Can Less Be More? Benchmarking Lightweight Models Against State-of-the-Art Deep Learning Architectures for Deployable Seizure Detection

Isaiah Essien[1,*] Donna-lee Ginsberg[2,†] Jesse Thornburg[3,†]
[1]African Leadership University, Kigali, Rwanda [2]GeoPoll, Panama City, Panama
[3]College of Engineering, Carnegie Mellon University, Kigali, Rwanda
[1]i.essien@alustudent.com [2]donna-lee@geopoll.com [3]jthornbu@andrew.cmu.edu

Over the past decades, emerging research in seizure detection has highlighted the critical need for resource-constrained, deployable models that can operate in low-infrastructure environments. Seizure detection models that achieve high accuracy on benchmarks rarely run on the hardware available in low-resource contexts like developing countries, where epilepsy takes the heaviest toll. This work addresses the fundamental disconnect between model performance and real-world deployability by developing and evaluating parsimonious deep learning architectures for real-time epileptic seizure detection on consumer smartphones. This study systematically develops and compares two lightweight models: a Convolutional Neural Network with Gated Recurrent Units (CNN-GRU) and a 1D Convolutional Network with Multi-Head Attention (1D CNN-MHA). The optimal model is selected for both detection performance and deployment feasibility. The parsimonious 1D CNN-MHA model achieved superior performance with 96% accuracy, 93% sensitivity, and 0.99 AUC, outperforming the CNN-GRU model in both accuracy and sensitivity. Benchmarking against state-of-the-art models reveals a persistent deployment gap: while "lightweight" models in the literature lack deployment evidence, and high-accuracy models are bound to server-grade hardware, the 23.8 KB TensorFlow Lite model bridges this gap by delivering competitive accuracy while running in real-time on mid-range Android devices. Crucially, these results establish deployment feasibility rather than clinical validity: the system demonstrates that seizure-like motion patterns can be reliably discriminated under strict on-device constraints using commodity smartphones. The findings therefore support the principle that carefully designed parsimonious architectures can approach the performance of heavier models while remaining executable in real-world edge environments. This work can be interpreted as a feasibility study of deployability designed to enable subsequent large-scale clinical validation rather than as a population-level diagnostic model.

*Corresponding author.
†These authors contributed equally as second and third authors.

Third Conference on Parsimony and Learning (CPAL 2026).

1. Introduction

Epilepsy affects more than 50 million people worldwide, with more than 75% of cases in low- and middle-income countries (LMICs) where access to diagnosis and continuous monitoring is severely limited [1]. The consequences are dire: Sudden Unexpected Death in Epilepsy (SUDEP) claims thousands of lives annually, disproportionately in regions without sufficient infrastructure for neurological care [2]. Although mobile health technologies offer a potential pathway to bridge this gap, many proposed solutions remain fundamentally impractical for the resource-constrained settings where they are most needed [3, 4].

A critical technological barrier persists. State-of-the-art deep learning models for seizure detection, particularly transformers and complex recurrent networks, achieve high accuracy in controlled settings [5–7]. However, their computational complexity and reliance on operations incompatible with mobile inference engines, such as TensorFlow Lite, render them unusable on the mid-range smartphones that predominate (compared with higher-performing devices) in underserved regions [8]. This creates a fundamental disconnect between model performance in research and deployability in reality.

This work addresses this disconnect through the principle of parsimonious design. This study argues that for a life-critical application like seizure detection, model efficiency cannot be an afterthought; it must be the foundational design constraint. In this work, two deep learning architectures are systematically developed and evaluated—a Convolutional Recurrent Neural Network with Gated Recurrent Units (CNN-GRU) and a Convolutional Network with Multi-Head Attention (CNN-MHA). The optimal model is selected based on both detection performance and parsimony, with deployment feasibility as the leading factor in the latter. The approach demonstrates that strategic architectural choices can maintain high detection accuracy while ensuring compatibility, low latency, and minimal resource consumption on consumer-grade Android devices. By benchmarking the final model against both heavyweight state-of-the-art models and other "lightweight" claimants, this study establishes parsimony not as a performance compromise, but as a prerequisite for real-world impact in global neurology [9–11]. It is important to clarify that the objective of this study is not to claim clinical readiness or population-level seizure detection performance, but to evaluate whether a modern deep learning architecture can operate reliably under the strict computational constraints imposed by commodity smartphones in low-resource environments. Accordingly, the work should be interpreted as a feasibility investigation of deployable edge intelligence rather than a clinical validation study. Establishing technical viability is a necessary prerequisite for subsequent large-scale subject-independent trials.

2. Related Work

This work is situated at the intersection of clinical epilepsy care, mobile health (mHealth) implementation science, and efficient deep learning design. A comprehensive analysis of these domains reveals a consistent gap between algorithmic ambition and deployment feasibility.

Clinical Need and mHealth Context. The staggering treatment gap for epilepsy in LMICs is well-documented [3, 12]. mHealth solutions, including wearable sensors and smartphone-based monitoring, have emerged as promising avenues to address this [13, 14]. Commercial devices like the Empatica Embrace and various research prototypes demonstrate the technical feasibility of automated seizure detection [15, 16]. However, systematic reviews consistently highlight that cost, infrastructure, and technical complexity remain formidable barriers to widespread adoption in low-resource settings [17, 18]. The challenge, therefore, is not merely to detect seizures but to do so within the severe constraints of the target environment.

The Computational Overhead of State-of-the-Art Models. The deep learning community has made significant strides in seizure detection accuracy. Transformer-based architectures have shown exceptional performance by leveraging self-attention mechanisms to model long-range dependencies in physiological signals. For instance, [7] combined the Stockwell Transform with a multi-

layer Transformer, achieving 96.15% accuracy, but required an NVIDIA GeForce RTX 3050 GPU for inference, making it a server-bound solution. Similarly, [6] employed a Multi-Channel Vision Transformer, explicitly acknowledging its "high computational cost" and unsuitability for resource-constrained systems. The recent trend of hybridizing heavy architectures, such as the DenseNet-ViT model by [19], further exacerbates this issue, stacking computationally expensive components and moving further from edge deployment.

