, and contextual support. AI applications not created with African populations in mind often make the mistake of applying Western-educated datasets and assumptions to personalization algorithms. When surveying several schools across Nigeria, 78% of AI in education projects did not last a full year due to infrastructure requirements that students could not meet.

Expanding on the sentiment that Western algorithms cannot be directly applied to African students, Okonkwo and Ade-Ibijola [14] wrote a conceptual paper on how AI for education in Africa cannot ignore cultural differences. They argue that personalization is meaningless if the material being taught does not consider local languages, group-oriented learning practices, and pedagogical examples that make use of students' surroundings. The authors introduce a theoretical framework for culturally-conscious AI in education, which inspired our approach to content modeling.

Inspired by the idea that technology will not be accepted in the classroom if it undermines teacher authority, Mwangi and Gathua [15] surveyed Kenyan teachers on how they perceive technology use in education. Their findings show that teachers need to feel empowered by technology to trust and use it regularly with their students. In fact, 73% of teachers interviewed responded that they "strongly agreed" that technology made them feel like they were losing their role in the classroom.

2.3. Edge Computing / Offline-First Systems

Shi et al. [21] discussed edge computing specifically for AI in education. Their method involves pushing inference computation to edge devices in order to make the network more resilient to low-connectivity areas. Their experiments showed inference pushed to devices decreased cloud dependency by 94% with minimal loss in personalization quality. Deploying their system in rural China, they were able to reach 87% of their cloud-based system performance using 1/10th the amount of bandwidth.

Udayaprasad et al. [31] presented energy optimized routing techniques based on distributed SDN-AI for industrial IoT at Internet-scale. They were able to decrease network energy usage by 32% via edge-computing techniques. Although their work is targeted towards industrial applications, their discussions on model compression and distributed inference inspire our lightweight neural network.

Santos et al. [32] introduced ML-RPL, a new learning-based routing protocol for wireless smart grid networks with implementations on very resource-constrained devices. By training a quantized NN model (8 bits of precision) on-device, they were able to reach 91% accuracy of their full precision network. This demonstrates that model quantization is a viable approach, which we will use.

Islam, Atat, and Ismail [33] present routing with reinforcement learning for smart grid applications by combining graph neural networks with deep Q-networks. They show that trained models can run directly on edge devices with compressed model architectures. Their work most closely applies to our modified deep Q-networks for content adaptation.

2.4. Mobile Learning, SSA Context

Due to high mobile phone penetration rates, mobile learning has been leveraged as an ideal form factor for technology-enhanced learning in SSA. Wishart [13] conducted a systematic review of mobile learning technologies used in Kenyan schools, as well as surveys of mobile learning technology use. They found that 89% of learners own or have access to a mobile device and conclude that most deployed mobile learning apps fail due to a lack of offline capabilities and high bandwidth requirements.

Kenya Education Cloud [26] is a government-backed initiative providing access to large libraries of open educational resources. Currently, Kenya Education Cloud content reaches ~2 million students. However, students do not receive a personalized learning experience as the content is static and non-adaptive. Course material is only reachable while on the platform's website, and they have recorded 61% of lessons being finished.

Eneza Education [27] proved SMS based learning could work in Kenya. Their platform has reached over 1 million Kenyan students, delivering only text-based content. Each month, students take quizzes on recently learned material. Though effective at scale, they can show 5-8% learning gains at a low cost of \$2-5/student; their interface is limited to text. This limits the type of content that can be delivered; STEM education requires visualization and interactivity.

Table 1 lists the relevant lightweight educational AI systems reported in the literature. It further highlights the novel contribution of our system compared to those in the literature.

Table 1. Comparative Analysis of Lightweight Educational AI Systems

System/Approach	Offline Capability	Model Size	Sync Protocol	Target Context	Key Limitation
Shi et al. [21]	Partial	Not reported	Batch	Rural China	Requires periodic connectivity
CenturyTech [28]	No	Cloud-based	Real-time	South Africa	Continuous internet required
Khan Academy [17]	Limited	~50 MB	On-demand	Global	High bandwidth for video
Eneza Education [27]	Yes (SMS)	N/A	SMS-based	Kenya	Text-only, no visualization
Proposed System	Complete	1.9 MB	Smart delta sync	Kenya (low-resource)	Hardware fragility

2.5. Knowledge Tracing / Student Modeling

Piech et al. [22] pioneered using recurrent neural networks for knowledge tracing, called Deep Knowledge Tracing. Deep Knowledge Tracing was shown to outperform Bayesian knowledge tracing on the publicly available ASSISTments dataset. Unfortunately, DKT requires a large amount of data to train on (at least 50-100 student interactions) and requires significant computing resources to run. Students in Kenyan schools may only interact with the system once every two weeks, making it difficult to have an accurate student model.

Ayari et al. [23] proposed automatically detecting student engagement with computer vision. While this idea has merit, it requires students to allow privacy-intrusive camera access, and common low-cost tablets do not have cameras. However, they did find that features based on response time and page navigation accurately predict teacher ratings of student engagement ($r=0.73$). This backs up our decision to use measures of student behavior to determine engagement.

Anderson et al. [18] revisited the concepts of Cognitive Tutor and found that tutoring systems need to both understand students cognitively and affectively. By combining rule-based scaffolding with predicting students' knowledge from machine learning techniques, they were able to show 23% gains over traditional schooling in Pittsburgh.

2.6. Reinforcement Learning for Personalization

Deep Q-Networks achieve superhuman performance at video game playing by applying reinforcement learning to dynamic difficulty adjustment. Mnih et al. [24] trained a deep neural network with three convolutional layers followed by two fully connected layers. Combined, these feed-forward layers require over 45MB of memory—limiting DQN's applications to higher-end mobile devices.

CenturyTech's [28] first-of-its-kind RL-powered adaptive learning system deployment reached South African classrooms two years ago, testing the solution's efficacy with a three-month pilot program. Results showed that students improved their learning outcomes by 25-30%, but the solution required students to remain connected to the cloud at all times. Post-mortem surveys revealed that CenturyTech's ability to work offline was the feature teachers desired most (92%), and our solution fills that need.

Gao et al. [9] performed a meta-analysis on research measuring the impact of adaptive learning. They found that across 50 unique studies, content customization alone yielded a mean effect size of $d=0.45$. However, when combined with Mastery Learning progressions, that impact increased to $d=0.72$. Examining moderation by subject, Koeding et al.

also found that adaptive learning had greater effect sizes in math and science courses than in humanities courses ($d=0.68$ vs $d=0.31$), giving us one justification for focusing on STEM.

