REST: Efficient and Accelerated EEG Seizure Analysis through Residual State Updates

Arshia Afzal [1 2] **Grigorios Chrysos** [3] **Volkan Cevher** [* 2] **Mahsa Shoaran** [* 1]

Abstract

EEG-based seizure detection models face challenges in terms of inference speed and memory efficiency, limiting their real-time implementation in clinical devices. This paper introduces a novel graph-based residual state update mechanism (REST) for real-time EEG signal analysis in applications such as epileptic seizure detection. By leveraging a combination of graph neural networks and recurrent structures, REST efficiently captures both non-Euclidean geometry and temporal dependencies within EEG data. Our model demonstrates high accuracy in both seizure detection and classification tasks. Notably, REST achieves a remarkable 9-fold acceleration in inference speed compared to state-of-the-art models, while simultaneously demanding substantially less memory than the smallest model employed for this task. These attributes position REST as a promising candidate for real-time implementation in clinical devices, such as Responsive Neurostimulation or seizure alert systems.

1. Introduction

Brain disorders, including epilepsy, present substantial challenges globally, prompting the need for innovative approaches in diagnosis and treatment. Recurrent seizures, recognized as one of the most prevalent neurological emergencies globally (Strein et al., 2019), impact approximately 50 million people worldwide (Beghi et al., 2019).

Detecting changes in the rhythms of brain activity through the monitoring of electroencephalography (EEG) signal allows us to pinpoint the onset zone and time of seizures (Gotman, 1990; Siddiqui et al., 2020), making EEG an invaluable and extensively utilized tool for seizure detection and localization. Traditionally, neurological experts perform these tasks, involving the time-consuming process of manually labeling periods spanning from hours to days for each individual patient (Harrer et al., 2019; Ahmedt-Aristizabal et al., 2020). Several studies have explored the application of Machine Learning (ML) in seizure analysis, aiming to simplify the handling of large seizure datasets for experts (Tang et al., 2021; Ahmedt-Aristizabal et al., 2020; Covert et al., 2019; Siddiqui et al., 2020). These studies predominantly focus on deep models, known for their accuracy and suitability for clinical applications.

Taking inspiration from computer vision (Voulodimos et al., 2018), many studies have applied different variations of Convolutional Neural Networks (CNN) for seizure detection, as demonstrated in Saab et al. (2020). Various versions of Graph Neural Networks (GNN) effectively capture non-Euclidean geometry in datasets like EEG signals, contributing to enhanced seizure detection and classification (Li et al., 2022; Tang et al., 2021; Ho & Armanfard, 2023). Additionally, to enhance the performance of deep neural networks and accounting for time-series nature of brain rhythms, different variations of Recurrent Neural Networks (RNN) have been utilized in seizure analyses (Ahmedt-Aristizabal et al., 2020).

While these models excel in achieving high accuracy in seizure detection and classification tasks, they often struggle with issues such as complexity, inefficient memory usage, and slow inference speeds. One of the main reasons behind this inefficiency lies in structures such as the gating mechanism found in RNN models (e.g., Long Short-Term Memory (LSTM) (Hochreiter & Schmidhuber, 1997) or the presence of deep convolutional layers in CNNs and GNNs.

Both inference time and memory storage considerations become critically important in the context of modern seizure treatment devices like Responsive Neurostimulation (RNS) and Deep Brain Stimulation (DBS) (Fisher & Velasco, 2014a; Sun & Morrell, 2014). These devices, which have shown promise in suppressing seizure attacks, require a small yet accurate ML model to trigger stimulation commands for symptom suppression (Shoaran et al., 2016; Shin et al., 2022). Furthermore, the model must exhibit low

[1]INL, EPFL, Switzerland [2]LIONS, EPFL, Switzerland [3]Department of Electrical and Computer Engineering, University of Wisconsin-Madison, USA. Correspondence to: Arshia Afzal <arshia.afzal@epfl.ch>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024. Copyright 2024 by the author(s).

inference time in activating the stimulator to ensure its effectiveness (Fisher & Velasco, 2014b; Zhu et al., 2021). Unfortunately the aforementioned methods do not have such a low inference.