The "Lightweight" Fallacy and the Deployment Gap. In response to these heavy models, a recent wave of literature claims to offer "lightweight" or "wearable-ready" alternatives. Zhang et al. present a remarkably small 21.32 KB CNN, yet provided no evidence of its operation on an actual wearable device [9]. Rukhsar and Tiwari propose a "Lightweight Convolution Transformer" but omit reporting the model size and required 18 EEG channels, a setup impractical for a consumer wearable [10]. Most recently, Li et al. offer a "lightweight" and "real-time" hybrid model but provided no deployment metrics, inference times, or evidence of mobile compatibility [11]. These works exemplify a "deployment gap," where efficiency claims remain theoretical, untested on the hardware and in the environments they purport to serve.

The Position: Parsimonious Design. In contrast to the above and to address the gaps identified, this study advocates for parsimony as a first-order design principle. This means architecting models that are intrinsically efficient yet less complex, reproducible, and deployable, avoiding operations that hinder platform-specific deployments, such as mobile deployment (e.g., certain recurrent layers requiring Flex delegates), and rigorously validating not just on benchmark datasets but on target hardware with real-world data. The approach is grounded in the understanding that for a solution to be viable in LMICs, it must be available for use on go-to devices, such as smartphones already in the hands of the population, without reliance on cloud infrastructure or specialized hardware.

In Section 4, the parsimonious best model will be benchmarked against the performance claims of these state-of-the-art and "lightweight" models, evaluating not only accuracy and sensitivity but, crucially, the practical metrics of model size, inference latency, and demonstrated on-device performance that define true deployability.

3. Methodology

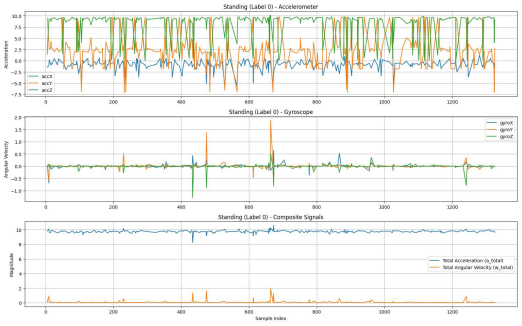
3.1. Data Collection and Exploratory Analysis

The inertial data used in this work was collected using a custom-built mobile application developed in Flutter. The application provided an intuitive interface that allowed users to simulate or record specific activities in real time while capturing motion sensor data from the phone's built-in accelerometer and gyroscope. The app enabled activity labeling at the time of recording, with options for standing, walking, jumping, and seizure-like movements. This simple yet effective design eliminated the need for external wearables and enabled controlled data collection across multiple activity types. Internally, the application subscribed to motion sensor streams and logged six channels of data, three from the accelerometer and three from the gyroscope, alongside timestamps and activity labels, which were saved in CSV format and could be directly shared or stored on the device.

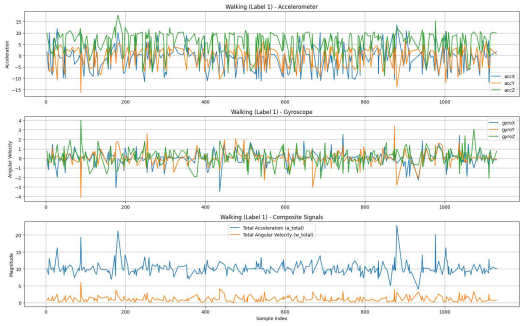
Data was gathered from seven individuals, including three consenting epilepsy patients who permitted recordings during seizure episodes, and four healthy volunteers who mimicked seizure-like behavior based on clinical references. The recordings captured naturalistic movement, with the phone held in hand or placed in the pocket, replicating real-world scenarios. This method yielded a dataset comprising over 35,000 labeled entries. The diversity in user profiles, device placements, and movement intensities introduced meaningful variability to the dataset, an intentional design to improve model generalizability.

The goal of exploratory analysis was to identify distinguishable patterns across the four activity classes: standing, walking, seizure-like movements, and jumping. These activities were deliberately selected to include both similar motion patterns (walking vs. seizure-like) and distinct behaviors

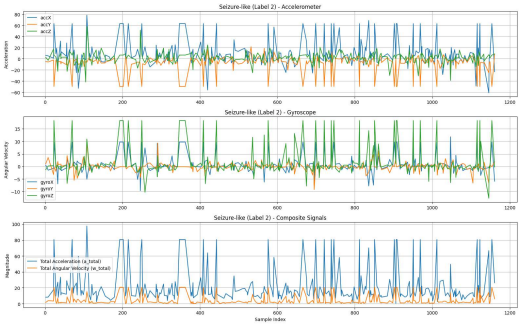
(standing vs. jumping) to challenge the model and enhance robustness. By including activities that could potentially confuse a naive classifier, the model is forced to learn more discriminative features, ultimately improving its real-world performance.



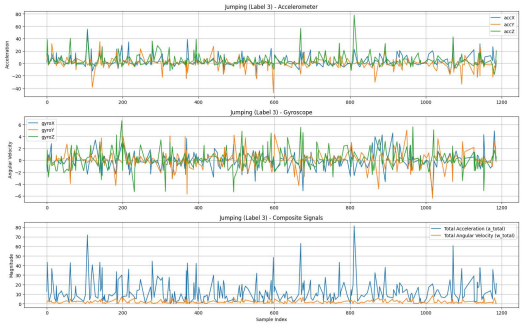
(a) Standing Activity



(b) Walking Activity



(c) Seizure-like Activity



(d) Jumping Activity

Figure 1: Comparative analysis of sensor signals across four activity classes. Each panel shows accelerometer signals (top), gyroscope signals (middle), and composite signals (bottom). Composite signals were computed using Equation 1.