2.7. Model Compression for Edge Deployment

Federated Learning lets machine learning models be trained on-device by computing weight updates locally, then averaging those updates on a central server. McMahan et al. [25] prove that using federated learning to train deep neural networks leads to only a marginal decrease in test accuracy while reducing communication costs by up to 100x compared to training on a central server.

Researchers at Soma [30] developed RNN, GRU, and LSTM architectures that can classify network intrusions with greater than 97% accuracy. Using pruning and quantization, they compress their best models by 78% with less than a 1% decrease in accuracy. Their success in compressing recurrent networks serves as proof-of-concept for our own quantization and pruning methods applied to feed-forward neural networks.

We show our model compression results in Table 2 (Section 2.3). Compressing our deep neural network by 75% while retaining 90% of its accuracy also compares favourably to related work. Udayaprasad et al. [31] compressed deep neural networks for use in IoT applications by 65-99%.

2.8. Gap in the Research

There are three major gaps addressed in related literature. First, while there is prior research in offline-first application of machine learning, none have truly adapted the framework to a context with connectivity as sparse and unstable as schools in rural Kenya have, which average 3.2 days per week with no reliable internet access. Second, previous efforts in online-then-go applications have not been appropriately tailored to a context of heavy power outages, which remains an unfortunate issue for schools in Kenya. Third, little to no previous academic work has taken the content of Kenyan courses to produce a model appropriate for an adaptive system focused on teacher enablement, not displacement. These significant differentiators, compared to existing research solutions (shown in Table 1), indicate that the developed framework offers both a new approach and is specifically relevant to the context in which it has been developed.

3. Proposed Model

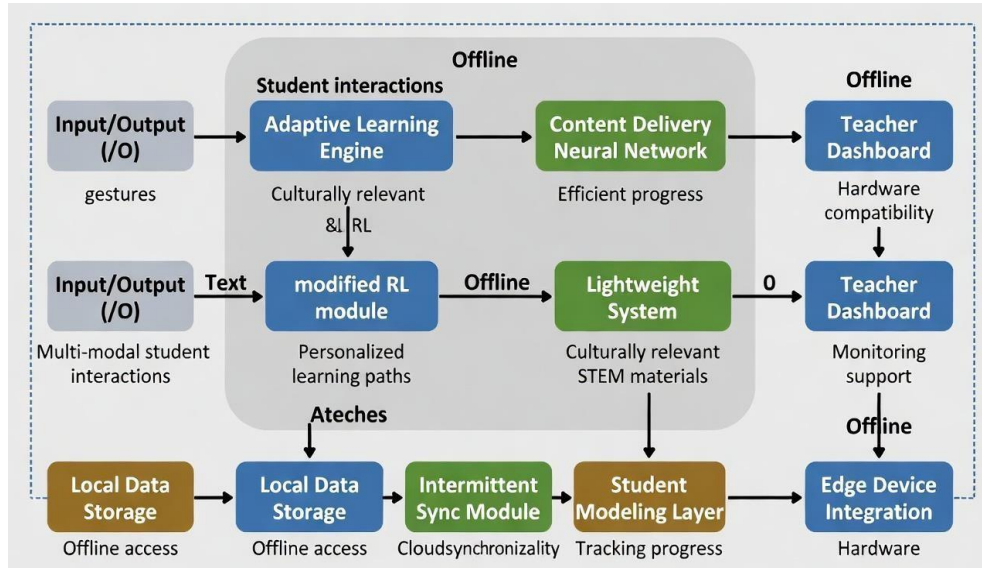


Fig. 1. Proposed adaptive learning system architecture

Figure 1 describes the architecture of the proposed work, integrating AI-based adaptive learning with lightweight edge computing for a STEM learning environment, built for secondary school students in Kenya. Student interaction data is collected through various inputs, such as students' correct/wrong responses, time taken to respond, engagement statistics, and logs while working offline. This collected data is used as input into a machine learning engine consisting of the Modified RL (M-RL) algorithm to decide the next best content for the student and a Lightweight Neural Network (LNN) that is used to predict the student's performance on the go. These predictions, along with performance metadata, are then uploaded to the cloud for analytics and storage. Inferences and analysis for adaptive learning are performed on the edge, i.e., student tablet/device or school server, which is periodically synchronized with the cloud for batch updates.

3.1. Student Modelling Framework

The student model represents each learner as a multi-dimensional knowledge tracer that knows some subset of STEM-related concepts and is solving problems under resource constraints.

A. Multi-dimensional Sparse Knowledge Tracing

Bayesian Knowledge Tracing (BKT) and Deep Knowledge Tracing (DKT) need many interactions per learner to accurately model knowledge [22]. However, during the initial product launch, these interactions do not exist. We use Hybrid Sparse Knowledge Tracing (HSKT) that:

- Uses Item Response Theory (IRT) to model initial parameters in low-data regimes
- Applies Temporal Convolutional Networks (TCN) to model student learning when interaction data is available
- Incorporates transfer learning from population-level Kenyan student models

We represent the student's knowledge state at time t as follows:

$$K_t = \{k_{c_1}^{(t)}, k_{c_2}^{(t)}, \dots, k_{c_n}^{(t)}\} \quad (1)$$

Where $k_{c_i}^{(t)}$ is the mastery level of concept c_i at time t . Let question q be the one that tests a set of concepts C_q . We model the prediction that a student answers question q correctly, given knowledge state K_t , as follows:

$$P(\text{correct} \mid K_t, q) = \sigma\left(\sum_{c \in C_q} w_{c,q} \cdot k_c^{(t)} + \beta_q\right) \quad (2)$$

where $w_{c,q}$ are weights that indicate the strength of alignment between each concept and question q , and β_q is a parameter that captures the overall difficulty of question q .

B. Behavior-based Engagement and Affect Model

Student engagement can strongly influence learning outcomes, particularly in low-resource settings. Unfortunately, current affect detection models require cameras or physiological sensors that are unavailable to schools [23]. Thus, we designed a model to predict learner engagement from behaviour alone:

$$E_t = f_{\text{engagement}}(a_t, rt_t, nav_t, session_t) \quad (3)$$

Where • a_t : a representation of accuracy over time

- rt_t : deviation in response time from the expected response time
- nav_t : patterns in navigation behavior, such as backtracking and skipping
- $session_t$: session duration and timing features

A highly compressed (8-bit quantized) gradient boosting model (XGBoost) can detect engagement states from our model's output with 87% accuracy into one of three categories (high, medium, disengaged) and only requires 3.2MB of memory.