In this study, we introduce REST, a graph-based residual update mechanism designed to efficiently detect both spatial and temporal information from EEG. REST captures spatio-temporal dependencies in EEG signals without relying on computationally expensive gating mechanisms commonly found in existing models (Hochreiter & Schmidhuber, 1997; Cho et al., 2014; Asif et al., 2020; Tang et al., 2021). The ability to dynamically capture spatial information over time and update the state accordingly contributes to the high accuracy of REST in localizing and detecting seizures. Notably, REST attains comparable accuracy to state-of-the-art models, while achieving significantly faster processing during inference and substantially reducing computational and memory overhead [1]. Our contributions are as follows:

- We present a novel graph-based residual update mechanism designed to capture spatio-temporal dependencies in EEG signals.
- We enhance the model's performance while maintaining its small size and rapid detection and classification speed using binary random masking the state and multiple state updates.
- Our model delivers predictions with an impressive inference latency of 1.29ms. This unmatched inference speed is achieved with a light memory footprint of 37KB.
- Our model is 14× smaller than the smallest competitive models for seizure detection. Remarkably, our architecture can match the performance of the state-of-the-art deep neural networks with less than 10K parameters.

2. Related Work

Many studies have attempted to develop ML and deep learning models for seizure detection (Siddiqui et al., 2020; O'Shea et al., 2020; Saab et al., 2020) and classification of seizure types (Ahmedt-Aristizabal et al., 2020; Iešmantas & Alzbutas, 2020; Tang et al., 2021). Here, we examine existing seizure detection and classification models, assessing their strengths and limitations across three key aspects. Firstly, we explore how these studies capture the spatio-temporal features present in EEG. Secondly, we delve into the inference speed and the impact of varying clip lengths on seizure analysis. Lastly, we study the memory requirements and model size of current models.

[1]Visit our web site at https://arshiaafzal.github.io/REST/

Spatio-Temporal Nature of EEG Signals: As introduced earlier, the nature of EEG signals involves both spatial and temporal components, which are pivotal for accurate analysis in epilepsy studies. Notably, some studies, like Asif et al. (2020), extract spectral features to represent temporal dependencies, incorporating them into a CNN architecture. In contrast, Saab et al. (2020) employ a CNN model that treats EEG signals as multi-channel images, a methodology that does not align with the time-series structure of EEG. Recent advancements involve the utilization of various RNN variations or transformers (Vaswani et al., 2017) to effectively capture temporal patterns in alignment with the intricate dynamics of EEG signals.

RNNs capture temporal dependencies within time-series data by mapping the input $x(t)$ into a latent space $h(t)$ and employ recurrence within that space through linear or non-linear transformations. Despite their effectiveness in capturing time-series dependencies, RNNs suffer from a significant challenge known as gradient vanishing. This issue occurs during backpropagation, causing gradients to diminish and hindering the effective learning of long-range dependencies in sequential data. To address the vanishing gradient problem (Pascanu et al., 2013), RNN variants like LSTM (Hochreiter & Schmidhuber, 1997) or Gated Reccurent Unit (GRU) (Cho et al., 2014) leverage gating mechanisms, introducing different gates that contribute to creating the next state $h(t)$ from the current input $x(t)$ and the previous state $h(t-1)$. Thodoroff et al. (2016) used an LSTM based model for seizure detection.

On the other hand, attention-based models or transformers (Vaswani et al., 2017) are more complex than RNNs. Rather than constructing an explicit state, they directly use previous inputs to predict the future. However, this approach is more memory-intensive and time-demanding due to the necessity of retaining all prior inputs up to a specified time point and storing weights for each input to construct the attention matrix. Yan et al. (2022b) employed a transformer-based model for the seizure detection task.

In the context of EEG analysis where spatial details are critical at each time point, a common strategy is to utilize a CNN or graph convolution network independently across all time points, mapping them into a new feature space. This approach is then complemented by RNN to capture temporal dependencies. Ahmedt-Aristizabal et al. (2020) further employ a CNN-LSTM model, effectively addressing both spatial and temporal dependencies in EEG data.

Nevertheless, these approaches assume Euclidean geometry for EEG signals, overlooking the natural geometry of electrode placement (Figure 1 a) and brain network connectivity (Tang et al., 2021). Recent studies exploit GNNs and graph-based modeling to capture the non-Euclidean geometry of EEG signals (Tang et al., 2021; Ho & Armanfard, 2023;

Table 1. Comparison of seizure detection and classification methods. **A)** Capturing the non-Euclidean geometry of EEG signals. **B)** Capturing the temporal behavior of EEG signals. **C)** Evaluated for both short and long-term seizure detection. **D)** Runtime efficient **E)** Memory efficient.