The composite signals shown in the bottom panel of each subfigure were computed using the magnitude of the three-axis sensor readings:

$$a_{\text{total}} = \sqrt{\text{accX}^2 + \text{accY}^2 + \text{accZ}^2}, \quad w_{\text{total}} = \sqrt{\text{gyroX}^2 + \text{gyroY}^2 + \text{gyroZ}^2} \tag{1}$$

Analysis of the signals reveals distinct patterns across activities: Standing (Figure 1a) shows minimal fluctuations around gravitational acceleration ($9.8\,\text{m/s}^2$), with gyroscope readings near zero, indicating stable device orientation. Composite signals remain flat and low, confirming minimal movement. Walking (Figure 1b) exhibits rhythmic, periodic patterns particularly evident in the Z-axis, corresponding to gait cycles. Gyroscope signals show consistent small rotations during steps. Composite acceleration (a_{total}) displays regular peaks between 7-12 m/s^2, while angular velocity (w_{total}) shows minor fluctuations, characteristic of natural walking motion. Seizure-like movements (Figure 1c) display irregular, high-amplitude fluctuations across all axes. Composite signals show extreme peaks exceeding 30 m/s^2 in linear acceleration and 25 rad/s in angular velocity, with chaotic patterns distinct from rhythmic activities like walking. Jumping (Figure 1d) is characterized by sharp, high-magnitude spikes in accelerometer signals during takeoff and landing phases. Gyroscope signals show significant rotational changes during airborne phases. Composite acceleration shows intermittent high peaks separated by periods of minimal activity. This comparative analysis confirms that while some activities share superficial similarities, each exhibits unique spatiotemporal signatures that a well-designed model can learn to distinguish, supporting the approach of using diverse activities to build model robustness.

3.2. Data Preprocessing and Windowing

To simplify the classification task, the original activity labels were restructured. The three non-seizure classes—standing, walking, and jumping—were combined into a single *non-seizure* class, while seizure-like activity was retained as the sole *seizure* class:

$$y_i = \begin{cases} 1 & \text{if seizure occurs} \\ 0 & \text{otherwise (all other activities combined)} \end{cases} \tag{2}$$

This consolidation reduces label complexity and forces the model to focus on discriminating seizure patterns, even in the presence of variability from different non-seizure activities. By merging diverse non-seizure activities, the dataset introduces realistic noise that the model must learn to ignore, improving robustness in real-world continuous monitoring.

After merging the labels, it was observed that non-seizure activity examples vastly outnumbered seizure examples. To reduce data redundancy and computational load, the dataset was initially downsampled by selecting every 5th row. Formally, the downsampled sequence $\tilde{X}$ is given by:

$$\tilde{X} = X[:: s], \quad s = 5 \tag{3}$$

where s is the sampling stride. This step preserves essential temporal patterns while decreasing memory requirements, which is crucial for mobile deployment with continuous sensor streaming.

Feature normalization was then applied using standard scaling:

$$X_{ij}^{\text{scaled}} = \frac{X_{ij} - \mu_j}{\sigma_j}, \quad i = 1, \ldots, N,\ j = 1, \ldots, D \tag{4}$$

where μ_j and σ_j are the mean and standard deviation of the j-th feature. Standardization ensures that all sensor channels contribute proportionally during training and prevents dominance of features with larger magnitudes, facilitating stable learning.

Given the significant class imbalance, a downsampling-balanced windowing strategy was employed. Consecutive readings of the same label were grouped into variable-length windows of size $l \in [l_{\min}, l_{\max}]$, and majority-class (non-seizure) windows were randomly discarded with probability:

$$P(\text{keep non-seizure block}) = p_{\text{keep}}, \quad p_{\text{keep}} = 0.3 \tag{5}$$

Shorter sequences were zero-padded to a fixed maximum length $l_{\max}$:

$$X^{\text{seq}} \in \mathbb{R}^{N_{\text{seq}} \times l_{\max} \times D}, \quad y^{\text{seq}} \in \{0, 1\}^{N_{\text{seq}}} \tag{6}$$

Downsampling was chosen over more sophisticated methods such as SMOTE to maintain temporal integrity of the sensor sequences and avoid introducing synthetic patterns that may not reflect realistic movements. This is particularly important for mobile deployment with continuous streaming, where reliability and real-time performance are critical.

Finally, the dataset was shuffled to prevent temporal correlations from biasing the learning process, ensuring the model generalizes well to unseen sequences. Overall, this preprocessing pipeline produces a compact, normalized, and balanced dataset suitable for training efficient deep learning models for real-time seizure detection on resource-constrained devices.

3.3. Deep Learning Architectures for Spatiotemporal Seizure Detection

The seizure detection system investigated in this study leverages two advanced deep learning architectures designed to capture the spatiotemporal characteristics of inertial sensor data collected from Android smartphones: a convolutional recurrent model and a convolutional attention-based model. Both architectures begin with multivariate motion input $X \in \mathbb{R}^{T \times C}$, where T represents the sequence length (windowed sequences of 5 seconds) and C the six sensor channels (three axes of accelerometer and three axes of gyroscope). From input to prediction, the data flow can be described as a sequence of transformations that extract local temporal patterns, model temporal dependencies, and integrate salient features across channels.

In the **convolutional recurrent model**, initial 1D convolutional layers act as localized feature extractors, applying a kernel $W_c \in \mathbb{R}^{k \times C \times F}$ to compute temporal filters:

$$H_t^{(1)} = \sigma\left(\sum_{i=0}^{k-1} W_c[i] \cdot X_{t+i} + b_c\right),$$

where F denotes the number of convolutional filters, k the kernel size, b_c the bias, and σ the activation function. Max-pooling layers follow, reducing sequence length and emphasizing dominant temporal patterns. The resulting feature map is passed through a gated recurrent unit (GRU) network, which integrates temporal information according to:

$$h_t = (1 - z_t) \odot h_{t-1} + z_t \odot \tilde{h}_t, \quad \tilde{h}_t = \tanh(W_h H_t^{(1)} + U_h(r_t \odot h_{t-1}) + b_h),$$

with update gate z_t and reset gate r_t, allowing the network to retain relevant past information efficiently. Fully connected layers summarize the extracted features, and a sigmoid output produces the seizure probability. This design captures both *short-term temporal dependencies* through convolution and *long-term dependencies* through GRU layers, while remaining parameter-efficient due to pooling and a relatively small hidden state dimension.