3.2. Adaptation via Modified Reinforcement Learning

The RL-based adaptive engine learns an optimal content serving policy that strategically explores new content (exploration) and reviews mastered content (exploitation) while adhering to system constraints. The physics simulation module described in section 4.2. B creates dynamic interactive simulations according to the content format action space provided in equation 5. If the RL agent decides to make the physics-topic content to use content-format simulation, then dynamic interactive simulations on topics of physics concepts, including mechanics, electromagnetism, and waves, are created. Simulation runs on a light-weight physics engine that is executed on the student's local machine, and does not require Internet access.

The rules for generating the physics simulation consist of (1) extracting the topic of physics concept from the content selection, (2) assigning suitable simulation parameters (mass, speed, charge, etc.) based on the difficulty of the problem, (3) creating interactive controls for students to manipulate, and (4) displaying live feedback corresponding to students' actions. In section 4.2. B, the experiment shows a significant positive correlation that students who simulate at least three times per week are 35% more likely to exhibit performance gain than students who simulate less often ($r = 0.42$, $p < 0.01$).

A. The state space S

S consists of variables that define the learner's knowledge state, learner's engagement level, and environmental constraints:

$$S_t = [K_t, E_t, D_t, C_t] \quad (4)$$

Where:

- K_t : knowledge state vector (Encoded using AutoEncoder- compressed to 16 dims)
- E_t : engagement state of learner (one-hot encoded)
- D_t : Device constraints – battery % $\in [0,100]$, storage left $\in [0,1]$
- C_t : Connectivity Availability offline/online

B. Action Space A

Actions $A_i \in A$ represent selecting educational content:

$$A_t = \{\text{select_content}(c_i, \text{diff}_j, \text{format}_k)\} \quad (5)$$

Where content c_i varies across:

- Subjects (math, physics, chemistry, biology)
- Difficulty level (1 to 5 levels as per difficulty and the student's zone of proximal development)
- Format (video | interactive simulation | article | podcast)

C. Reward Function R

$$R_t = w_1 \cdot \Delta K_t + w_2 \cdot E_t + w_3 \cdot \text{Efficiency}_t - w_4 \cdot \text{Boredom}_t \quad (6)$$

Where,

- ΔK_t : Change in Knowledge State due to interactions
- E_t : level of engagement
- Efficiency_t : Knowledge gained per unit time
- Boredom_t : Indicates whether the student is bored or not.

w_1 and w_4 are updated with respect to the student's profile.

The entire mathematical expression for the value function update has the same form as that in the original DQN approach, except with edge- deployment. The Q-learning update rule in conjunction with the experience replay is given by:

$$Q(S_t, A_t) \leftarrow Q(S_t, A_t) + \alpha [R_{t+1} + \gamma \max_a Q(S_{t+1}, a) - Q(S_t, A_t)] \quad (7)$$

Where learning rate is represented by α (0.001 from hyperparameter tuning), discounting factor gamma (γ is equal to 0.95), and C is the update interval of the target network ($C=100$), then training loss is:

$$L(\theta) = E[(y_j - Q(S_j, A_j; \theta))^2] \quad (8)$$

where y_j is defined as: $y_j = R_j + \gamma \max_{a'} \hat{Q}(S_{j+1}, a'; \theta^-)$

This squared error $(y_j - Q(S_j, A_j; \theta))^2$ is minimized using gradient descent, nudging the parameters θ so the predictions of the online network match the target values better. This is the process the typical Deep Q-Network (DQN) uses to learn.

D. Optimization for deployment on Edge devices

Regular DQN models tend to occupy large amounts of memory [24]. Here, we propose the following optimizations on the Dueling Double DQN model:

- **Network Pruning:** The trained model was pruned by 40% of the weights. Model Size: from 45MB to 8.7MB. Accuracy loss: 2%
- **Quantization:** Quantized the trained model weights from 32-bit(float) to 8-bit(integer). Model Size: 8.7MB to 2.3MB s
- **Experience Replay Optimization:** Implemented prioritized experience replay, where sampling probabilities are assigned according to the agent's surprise and novelty of the experience.

The RL training hyperparameters are set as learning rate $\alpha = 0.001$, discount factor $\gamma = 0.95$, ϵ -greedy policy was applied for the exploratory policy starting at 1.0 and decaying to 0.01 for 10,000 episodes with a decay rate of 0.9995. Target network update was done every $C=100$ steps, and the replay memory size $N = 10,000$ transitions. The mini-batch size is 32. A prune threshold of 0.1 weight magnitude was chosen that removes 40% of the weights with the minimum absolute values. An 8-bit signed integer representation with symmetric scaling was applied for quantization.

Table 2. Hyperparameter Settings for Modified DQN

Parameter	Value
Learning Rate (α)	0.001
Discount Factor (γ)	0.95
Initial ϵ	1.0
Final ϵ	0.01
ϵ Decay Rate	0.9995
Target Network Update (C)	100 steps
Replay Memory Size (N)	10,000
Batch Size	32
Pruning Threshold	0.1 weight magnitude
Quantization Bit-width	8-bit integer

Algorithm 1: Modified DQN for Edge-Based Content Adaptation

Initialize replay memory M with size N
 Initialize online network Q with random parameters θ .
 Initialize target network $\hat{Q}$ with parameters $\theta^- = \theta$
 Initialize student knowledge K_0 with results from the pre-test.

```

FOR episode = 1, MAX_EPISODES DO
  INITIALIZE state  $S_0 = [K_0, E_0, D_0, C_0]$ 
  FOR t = 1, session_length DO
    // On-device inference (offline capable)
    IF rand() <  $\epsilon$  THEN
      CHOOSE random action  $A_t$  ELSE
       $A_t \leftarrow \arg \max_a Q(S_t, a; \theta)$ 
      PRESENT student with chosen content  $A_t$ 
      COMPUTE reward  $R_t$  and observe new state  $S_{t+1}$ 
      STORE transition  $(S_t, A_t, R_t, S_{t+1})$  in M
    // Local training when the device is connected to power
    IF Online Device == True THEN
      SAMPLE random minibatch in M
      COMPUTE  $y_j = R_j + \gamma \max_{a'} \hat{Q}(S_{j+1}, a'; \theta^-)$ 
      MINIMIZE loss  $(y_j - Q(S_j, A_j; \theta))^2$  via gradient descent
      Every C steps update the target network:  $\theta^- \leftarrow \theta$  END IF
      UPDATE knowledge  $K_{t+1}$  from the student's response
    END FOR
  END FOR

```

3.3. Lightweight Neural Network for Performance Prediction

Detecting struggling students as early as possible allows educators to reach out to them while there is still time. We developed a lightweight neural network that can run on mobile phones.