Method	A	B	C	D	E
SeizureNet (Asif et al., 2020)	✗	✔	✗	✗	✗
Transformer (Yan et al., 2022a)	✗	✔	✔	✗	✗
EEG-CGS (Ho & Armanfard, 2023)	✔	✔	✗	✗	✗
GGN (Li et al., 2022)	✔	✔	✔	✗	✗
LSTM (Hochreiter & Schmidhuber, 1997)	✗	✔	✗	✗	✗
CNN-LSTM [1] (Ahmedt-Aristizabal et al., 2020)	✗	✔	✗	✗	✗
CNN-LSTM [2] (Thodoroff et al., 2016)	✗	✔	✗	✗	✗
DCRNN (Tang et al., 2021)	✔	✔	✔	✗	✗
REST (Ours)	✔	✔	✔	✔	✔

Covert et al., 2019; Li et al., 2022). For instance, Tang et al. (2021) implement a self-supervised diffusion graph convolution model for both detection and classification tasks. Similarly, Ho & Armanfard (2023) employ a self-supervised graph network for channel anomaly detection. These studies (Ho & Armanfard, 2023; Tang et al., 2021) align more closely with the dynamic changes in EEG rhythms by replacing the weights of the RNN network with graph convolution filters. This approach represents the evolution of spectral features within each time point of the time-series data, offering a more integrated approach compared to the sequential mapping from CNN to LSTM (Ahmedt-Aristizabal et al., 2020).

Significance of Inference Time: Timely detection of seizure events is essential for the efficacy of closed-loop epileptic treatments such as RNS and DBS (Shoaran et al., 2016). To the best of our knowledge, most previous studies either overlook the importance of inference runtime or, as observed in Asif et al. (2020), consider a 90ms delay for giving predictions. This delay is still significant, especially for edge devices like RNS and DBS. Furthermore, current studies often evaluate models using a limited range of long window sizes, typically exceeding 10 seconds or even 1 minute (Tang et al., 2021; Saab et al., 2020). However, shorter window sizes are preferable for real-time seizure detection and responsive intervention (Christou et al., 2022; Zhu et al., 2020). The chosen window size influences a model's ability to localize seizures and its overall detection performance. For instance, a model designed for extended window sizes may lose accuracy in short-term seizure detection scenarios, an aspect that has not been extensively explored in the literature.

Memory Requirement in Seizure Detection Models: While numerous studies have focused on enhancing the accuracy of seizure detection and classification tasks, the crucial aspect of memory demand remains largely overlooked. For instance, Tang et al. (2021) utilize 240K parameters with complex gating units, Ho & Armanfard (2023) employ 58K for channel anomaly detection, and Asif et al. (2020) address seizure classification task with a substantial number of 45.94 Million parameters. These examples underscore the need for an efficient model tailored for seizure detection and classification problems, especially one suitable for resource-constrained stimulation devices deployed at the edge, which do not have access to extensive memory storage for model weights and states (Zhu et al., 2020).

In Table 1, we present a summary of current models, highlighting their respective strengths and weaknesses.

3. Method

Below, we first formulate the tasks of seizure detection and classification, outlining the graph representation of EEG signals. Next, we describe the design of REST's structure using various updating strategies.

3.1. Seizure Detection and Classification Problem Setting

Following the preprocessing of raw EEG signals and constructing the EEG graph, we obtain an EEG clip X and a label y for both detection and classification tasks. Here, $X \in \mathbb{R}^{T \times M \times N}$ with N electrodes, T time points and M features per node while y denotes the label. For detection, the label is binary, whereas for classification, the label falls within the range of {0,1,2,3,4} where each class represents a unique seizure type [2]. The goal for both tasks is to predict the label y based on a given EEG clip X.

3.2. EEG Distance Graph Construction

For each EEG clip, we denote a graph as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathcal{A}\}$ where $\mathcal{V} = \{v_1, ..., v_N\}$ represents the nodes corresponding to EEG electrodes, $\mathcal{E}$ represents the edges, and $\mathcal{A} \in \mathbb{R}^{N \times N}$ denotes the adjacency matrix of the graph where N is the number of nodes which in case of EEG data it is the EEG electrodes. We build a distance-based EEG graph (Figure 1a) that precisely represents the electrode placement geometry in the standard 10/20 system (Jasper, 1958). Unlike correlation graphs, our graph remains static over time, reducing computations during inference, as the graph struc-

[2]The five seizure types include: focal, generalized non-specific, complex partial, absence, and tonic-clonic.