The **convolutional attention-based model** follows a similar front-end, with 1D convolutions and pooling to extract local temporal features. Instead of a recurrent layer, the architecture employs a multi-head self-attention mechanism to capture dependencies across all time steps simultaneously. For each attention head, the input $H^{(1)} \in \mathbb{R}^{T' \times F}$ (after convolution and pooling) is linearly projected to queries Q, keys K, and values V, and the attention output is computed as:

$$\text{head}_i = \text{softmax}\left(\frac{Q_i K_i^\top}{\sqrt{d_k}}\right) V_i,$$

where d_k is the key dimension. Multiple heads are concatenated and passed through a feedforward network with dropout and layer normalization, allowing the model to learn complementary spatiotemporal patterns across channels. A global average pooling layer condenses temporal information, and fully connected layers project the features to a final sigmoid output. By avoiding recurrent computations, this architecture achieves *efficient modeling of long-range temporal correlations* with fewer parameters, which is critical for smartphone deployment.

Both architectures integrate convolutional preprocessing with mechanisms that emphasize the most informative temporal and spatial features. The convolutional recurrent model excels at sequential dynamics modeling with compact GRU units, whereas the attention-based model provides global temporal context with parallelizable operations. In both cases, the parsimonious design—achieved via convolutional downsampling, small hidden dimensions, and attention heads—ensures a balance between expressivity and computational efficiency.

Hyperparameters, including convolutional filter counts, kernel sizes, GRU hidden dimensions, attention head numbers, and dropout rates, were optimized using Optuna[3]. This search ensured that both architectures were carefully tuned for the specific characteristics of smartphone inertial sensor data.

Given its successful deployment in the Android-based system and its ability to efficiently capture spatiotemporal dependencies, the **convolutional attention-based architecture** was selected for further evaluation. Its comparative performance against the convolutional recurrent model is presented in the subsequent subsection.

[3]Optuna is a state-of-the-art hyperparameter optimization framework that uses adaptive sampling strategies to efficiently explore the parameter space.

Table 1: Comparative performance metrics of the 1D CNN-GRU and 1D CNN-MHA models on the held-out test set.

Model	Accuracy	Precision	Recall (Sensitivity)	F1-Score	ROC-AUC
1D CNN–GRU	0.94	0.73	0.85	0.79	0.98
1D CNN–MHA	**0.96**	**0.81**	**0.93**	**0.87**	**0.99**

4. Results and Findings

4.1. Evaluation Protocol and Generalization Scope

The dataset was partitioned using a stratified window-level split rather than a subject-wise split. Consequently, the reported metrics quantify the separability of seizure-like motion patterns under realistic temporal conditions rather than subject-independent clinical generalization. This protocol was selected because the primary goal of the study is feasibility validation of an on-device architecture under deployment constraints, not estimation of population-level diagnostic performance. A subject-wise evaluation is a necessary next step for clinical validation and is therefore identified as future work. The present results should thus be interpreted as demonstrating architectural viability and robustness to motion variability rather than cross-patient predictive capability.

4.2. Comparative Model Performance and Selection

This section presents a detailed analysis of the training and evaluation results for the two parsimonious architectures developed in this study: the 1D Convolutional Neural Network with Gated Recurrent Units (1D CNN-GRU) and the 1D Convolutional Neural Network with Multi-Head Attention (1D CNN-MHA). The primary objective was to identify the model that best balances high detection performance with intrinsic compatibility for mobile deployment. The analysis is grounded in the classification metrics presented in Table 1 and the visual diagnostics compiled in Figure 2.

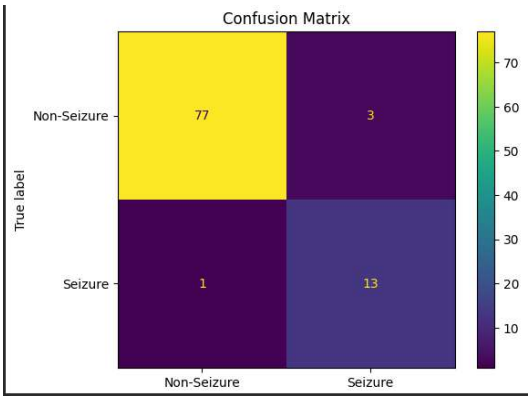
Training Dynamics and Convergence

The 1D CNN-GRU model exhibited a noisier training trajectory. After a rapid initial drop, its validation loss plateaued around epoch 8 and showed minor fluctuations until the early stopping trigger, indicating sensitivity to hyperparameters and a potential struggle to generalize stably from the inertial sensor data. In contrast, the 1D CNN-MHA model demonstrated smoother and more consistent convergence. Its validation loss descended steadily, while validation accuracy climbed more reliably, achieving a peak of 95.74% by epoch 9. This suggests that the multi-head attention mechanism provided a more stable gradient flow and better feature integration for this spatiotemporal classification task.

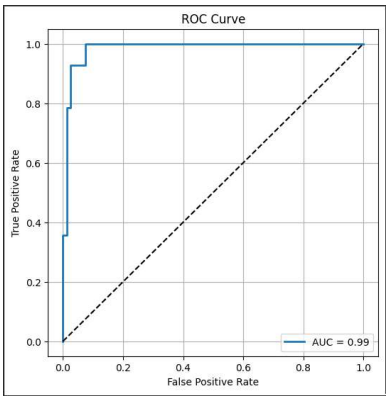
Final Classification Performance and Critical Trade-offs

The final evaluation on the independent test set, summarized in Table 1, shows that both models achieved high overall accuracy (94% and 96%, respectively). However, a deeper examination of the confusion matrices (Figure 2a,b) and per-class metrics reveals the decisive advantage of the 1D CNN-MHA architecture for a safety-critical application.