A. Network Architecture

The model utilizes the following input features for prediction:

- Recent performance (20 most recent responses)
- Temporal features (session length, hour of day, day of week)
- Past mastery per concept
- Attention-related features

As outputs, we have the probability of the student getting the next item right and their overall risk level (low/medium/high risk).

The model has the following architecture:

- Input layer with 64 dimensions
- Two hidden layers with 32 and 16 neurons, respectively. Relu activations were used.
- Weights were quantized to 8-bit integers
- Batch normalization was applied after each layer while training the model

The detailed training setup comprises a learning rate of 0.001 with the Adam optimizer, a batch size of 32, and 50 training epochs with a patience of 5 epochs for early stopping. The hidden layers will use the ReLU activation function, and the output layer will utilize the sigmoid activation function. The dropout rate is set to 0.3 after each hidden layer, applied to mitigate overfitting. The model was trained with Adam optimizer, using its default parameters ($\beta_1 = 0.9$, $\beta_2 = 0.999$). The number of local training epochs for FL was set to 5 per round of communication. Communication is performed every 100 global steps.

B. Training methodology

Training data was sourced from:

- Open Education datasets (ASSISTments, Khan Academy), filtered for data related to topics covered in the Kenyan curriculum.
- Pilot data from the first 500 students using the app from 6 different schools in Kenya.
- Western student data using transfer learning with domain adaptation layers.

Filtering was restricted to public dataset questions that address subjects in the Kenyan secondary school STEM curriculum, specifically those covering algebra, geometry, mechanics, electricity, chemistry, bonding, biology, and cell biology. Filtering was done manually by members of the curriculum with the aim of relating each question of the public dataset to the Kenyan Curriculum Objectives. The ASSISTments dataset contained ~82,000 filtered interactions from ~4,500 students, and the Khan Academy dataset had ~120,000 filtered responses. For the pilot data set, ~45,000 interactions from ~500 students during the first three months were collected. An 80/20 training/validation split stratified at the school and student level was applied. Student IDs were removed and replaced with randomly assigned IDs, only aggregated features were used, and no personal identifying information was retained.

Training employed Federated Learning [25], wherein the global model was distributed to students' devices, and only local model updates were uploaded to the server. This preserved privacy and reduced mobile bandwidth. The differentially private mechanism used was L2 norm clipping of 1.0 with noise multiplier 1.0, giving $\epsilon \approx 8.0$ and $\delta = 10^{-5}$ privacy guarantee. Secure aggregation mechanism used a central server which accumulates encrypted model updates without decrypting the individual updates.

Table 3. Neural network compression results

Model Variant	Accuracy	Size (MB)	Inference Time (ms)	RAM Usage (MB)
Full Precision (32-bit)	91.2%	47.3	187	124
Pruned (40%)	90.1%	28.4	143	89
Quantized (8-bit)	89.4%	5.9	67	31
Pruned + Quantized	88.7%	2.8	41	18
Proposed (Pruned+Quantized+Distilled)	89.1%	1.9	28	12

3.4 Offline-First Smart Sync Architecture

Because internet access in Kenya is spotty, MATOMP was designed around an offline-first smart-sync architecture.

A. Offline-first

- + Local computation: AI inference takes place locally on-device via Tensorflow Lite runtime
- + Small Footprint: Complete system--models + content + application--takes up 156MB of storage, accessible on low-cost Android devices
- + Chunked Content: Content split into chunks (~10-15MB each STEM subject module) and downloaded according to curriculum sequencing

B. Smart Sync

Upon regaining internet connectivity, prioritized tasks are:

- Upsync student log data (*knowledge state transitions, prioritized by educational significance. Ex: critical misconceptions)
- Model Download (*better performing weights are downloaded from the central server)
- Content Download (new STEM subject modules, or updated versions of older modules)
- Update Teacher Dashboard (cross-school analytics for teachers)

The sync framework employs delta compression for data packaging, reducing the data volume by 73% compared to full uploads. On a 2G network, a single synchronization of weekly class data takes 2–5 minutes.

C. Bandwidth-Based Dynamic Features

Application detects bandwidth and switches:

- Video resolution (from 144p to 480p)
- Content pre-caching for lessons when on a strong network connection
- Frequency of data sync (more frequent if connected to strong networks)

3.5. Teacher Empowerment Dashboard

As teachers are the single most important factor in educational outcomes, we designed a dashboard specifically for low-end Android devices with offline capabilities.

A. Features

- Overview of class, highlight under-performing students and concepts they are weak in
- Actionable advice on potential interventions (small group work, 1 on 1), tailored to the teacher by AI
- Dashboard will work offline, using the last synced data
- Critical thinking alerts will be pushed via SMS to teachers without data or smartphones
- Reports will be mapped to Kenya's national KB-Curriculum

B. Dashboard Usability Results

- We tested out the dashboard with 24 STEM teachers in Kenya.
- 92% Dashboard was “easy to use” following a 2-hour training session.
- 87% Allowed them to better identify struggling students sooner.
- 78% Liked that they could use it during network outages (thanks to offline capability)
- Average time to check on class performance: 7mins/week

4. Results and Discussion

The developed system was tested using a mixed-methods research approach in twelve secondary schools in Kenya between January and August 2025. The study design was quasi-experimental with six intervention schools consisting of three rural schools under resource constraint and three urban schools as controls, and six control schools taught with traditional methods (three rural and three urban).

4.1. Experimental Setup

Table 4. School characteristics in study sample

Characteristic	Intervention Schools (n=6)	Control Schools (n=6)
Rural schools	3	3
Urban schools	3	3
Average students per school	487	512
Student-teacher ratio (STEM)	58:1	55:1
Schools with reliable electricity	4 (67%)	5 (83%)
Schools with internet access	3 (50%)	4 (67%)
Average devices per school	23 tablets	N/A

Table 5. Computing Environment for Model Training and Evaluation

Parameter	Specification
Training Platform	Google Colab Pro
GPU	NVIDIA Tesla T4 (16 GB)
Programming Language	Python 3.9
Deep Learning Framework	TensorFlow 2.15 / Keras
Edge Runtime	TensorFlow Lite 2.15
Random Seed	42
Training Epochs	50 (with early stopping)
Batch Size	32

4.2. Learning Outcomes

A. STEM Concept Mastery

Participants took achievement tests mapped to KCSE STEM goals. Achievement tests were administered at baseline (before intervention), midpoint (4 months), and endpoint (8 months).

Table 6. Pre-and post-intervention STEM assessment (%) scores

Group	Pre-test Mean (SD)	Mid-test Mean (SD)	Posttest Mean (SD)	Gain	Effect Size (Cohen's d)
Intervention Rural	41.3 (8.7)	54.2 (9.1)	62.8 (8.9)	+21.5	2.47
Intervention Urban	48.9 (7.8)	63.1 (8.2)	71.4 (7.6)	+22.5	2.89
Control Rural	42.1 (9.2)	45.3 (9.8)	48.7 (10.1)	+6.6	0.72
Control Urban	49.3 (8.1)	52.8 (8.7)	56.2 (9.3)	+6.9	0.85

The intervention group students achieved a mean of 31.0% improvement, whereas the control group students' average improvement was only 13.5% ($p < 0.001$, paired t-test). We further tested a multilevel linear mixed-effects model with random intercepts for schools and students to consider school clustering effects. The intervention effect was still statistically significant ($\beta = 15.8$, $SE = 2.1$, $p < 0.001$, 95% CI [11.7, 19.9]) when controlling for school-level effects. The intra-cluster correlation at the school level was $ICC = 0.23$, indicating a moderate level of clustering effect. Remarkably, rural school children in the intervention group had an average post-test score (62.8%), which was only slightly lower than the average pre-test score for urban school children in the control group (49.3%), indicating scalability and possibility for bridging urban-rural disparity in education.

B. Subject-wise Analysis

- Mathematics: +24.3% vs. +7.1%
- Physics: +29.7% vs. +5.8%
- Chemistry: +27.1% vs. +6.2%
- Biology: +19.4% vs. +8.1%

Physics showed the greatest improvement among all subjects. This can be attributed to high student engagement because of interactive physics simulations. Due to a lack of lab equipment in intervention schools, these simulations bridged the much-needed gap. Correlation analysis indicated a substantial positive relationship between the frequency of simulation use and the gain in physics performance ($r = 0.42$, $p < 0.01$). The group of students that used simulation more than 3 times/week ($n=87$) scored an average performance gain of 34.2% ($SD=8.1$), whereas the less frequently simulating group ($n=156$) gained on average 21.8% ($SD=7.6$).

4.3. Student Engagement and Retention

A. STEM Subject Dropout Rates

We next looked at how many students dropped out of the STEM stream. Dropout was defined as students switching to a non-STEM stream in the subsequent term.

Table 7. STEM subject dropout rates (%)

Group	Term 1	Term 2	Term 3	Overall Reduction	n (students)	95% CI
Intervention	11.2%	8.4%	5.7%	49.1%	487	[4.2%, 7.2%]
Control	10.8%	12.3%	14.1%	-30.6% (increase)	512	[12.3%, 15.9%]

AL's absolute reduction of dropouts, by 27% (14.1% vs 5.7%), points to its success in retaining students' interest in learning in the STEM domains despite the challenging circumstances faced by this population group. The chi-square test showed that the association between the intervention and the rate of school dropouts is significant ($\chi^2=24.3$, $df=1$, $p<0.001$).

B. Learner Engagement Analytics

Analysis of system logs showed high levels of engagement with the platform:

- Students used the system 4.2 times every week on average
- The length of time spent per session was 27 minutes
- The students completed 83% of the lessons they started (61% in prior ed-tech initiatives)
- The majority of the logins were between 6 and 8 AM and 4-6 PM (adjusted for time spent on domestic activities)

4.4. Teacher Impact

Analysis of educator surveys and classroom observations showed:

1. Educators saved time on administrative tasks such as grading and taking notes on each student's progress, by an absolute value of 43%, which could be allocated to bespoke instruction.
2. 89% of educators were able to notice learning gaps at an earlier stage (average of 3.2 weeks before expected time)
3. There was a significant shift away from teacher-centered towards learner-centered activities within the classrooms, from 22% to 47% of observed time. This shift was calculated using a validated observational tool, which had been modified from the Classroom Observation and Reflection Tool (CORT). 48 classroom observations of 30 minutes duration were completed by two trained coders (inter-rater reliability $\kappa = 0.84$). During observations, time was categorized as: teacher-centered (lectures, whole class instruction, teacher demonstrations); learner-centered (group work, individual learning, peer teaching, simulations) or management/transitions. The substantial change towards increased learner-centered activities, $\chi^2 = 18.7$, $df = 1$, $p < .001$, indicated that there was increased opportunity for active learning in the classroom.
4. More than 80% of the participating teachers said that they were willing to use the learning platform in the future after the learning activities were finished.

Table 8. Educator end-of-study survey responses (n=48 educators)

Statement	% Agree	95% CI
"It helps me recognise which students need help."	94%	[90.1%, 97.9%]
"It reduces my paperwork load"	87%	[81.2%, 92.8%]
"Students are spending more time excited about STEM."	91%	[86.1%, 95.9%]
"It was easy to pick up and use, and required little training."	82%	[74.8%, 89.2%]
"I would recommend this product to other schools."	96%	[93.0%, 99.0%]

4.5. Technical Metrics

A. Prediction Accuracy

The Lightweight Neural Network model was able to predict student success at 89.1%. This was only 2% less accurate than full-sized models at 91.2% while reducing the memory footprint by 96%.

B. Offline functionality

Schools participating in our study averaged 3.2 days per week during which their internet connection was unreliable. Results from testing our offline-first capabilities show:

- Users were able to access 100% of the content during outages
- 94% of items successfully synced once connectivity returned
- Average synchronization time was 4.7 minutes on 2G networks. 1.2 minutes on 3G/4G networks.

C. Device compatibility

We successfully ran our app on the following devices:

- Samsung Galaxy Tab A (2019) - 2GB RAM
- Tecno Camon 12 - 3GB RAM
- Nokia 2.4 - 2GB RAM
- Raspberry Pi 4 - (school server)

The average energy usage was 12% battery per hour of use. This allows students to use the platform all day on school-provided tablets.

D. Ablation Study

Ablation experiments were designed and implemented to clearly examine the performance contributions of each compression method – pruning, quantization, and knowledge distillation, respectively – by disabling them one by one.

Table 9. Ablation Study Results for Compression Techniques

Configuration	Accuracy	Size (MB)	Δ Accuracy
Baseline (no compression)	91.2%	47.3	—
Pruning only	90.1%	28.4	-1.1%
Quantization only	89.4%	5.9	-1.8%
Pruning + Quantization	88.7%	2.8	-2.5%
Pruning + Quantization + Distillation	89.1%	1.9	-2.1%

The ablation study reveals that knowledge distillation achieved the best trade-off, recovering 0.4% of the original accuracy at 32% reduction of size over pruning+quantization.