The 1D CNN-GRU model correctly identified 85% of seizure instances (recall/sensitivity of 0.85) but at a lower precision (0.73), meaning it had a higher rate of false positive alarms. The 1D CNN-MHA model significantly improved upon this critical trade-off, achieving a superior sensitivity of 0.93 (correctly detecting 93% of seizures) while also raising precision to 0.81, thereby reducing false alarms. This is reflected in its higher F1-score (0.87 vs. 0.79) and near-perfect AUC (0.99 vs. 0.98). In the context of epileptic seizure detection, maximizing recall is paramount to prevent missed events that could lead to injury or SUDEP. The 1D CNN-MHA's 8-percentage-point improvement in sensitivity, without sacrificing overall accuracy, represents a clinically significant enhancement.



(a) Confusion Matrix: CNN–MHA



(b) ROC Curve: CNN–MHA

Figure 2: Performance analysis of the 1D CNN-MHA model. (a) Confusion matrix showing the model's classification performance on the test set. (b) Receiver Operating Characteristic (ROC) curve demonstrating the model's discrimination capability with an AUC of 0.99.

Table 2: Benchmarking against models claiming "lightweight" or efficient design.

Study / Model	**Model Size**	**Key Performance**	**Deployment Evidence**
Zhang et al. (2022) [9]	21.32 KB	94.44% Sens., 0.98 AUC	None; tested only on competition dataset.
Rukhsar and Tiwari (2023) [10]	Not Reported	96.31% Acc., 96.32% F1	None; requires 18-channel EEG cap.
Li et al. (2025) [11]	Not Reported	"Superior results" (CHB-MIT)	None; claims "real-time" without metrics.
1D CNN–MHA	**23.8 KB (TFLite)**	**93% Sens., 96% Acc., 0.99 AUC**	**Deployed on Flutter app; real-time inference on mid-range Android.**

Model Selection for Deployment

Based on the exhaustive analysis above, the 1D CNN-MHA model was selected for further benchmarking and deployment. This decision was grounded on two pillars: 1. **Superior Detection Performance:** It provided a better balance of high sensitivity and precision, as evidenced by superior recall, F1-score, and AUC. 2. **Intrinsic Deployment Feasibility:** Crucially, unlike the GRU-based model, the 1D CNN-MHA architecture does not rely on TensorFlow operations (e.g., `FlexTensorListReserve`) that require the non-standard TensorFlow Lite Flex delegate. This allows for straightforward conversion to a standard TFLite model and seamless integration into mobile applications, a prerequisite for use in low-resource settings.

4.3. Benchmarking Against State-of-the-Art and "Lightweight" Models

Having selected the 1D CNN-MHA model based on its superior performance and intrinsic mobile compatibility, it is now benchmarked against related work to validate its parsimony. The comparison is structured across two dimensions: (1) models that claim efficiency ("lightweight") but lack deployment evidence, and (2) models that represent the performance ceiling but are computationally prohibitive for mobile deployment. This dual-axis analysis, presented in Tables 2 and 3, highlights the persistent deployment gap in the literature and positions the model as a viable bridge between accuracy and real-world utility.

The benchmarking reveals a clear, two-fold gap in the state of the art. The models in Table 2, while making efficiency claims, provide no proof of deployment—they represent *theoretical parsimony*.

Table 3: Benchmarking against high-performance, computationally heavy models.

Study / Model	Key Performance	Compute Requirements	Deployment Feasibility
Zhong et al. (2024) [7]	96.15% Acc., 96.11% Sens.	NVIDIA RTX 3050 GPU; S-Transform	Server/cloud only.
Hussein et al. (2022) [6]	90% Acc., 87-92% Sens.	HPC system; multi-channel ViT	Cloud API; authors note "high computational cost".
Yuan et al. (2024) [19]	92.2% Acc., 93% Sens.	High-end GPU; Hybrid DenseNet-ViT	Research/server only; "lightweight" is future work.
1D CNN–MHA	**96% Acc., 93% Sens., 0.99 AUC**	**Mid-range smartphone CPU**	**On-device, real-time inference; field-tested.**

The models in Table 3 achieve high accuracy but are architecturally and computationally bound to server-grade hardware—they represent the *performance ceiling without deployability*.

The 1D CNN-MHA model passes the parsimony test on both fronts. First, it achieves competitive sensitivity and accuracy (93%, 96%, 0.99 AUC) that is on par with or superior to the heavy models, demonstrating that high performance does not necessitate complex transformers. Second, unlike the "lightweight" claimants, it provides concrete deployment evidence: a compact 23.8 KB TFLite model running in real-time on a consumer smartphone via a Flutter application. This end-to-end validation fulfills the promise of parsimonious design.

Beyond this specific application, the architecture's modularity is a key strength. The convolutional front-end for local feature extraction coupled with a flexible temporal modeling component (here, multi-head attention) creates a template that can be adapted to various sensor modalities and dataset sizes for any task requiring spatio-temporal pattern recognition.

4.4. Efficiency–Performance Trade-off

The two evaluated architectures provide empirical anchor points for the accuracy–efficiency trade-off central to parsimonious learning. The CNN–GRU model offers competitive accuracy but requires operations incompatible with standard mobile inference runtimes, while the CNN–MHA model achieves similar detection performance while remaining directly deployable as a compact TensorFlow Lite model (23.8 KB) with real-time latency on mid-range smartphones. This contrast illustrates a practical "knee" in the performance–deployability curve: increasing architectural complexity does not meaningfully improve detection performance but breaks deployment feasibility. The results therefore support the claim that, in constrained environments, architectural compatibility and computational simplicity can dominate marginal accuracy gains, embodying the principle that less can be more.

5. Conclusion and Discussion

5.1. Conclusion

This work demonstrates that parsimonious design is not merely an optimization strategy but a fundamental prerequisite for deploying life-saving neurological tools in low-resource settings. Two lightweight architectures were systematically developed and evaluated—1D CNN-GRU and 1D CNN-MHA—for epileptic seizure detection using smartphone inertial sensors. The 1D CNN-MHA model was selected based on its superior performance (96% accuracy, 93% sensitivity, 0.99 AUC) and, crucially, its intrinsic compatibility with mobile deployment, avoiding TensorFlow Flex delegate dependencies that rendered the GRU-based model impractical.

The dual-axis benchmarking against the state of the art revealed a persistent deployment gap. While "lightweight" models in the literature make efficiency claims without deployment proof [9–11], and high-accuracy models are architecturally bound to server-grade hardware [6, 7, 19], the 23.8 KB TFLite model bridges this gap. It delivers competitive accuracy while running in real-time on mid-range Android smartphones via a Flutter application, validated through field testing.

5.2. Ethical Considerations

The development and deployment of mobile health technologies for vulnerable populations necessitate careful ethical scrutiny. Two primary considerations are addressed here, with additional details provided in the Appendix.

Data Privacy and Security: All data collection followed informed consent protocols approved by relevant institutional review boards. Sensor data was anonymized at source, with no personally identifiable information stored alongside motion recordings. The deployed application processes all data locally on the device; no physiological data is transmitted to external servers, eliminating risks associated with data breaches or unauthorized cloud access.

Algorithmic Transparency and Accountability: While deep learning models are inherently complex, the interpretability of the parsimonious architecture is emphasized through visual diagnostics of training and confusion matrices. Clinical deployment would require clear communication to users and caregivers about the system's limitations—namely, that it is a detection aid, not a diagnostic tool, and cannot prevent all seizures or SUDEP.

5.3. Limitations

This study represents a pilot feasibility investigation and has several important limitations. The dataset consists of a small number of participants, including simulated seizure events from healthy volunteers, and therefore does not capture the full clinical diversity of seizure semiology. Additionally, evaluation used a stratified window-level split rather than a subject-wise protocol, meaning the reported performance reflects pattern discrimination capability rather than population generalization. The system should therefore not be interpreted as a clinically validated diagnostic tool. Instead, it establishes the technical viability of a deployable architecture intended to enable larger subject-independent clinical studies.

5.4. Future Work

While this study establishes the viability of parsimonious models for mobile seizure detection, several avenues warrant further investigation to advance clinical impact and scientific understanding.

Clinical Validation and Longitudinal Studies: The next critical step is a large-scale, longitudinal clinical trial across diverse epilepsy populations and environments. This will validate the model's sensitivity and specificity in real-world conditions, establish clinical utility metrics, and refine detection algorithms based on a broader range of seizure semiologies.

Personalization and Continual Learning: Developing efficient on-device personalization techniques would allow the model to adapt to an individual's unique movement patterns and seizure characteristics over time, improving accuracy while maintaining privacy. Research into tiny continual learning algorithms suitable for resource-constrained hardware is needed.

Socio-Technical Deployment Frameworks: Technical validation must be accompanied by research into sustainable deployment models. This includes understanding barriers to adoption by patients and healthcare workers, developing appropriate training materials for low-literacy settings, and creating viable maintenance and support ecosystems within existing local health infrastructures.

Acknowledgements

We thank the CPAL reviewers and area chairs for their constructive feedback. We are grateful to the study participants for their time and contributions. We also thank colleagues at African Leadership University and Carnegie Mellon University Africa for helpful discussions and logistical support.

References

[1] World Health Organization. Epilepsy. *WHO Fact Sheet*, 2024.

[2] Orrin Devinsky, Dale C Hesdorffer, David J Thurman, Samden Lhatoo, and George Richerson. Sudden unexpected death in epilepsy: epidemiology, mechanisms, and prevention. *The Lancet Neurology*, 15(10):1075–1088, 2016.

[3] Ali A Asadi-Pooya et al. How to successfully establish an epilepsy care center in resource-limited nations: a systematic review. *Seizure*, 112:1–8, 2023.

[4] Bry Sylla, Ouedraogo Ismaila, and Gayo Diallo. 25 years of digital health toward universal health coverage in low- and middle-income countries: Rapid systematic review. *Journal of Medical Internet Research*, 27:e59042, 2025.

[5] Khansa Rasheed, Adnan Qayyum, Junaid Qadir, et al. Machine learning for predicting epileptic seizures using eeg signals: a review. *IEEE Reviews in Biomedical Engineering*, 14:139–155, 2021.

[6] Ramy Hussein, Soojin Lee, and Rabab Ward. Multi-channel vision transformer for epileptic seizure prediction. *Biomedicines*, 10(7):1551, 2022. doi: 10.3390/biomedicines10071551.

[7] Xiangwen Zhong, Guoyang Liu, Xingchen Dong, Chuanyu Li, Haotian Li, Haozhou Cui, and Weidong Zhou. Automatic seizure detection based on stockwell transform and transformer. *Sensors*, 24(1):77, 2024. doi: 10.3390/s24010077.

[8] Xiaoqiong Xiong, Xuemei Xiong, Keda Zeng, and Chao Lian. A survey of motion data processing and classification techniques based on wearable sensors. *IGMIN Research*, 1(1):1–15, 2025.

[9] Yang Zhang, Yvon Savaria, Shiqi Zhao, Gonçalo Mordido, Mohamad Sawan, and François Leduc-Primeau. Tiny CNN for seizure prediction in wearable biomedical devices. In *2022 44th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, pages 1306–1309. IEEE, 2022. doi: 10.1109/EMBC48229.2022.9872006.