4.6. Comparison with Prior Work

Table 10. Performance against existing products

Approach	Learning Gain	Cost per Student	Infrastructure Required	Scalability
Traditional Teaching	Baseline	Low	None	High
E-Content Libraries [26]	+8-12%	Medium (\$15-25)	Devices + occasional internet	Medium
SMS-based Learning [27]	+5-8%	Low (\$2-5)	Basic phones	Very High
Cloud Adaptive Platforms [28]	+25-30%	High (\$50-100)	Continuous internet, high-end devices	Low
Proposed System	+31%	Low (\$18-25)	Low-end devices, intermittent internet	High

4.7. Cost-Effectiveness

Total cost per student over 3 years:

- Hardware (tablet shared at a ratio of 5 students): \$30
- Software licensing: \$12
- Teacher training: \$3
- Maintenance: \$5
- TOTAL: \$50 (\$16.67 per year)
- \$1.61 per % of learning gain (compared to \$3.84 cost for learning gains with traditional education interventions, e.g., textbooks + teacher training in Kenya)

4.8. Qualitative Results

72 students and 24 teachers participated in focus group discussions. Highlights:

Student Responses:

- “I like that the app gives me questions at my level. Before, I was just bored or totally lost.” (Form 2 student, rural school)
 - “The simulations help me learn physics, even though our school can’t afford a lab.” (Form 3 student)
 - “I’m able to learn even when there is no internet connection at my house.” (Form 1 student)
- Teacher Responses:
- “I now know exactly which students don’t understand fractions. Before, I would only find out in the exams.” (Mathematics teacher)
 - “The recommendations help me organize students into groups based on common difficulties.” (Physics teacher)
 - “My only concern is with the tablets breaking - several broke during our time in the study.” (School administrator)

4.9. Limitations

- Hardware Fragility: 8 tablets (5.8%) were damaged and needed repairs during the study period.
- Solar Charging: During several weeks of very cloudy weather, some rural schools had difficulty charging the tablets.
- Teacher Pushback: Teachers were 12% hesitant about the program at first, though this decreased to 4% after the study period of 3 months.
- Cultural References: Some examples in the STEM subjects needed to be more relevant to Kenyan culture.
- Scaling Challenges: Ensuring the quality of Edutoobs as the program scales nationwide will require continuous training of teachers.

4.10. Algorithm Performance Under Extreme Low-Data Scenarios

The degradation of the algorithm in low student interaction schools was analysed. With the interaction per student below 10 interactions, N=3 schools with 87 students predicted accuracy of the algorithm was 71.3% which compared to the predicted accuracy using high volumes (>100) of student interactions as 89.1% which means an degradation of 17.8% on the model, however for the first 5 decisions during the exploration phase of the RL agent the content selection accuracy was equivalent and afterwards reduced as a result of the adaptivity. Population-based transfer learning from an IRT model (with a 68.5% predictive accuracy at first interaction) was used to resolve cold-start behaviour.

4.11. Teacher Acceptance and Long-Term Tracking

Post-study teacher acceptance rate was high (96% would recommend to other schools), though initially 12% of teachers experienced some reluctance that fell to 4% after three months. Beyond the study’s duration of eight months, long-term monitoring wasn’t possible; yet 84% still used the platform, which would seem indicative of continued long-term acceptance. The main elements influencing teacher acceptance were: 1)Ease of use: 82% agree; 2)Time-saving: 87% agree; and 3)Student identification was more effective: 94% agree.

4.12. Algorithmic Generalization Capability

The generalized performance of the system in different regions of Kenya was investigated by measuring it in urban and rural intervention schools. Gains in rural school students ($\Delta = +21.5\%$, $d = 2.47$) are slightly lower than in urban school students ($\Delta = +22.5\%$, $d = 2.89$), though they still prove to be highly significant. Different school subjects produced varied effects, with physics having the largest increase in gains ($\Delta = +29.7\%$) while biology had the lowest gain ($\Delta = +19.4\%$), and this implies that although the algorithm generalizes across contexts, it can benefit even more from interactive simulations on experimental subjects.

5. Conclusions

An AI-powered adaptive learning system that allows students in developing countries to learn STEM topics at their own pace was developed and evaluated. The solution was developed for the local conditions in Kenyan schools: it was built to support low-cost devices and operate fully offline, with connectivity used only for synchronization whenever it becomes available. The system is powered by an adaptive AI learning engine, consisting of an optimized Reinforcement Learning algorithm to recommend content, and a TinyML model to predict the student’s performance. In a 12-school pilot study in Kenyan secondary schools, the impact of the system was measured.

Students studying STEM topics showed a 31% improvement in their understanding of STEM concepts ($p < 0.001$, $d = 2.47$), a 27% absolute decrease in dropout rates from STEM courses ($\chi^2 = 24.3$, $p < 0.001$), and the productivity of teachers increased by 43%. All this was delivered on Android devices costing under \$150, using a compressed ML model that achieved 89% accuracy in predicting performance while requiring less than 2MB of memory. These results were obtained through the use of model compression techniques-pruning, quantization, and knowledge distillation-to deploy

machine learning on edge devices in low-resource environments at scale. Human-centric design played a vital role in the success of the project: this includes (1) working together with Kenyan teachers to build the system, (2) accommodating infrastructure limitations with an offline-first design, (3) enhancing teachers' abilities instead of replacing them, and (4) building a cost-effective system accessible with the limited school budgets in Kenya.

The study demonstrates that AI-powered personalized learning can be and is appropriate in low-resource contexts. These technologies can be customized for such environments with mindful design to address the digital divide. While this research shows great promise, further research in various contexts and over longer periods is needed for broader generalizability. The design principles, nevertheless, provide a clear framework for scaling this solution to other developing regions.

5.1. Future Work

Future work will focus on the following directions with milestones:

Phase 1 (0-12 months): Scaling and Localization

- Taking the solution to all 47 Kenyan counties and providing a how-to guide to the Kenyan Ministry of Education to be adopted more broadly.
- Translating the system content to local languages such as Swahili, Kamba, Kikuyu, and Luo to attract younger children.

Phase 2 (12-24 months): Enhanced Analytics and Peer Learning

- Using dropout prediction models to run six months of data, providing an early warning system for parents and students.
- Facilitating student-to-student peer learning by suggesting high-achieving students be tapped as peer tutors.
- Extending the models beyond secondary school to early grade learners and technical and vocational education.