[10] Salim Rukhsar and Anil Kumar Tiwari. Lightweight convolution transformer for cross-patient seizure detection in multi-channel EEG signals. *Computer Methods and Programs in Biomedicine*, 242:107856, 2023. ISSN 0169-2607. doi: 10.1016/j.cmpb.2023.107856.

[11] Chuanyu Li, Haotian Li, Xingchen Dong, Xiangwen Zhong, Haozhou Cui, Dezan Ji, Landi He, Guoyang Liu, and Weidong Zhou. CNN-Informer: A hybrid deep learning model for seizure detection on long-term EEG. *Neural Networks*, 181:106855, 2025. ISSN 0893-6080. doi: 10.1016/j.neunet.2024.106855.

[12] World Health Organization. Intersectoral global action plan on epilepsy and other neurological disorders 2022–2031. *WHO Technical Report*, 2023.

[13] Maxime Sasseville, Eugène Attisso, Marie-Pierre Gagnon, et al. Performance, impact and experiences of using wearable devices for seizure detection in community-based settings: a mixed methods systematic review. *mHealth*, 10:27, 2024.

[14] Elisa Bruno, Pedro F Viana, Michael R Sperling, and Mark P Richardson. Seizure detection at home: Do devices on the market match the needs of people living with epilepsy and their caregivers? *Epilepsia*, 61(S1):11–24, 2020.

[15] Sabri Hassan et al. Iot-based monitoring system for epileptic patients. *Heliyon*, 8(7):e09984, 2022.

[16] Natarajan Sriraam et al. Application of a low-cost mhealth solution for the remote automated detection of epileptic seizures. *JMIR Neurotechnology*, 2(1):e50660, 2023.

[17] Elliot Mbunge, Caron Lee Jack, Maureen Nokuthula Sibiya, and John Batani. Review of implementation barriers and strategic approaches for improving mhealth systems utilization in africa: Lessons learnt from south africa and kenya. *Telematics and Informatics Reports*, 19:100228, 2025.

[18] Jack Williams et al. Smartphone eeg and remote online interpretation for epilepsy in a low-resource setting: Experience from guinea. *Frontiers in Neurology*, 10:1–12, 2019.

[19] Shasha Yuan, Kuiting Yan, Shihan Wang, Jin-Xing Liu, and Juan Wang. EEG-based seizure prediction using hybrid DenseNet-ViT network with attention fusion. *Brain Sciences*, 14(8): 839, 2024. doi: 10.3390/brainsci14080839.

A. Appendix

A.1. Extended Ethical Considerations

The following considerations supplement the ethical discussion in the main paper:

Informed Consent and Autonomy: Participants, including epilepsy patients, provided explicit consent after a thorough explanation of the study's purpose, data usage, and potential risks. They retained the right to withdraw at any time without affecting their clinical care. For patients with limited literacy, consent procedures were conducted verbally with an impartial witness present.

Equity and Access: The parsimonious design philosophy is inherently equity-focused. By targeting hardware already prevalent in low-income regions, we intentionally avoid solutions that would exacerbate global health disparities. Future deployment must consider sustainable maintenance, local capacity building, and affordability to ensure genuine access rather than technological imposition.

Safety and Risk Mitigation: False negatives (missed seizures) pose the gravest risk. Our model's high sensitivity (93%) is prioritized to minimize this danger. However, clear user guidelines must stress that the system complements, rather than replaces, clinical supervision and standard care practices.

A.2. Complete Training Curves and Model Analysis

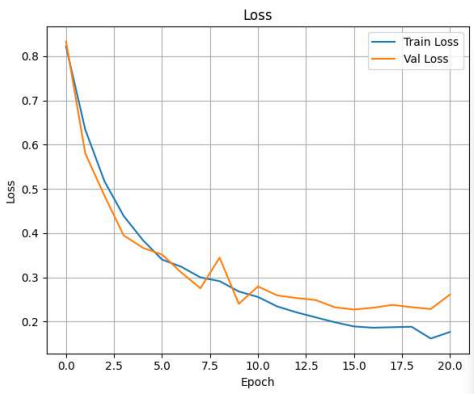


Figure 3: Loss Curve: CNN–GRU

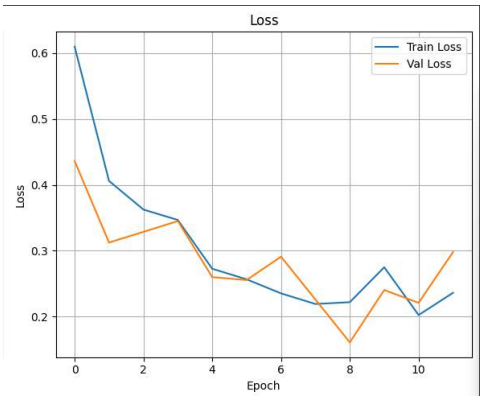


Figure 4: Loss Curve: CNN–MHA

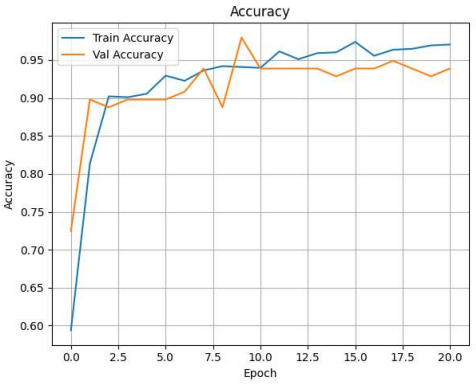


Figure 5: Accuracy Curve: CNN–GRU

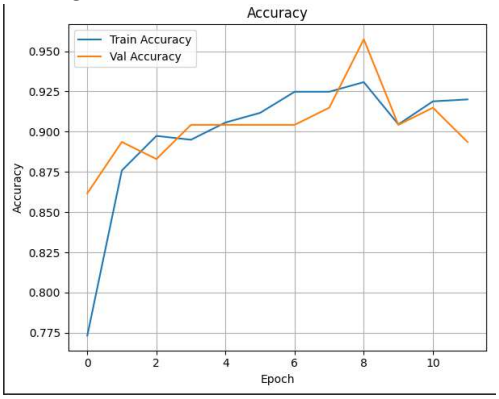


Figure 6: Accuracy Curve: CNN–MHA

Figure 7: Complete training and validation curves for both models. Top row shows loss curves, bottom row shows accuracy curves. These detailed training dynamics provide insight into model convergence behavior and generalization capability.

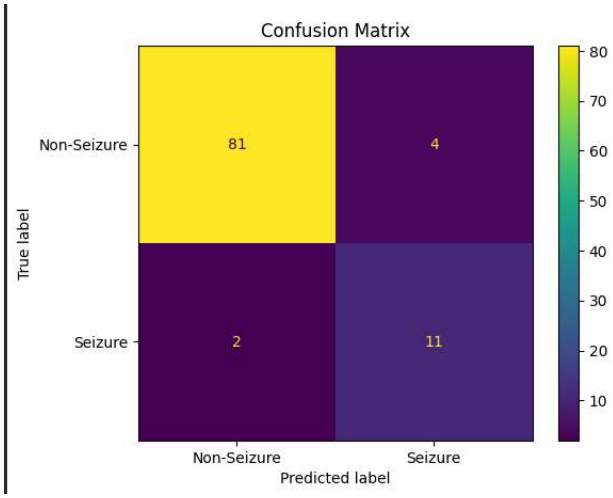


Figure 8: Confusion matrix for the 1D CNN-GRU model on the test set.

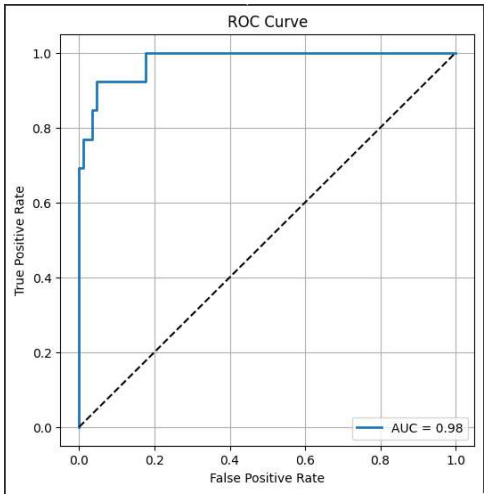


Figure 9: Receiver Operating Characteristic (ROC) curve demonstrating the CNN–GRU model's discrimination capability with an AUC of 0.98.

A.3. Additional Figures

Data collection code snippet and sample CSV file

```
class _DataCollectorScreenState extends State<DataCollectorScreen> {
  List<List<dynamic>> _data = [];
  StreamSubscription? _accelSub;
  StreamSubscription? _gyroSub;
  Timer? _recordingTimer;

  double ax = 0, ay = 0, az = 0, gx = 0, gy = 0, gz = 0;
  int activityLabel = 0; // 0=Standing, 1=Walking, 2=Seizure, 3=Jumping
  bool isRecording = false;

  @override
  void initState() {
    super.initState();
    _data.add(['timestamp', 'accX', 'accY', 'accZ', 'gyroX', 'gyroY', 'gyroZ', 'activity']);
  }

  void _startRecording() {
    setState(() => isRecording = true);

    _accelSub = accelerometerEventStream().listen((event) {
      ax = event.x;
      ay = event.y;
      az = event.z;
    });

    _gyroSub = gyroscopeEventStream().listen((event) {
      gx = event.x;
      gy = event.y;
      gz = event.z;
    });

    _recordingTimer = Timer.periodic(const Duration(milliseconds: 100), (timer) {
```

Figure 10: Data collection code implementation

	A	B	C	D	E	F	G	H
1	timestamp	accX	accY	accZ	gyroX	gyroY	gyroZ	activity
2	2025-04-08 12:20:26	-1.635314584	3.724105835	9.623344421	0.2563535571	0.6198916435	0.1171602234	2
3	2025-04-08 12:20:27	-1.387215376	3.745204926	9.327510834	-0.1670886278	-0.1993472427	0.1351797432	2
4	2025-04-08 12:20:27	-1.004143119	3.499873877	9.012523651	-0.05927124619	-0.0933617875	0.0648836121	2
5	2025-04-08 12:20:27	-1.28501296	3.176881313	8.782081604	-0.05927124619	-0.0933617875	0.0648836121	2
6	2025-04-08 12:20:27	-1.109712481	3.21249485	9.189544678	-0.06648238748	-0.02468108758	-0.04574827477	2
7	2025-04-08 12:20:27	-0.8784475923	3.385850191	9.272743225	-0.1020051688	-0.06729842722	-0.01005895808	2

Figure 11: Sample collected CSV data

A.4. Implementation Resources

To promote reproducibility and further development, we provide the following resources:

- **Model Development Notebook:** The complete Jupyter notebook containing data preprocessing, model architectures, training code, and evaluation scripts is available at: `https://github.com/Isaiah-Essien/parsimony_seizure_experiments`
- **Mobile Application Repository:** The Flutter application source code, including the integrated TFLite model and real-time inference pipeline, is available at: `https://github.com/Isaiah-Essien/uboho_mobile`
- **Model Files:** The trained 1D CNN-MHA model in TensorFlow Lite (.tflite) format is included in the '/assets/model' directory of the application repository.

These resources are provided under open-source licenses to facilitate research collaboration and clinical translation efforts.

A.5. Statement on the Use of Large Language Models

This work was conducted without the use of artificial intelligence generation tools. All model architectures and design decisions were developed by the authors based on a self-thought out process and a review of relevant literature, as documented herein. The complete analytical process, including all visualizations referenced in this paper, is provided in the accompanying computational notebook included in the appendix.