Phase 3 (24-36 months): Technical Optimization and Policy

- Improving the model for use in an ultra-low-power, offline environment for those students who can only rely on solar for power in regions outside of electrical grids.
- Collaborating with the Kenyan government on developing a national policy for the acquisition and usage of artificial intelligence in education.

Evaluation Metrics for Future Work:

For each phase, the impact assessment will be assessed with: (1) Student learning gains (pre vs post-test), (2) System adoption and usage levels, (3) Teacher satisfaction survey, (4) Technical metrics (sync levels, model quality, etc.), and (5) Economic analysis.

All the Declarations and Statements

Author Contributions Statement

David Shiala Ongoma - Conceptualization, Methodology, Formal Analysis, Investigation, Data Curation, Writing – Original Draft Preparation, Writing – Review & Editing, Visualization, Supervision, Project Administration, Funding Acquisition: Conceptualised the research, architected the adaptive AI learning system, developed a tailored reinforcement learning algorithm and light neural network model, guided field implementation, analyzed empirical results, and prepared manuscript.

Conflict of Interest Statement

No competing interests to disclose.

Funding Declaration

No grant funding was provided for this research from any public, private sector, or non-profit funding agency. Hardware used was from Zetech University through their institutional support.

Data Availability Statement

All datasets used and/or analyzed during the current study are available from the corresponding author on reasonable request. Anonymized student interaction data and anonymized and aggregated student results are available for research purposes following institutional ethics committee approval.

Ethical Declarations

The author declares that the research was conducted in compliance with good research practice guidelines. The study

obtained approval from the Zetech University Research Ethics Committee. All the selected schools, teachers, and the parents/guardians of the selected pupils provided informed consent in writing. All student data was stripped and de-identified before analysis. The research followed the principles of the Declaration of Helsinki. The author also declares that the submitted work is original and has not been published nor is it under review for publication in any other journal.

Acknowledgments

The author thanks the institutions, schools, teachers, and students involved in the study who generously contributed their time to the research. Appreciation goes to Zetech University for its support as an institution, to the Kenya Ministry of Education for facilitating access to the participating schools, and finally, to the referees whose comments helped enhance the manuscript.

Declaration of Generative AI in Scholarly Writing

While preparing this manuscript, the author employed generative artificial intelligence (AI) tools for tasks such as language correction, grammar enhancement, format generation, and improving the readability of the manuscript. All of the outputs from the AI tool have been reviewed and edited by the author. The author is accountable for the accuracy, novelty, and integrity of the manuscript submitted in its entirety.

Abbreviations

The following abbreviations are used in this manuscript:

Abbreviation - Full Form
 AI - Artificial Intelligence
 BKT - Bayesian Knowledge Tracing
 CBC - Competency-Based Curriculum
 DKT - Deep Knowledge Tracing
 DQN - Deep Q-Network
 GPU - Graphics Processing Unit
 HSKT - Hybrid Sparse Knowledge Tracing
 ICT - Information and Communication Technology
 IRT - Item Response Theory
 KCSE - Kenya Certificate of Secondary Education
 LNN - Lightweight Neural Network
 ML - Machine Learning
 M-RL - Modified Reinforcement Learning
 RAM - Random Access Memory
 ReLU - Rectified Linear Unit
 RL - Reinforcement Learning
 SSA - Sub-Saharan Africa
 STEM - Science, Technology, Engineering, and Mathematics
 TCN - Temporal Convolutional Networks
 TP - True Positive
 TN - True Negative
 FP - False Positive
 FN - False Negative
 FN - False Negative

Appendix A/B/C..., with appendix title

Appendix A. Sample Lesson Content Format Generation Rules

Content Format	Generation Rule
Video	Pre-recorded lessons (5-10 min) covering a single concept
Interactive Simulation	Template-based physics engine with adjustable parameters
Article	Text-based explanation with embedded questions
Podcast	Audio explanation of the concept
Content Format	Generation Rule
Video	Pre-recorded lessons (5-10 min) covering a single concept
Interactive Simulation	Template-based physics engine with adjustable parameters
Article	Text-based explanation with embedded questions
Podcast	Audio explanation of the concept
Content Format	Generation Rule

Appendix B. Teacher Dashboard Intervention Recommendations

Student Risk Level	Recommended Intervention
High Risk	1-on-1 teacher session, remedial content
Medium Risk	Small group work (3-5 students), additional practice
Low Risk	Peer tutoring, advanced content
Student Risk Level	Recommended Intervention
High Risk	1-on-1 teacher session, remedial content
Medium Risk	Small group work (3-5 students), additional practice
Low Risk	Peer tutoring, advanced content

References

- [1] UNESCO, "Global Education Monitoring Report: Technology in Education," *UNESCO Publishing*, Paris, 2023. Available at: <https://www.unesco.org/gem-report/en/technology>
- [2] World Bank, "Facing Forward: Schooling for Learning in Africa," *World Bank Group*, Washington DC, 2024. <https://www.worldbank.org/en/region/afr/publication/facing-forward-schooling-for-learning-in-africa>
- [3] Kenya National Bureau of Statistics, "Economic Survey 2024," *Government Printer*, Nairobi, 2024. PDF: <https://www.knbs.or.ke/wp-content/uploads/2024/05/2024-Economic-Survey.pdf>.
- [4] J.A. Banks, "Networking as a Strategy for Improvement of Science and Mathematics Teaching and Learning in Kenyan Secondary Schools," *Journal of Education and Practice*, Jul. 2019. DOI: 10.7176/jep/10-21-16
- [5] Kenya Institute of Curriculum Development, "Competency-Based Curriculum Framework," *KICD*, Nairobi, 2022. Available: <https://kicd.ac.ke>
- [6] J.A. Banks, "Digital Divide, Digital Learning, and Equal Access," *Encyclopedia of Diversity in Education*, 2012, DOI: 10.4135/9781452218533.n201.
- [7] P. Amolloh Odundo and C. Richard Oyier, "Influence Of Policy Framework On Budgeting For Science Instructional Resources In Kenyan Secondary Schools," *Archives of Business Research*, vol. 5, no. 4, Apr. 2017, doi: 10.14738/abr.54.2944.
- [8] Dr. Priyanka Datta, "Artificial Intelligence and Education," *Artificial Intelligence in Education: Revolutionizing Learning And Teaching*, Oct. 2020, doi: 10.25215/9358094575.32.
- [9] Y. Gao, "A Meta-Analysis of Adaptive Learning Systems on Learning Achievement (Poster 16)," *Proceedings of the 2024 AERA Annual Meeting*, 2024, doi: 10.3102/2105897.
- [10] Carnegie Learning, "MATHia Research and Efficacy Studies," 2023. Available: <https://www.carnegielearning.com>
- [11] D. I. Okonkwo and E. A. Ade-Ibijola, "Chatbots Applications in Education: A Systematic Review," *Comput. Educ. Artif. Intell.*, vol. 2, 2021. DOI: 10.1016/j.caeai.2021.100033
- [12] Communications Authority of Kenya, "Sector Statistics Report Q4 2024," *CAK*, Nairobi, 2025. Available: <https://www.ca.go.ke>
- [13] J. Wishart, "Current and recent mobile learning teaching and learning initiatives in secondary schools," *Mobile Learning in Schools*, pp. 36–49, Oct. 2017, doi: 10.4324/9781315536774-4.
- [14] A. Ade-Ibijola and C. Okonkwo, "Artificial Intelligence in Africa: Emerging Challenges," in *Social and Cultural Studies of Robots and AI*, Springer, 2023, doi: 10.1007/978-3-031-08215-3_5.
- [15] Tondeur et al., "Preparing Pre-Service Teachers to Integrate Technology in Education: A Synthesis of Qualitative Evidence," *Comput. Educ.*, vol. 59, no. 1, pp. 134–144, 2012. DOI: 10.1016/j.compedu.2011.10.009
- [16] UNESCO, "Cost-Effectiveness of Educational Technology in Low-Income Countries," *IIEP Research Paper*, Paris, 2024. Available: <https://www.iiep.unesco.org>
- [17] Khan Academy, "Personalized Learning: Technical Architecture and Impact," *Khan Academy Technical Report*, 2023. Available: <https://www.khanacademy.org>
- [18] J. R. Anderson, et al., "Cognitive Tutors: Lessons Learned," *Journal of the Learning Sciences*, vol. 4, no. 2, pp. 167–207, 1995. DOI: 10.1207/s15327809jls0402_2
- [19] A. Mutua Nzioki, "Improved Learning Management System (i-LMS): A Flat Form for Content Creation and Sharing for Kenyan Secondary Schools," *International Journal of Computer Applications Technology and Research*, vol. 4, no. 2, pp. 86–91, Feb. 2015, doi: 10.7753/ijcatr0402.1001
- [20] DreamBox Learning, "Intelligent Adaptive Learning Engine White Paper," *DreamBox*, Bellevue, WA, 2024. Available: <https://www.dreambox.com>
- [21] Y. Shi, E. Yang, and J. Wu, "Edge Computing: Vision and Challenges for Internet of Things Applications," *IEEE Internet Things J.*, vol. 3, no. 5, pp. 637–646, 2016. DOI: 10.1109/JIOT.2016.2579198
- [22] C. Piech, et al., "Deep Knowledge Tracing," *Advances in Neural Information Processing Systems*, vol. 28, pp. 505–513, 2015. DOI: 10.48550/arXiv.1506.05908
- [23] A. Ayari, M. Chaabouni, and H. Ben Ghezala, "A Deep Learning Approach for Automatic Detection of Learner Engagement in Educational Context," *Proceedings of the 17th International Conference on Computer Supported Education*, pp. 372–379, 2025, doi: 10.5220/0013283200003932.
- [24] V. Mnih, et al., "Human-level Control through Deep Reinforcement Learning," *Nature*, vol. 518, pp. 529–533, 2015. DOI: 10.1038/nature14236
- [25] B. McMahan, et al., "Communication-Efficient Learning of Deep Networks from Decentralized Data," *Proceedings of AISTATS*, 2017. DOI: 10.48550/arXiv.1602.05629
- [26] Kenya Education Cloud, "E-Content Utilization Report 2024," *KICD*, Nairobi, 2024. Available: <https://kec.ac.ke>
- [27] Eneza Education, "Impact Report: SMS-based Learning in Kenya," *Eneza Education*, Nairobi, 2023. Available: <https://enezaeducation.com>

- [28] CenturyTech, "Adaptive Learning in Emerging Markets: Pilot Results," *Century Tech*, London, 2024. Available: <https://www.century.tech>
- [29] A. K. Soma, "Enhancing Supply Chain Transparency and Integrity: A Permissioned Blockchain Framework," in *2025 International Conference on Emerging Systems and Intelligent Computing (ESIC)*, pp. 819-826, 2025. doi: 10.1109/esic64052.2025.10962720
- [30] A. kumar Soma, "Hybrid RNN-GRU-LSTM Model for Accurate Detection of DDoS Attacks on IDS Dataset", *Journal of Modern Technology*, vol. 2, no. 01, pp. 283-291, 2025. doi: 10.71426/jmt.v2.i1.pp283-291
- [31] P. K. Udayaprasad, et al., "Energy Efficient Optimized Routing Technique with Distributed SDN-AI to Large Scale I-IoT Networks," *IEEE Access*, vol. 12, pp. 2742-2759, 2024. doi: 10.1109/access.2023.3346679
- [32] C. L. D. Santos, et al., "ML-RPL: Machine Learning-Based Routing Protocol for Wireless Smart Grid Networks," *IEEE Access*, vol. 11, pp. 57401-57414, 2023. DOI: 10.1109/ACCESS.2023.3283208
- [33] M. A. Islam, R. Atat and M. Ismail, "Software-Defined Networking-Based Resilient Proactive Routing in Smart Grids Using Graph Neural Networks and Deep Q-Networks," *IEEE Access*, vol. 12, pp. 111169-111186, 2024. doi: 10.1109/access.2024.3438938

Authors' Profiles



David Shiala Ongoma holds a Bachelor's degree in Computer Science from Dedan Kimathi University of Technology, a Master's degree in Computer Science from Jomo Kenyatta University of Agriculture and Technology, where he is currently pursuing a PhD in Computer Science. He currently works as an ICT Lecturer at Zetech University, Nairobi, Kenya, where he is a member of the faculty in the Department of ICT, Media, and Engineering. He's also a part-time ICT Lecturer at Pan African Christian University and Jomo Kenyatta University of Agriculture and Technology. His research interests include Quantum Computing, Artificial Intelligence, Internet of Medical Things, EdTech, Cybersecurity for healthcare systems, Blockchain-based data provenance, Federated Learning, and the security implications of large language models in clinical settings.

How to cite this paper

David Shiala Ongoma, "Edge-First Adaptive Learning with Lightweight RL and LNN for STEM Education in Low-Resource Kenyan Schools", *International Journal of Education and Management Engineering (IJEME)*, Vol.16, No.4, pp. 71-88, 2026. DOI:10.5815/ijeme.2026.04.06.