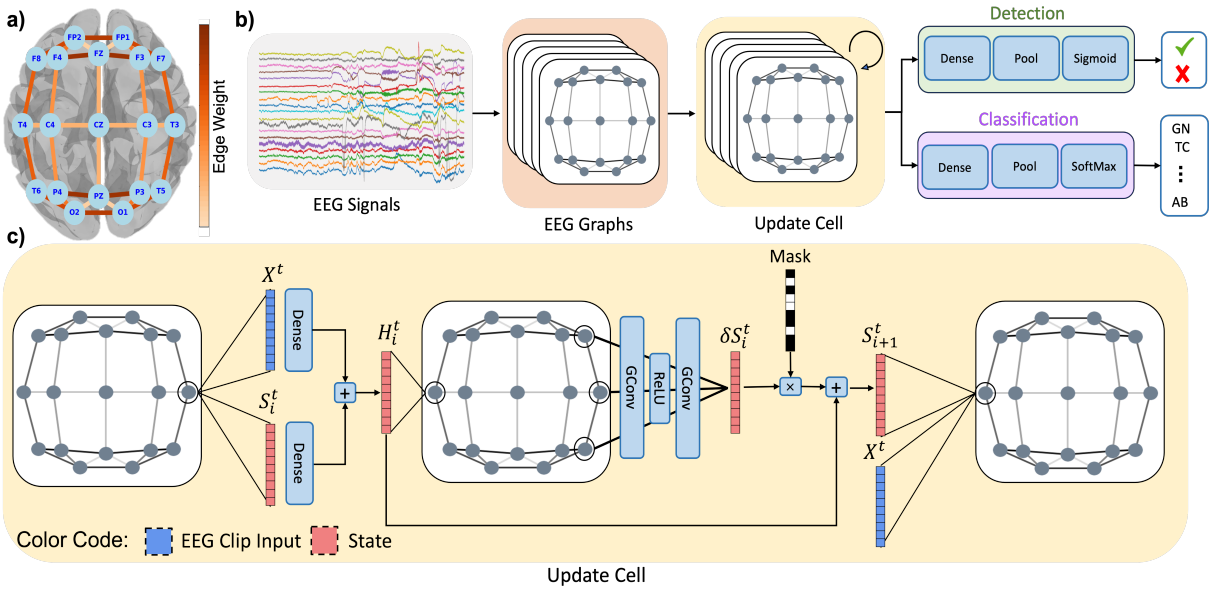

Figure 1. **(a)** EEG electrodes placement based on the 10/20 standard and its constructed distanced based EEG graph. Self edges are not shown for better visualization. **(b)** The REST framework, where raw EEG signals undergo preprocessing and are structured as a graph before feeding as input to the model. Following multiple (or single) updates, the model provides the detection or classification result. **(c)** Single update mechanism of the proposed model. Dense represents the fully connected layer and GConv is the graph convolution. See our web page for more visual results at https://arshiaafzal.github.io/REST/.

ture does not need to be constructed for each input (Ho & Armanfard, 2023). Details regarding the choice of k and visualization of distance graphs based on threshold values can be found in Appendix H.

For a distance graph, the adjacency matrix is constructed using the distance between electrode locations, as in previous studies (Tang et al., 2021; Li et al., 2022; Ho & Armanfard, 2023). As the EEG electrode placements are fixed, the adjacency matrix remains unchanged over time. Thus, for each element $a_{ij} \in \mathcal{A}$:

$$a_{ij} = \begin{cases} \exp(-\frac{||v_i - v_j||^2}{\sigma^2}) & \text{if } ||v_i - v_j|| \leq k, \\ 0 & \text{if Otherwise,} \end{cases} \tag{1}$$

where σ is the standard deviation of the distances and k is the Gaussian kernel's threshold (Shuman et al., 2013).

3.3. Residual State Update

Similar to RNNs, REST initially maps the input into a latent space, evolving the state over time to reach the final output. In contrast to RNNs, REST updates the state using a novel approach that avoids the complexity of gating mechanisms like LSTM or GRU, efficiently addressing the vanishing gradient problem with fewer parameters (details in Appendix B). For mapping to the state space, REST employs a linear mapping represented as:

$$H^t = WX^t + US^{t-1}. \tag{2}$$

Here, $X^t \in \mathbb{R}^{M \times N}$ represents the input, in our case, the preprocessed EEG clip at time point $t \in [1, ..., T]$, and $S^{t-1} \in \mathbb{R}^{Q \times N}$ is the previous state of the model at time point $t-1$. $W \in \mathbb{R}^{Q \times M}$ and $U \in \mathbb{R}^{Q \times Q}$ are the weights of the affine mapping, with Q being the state size, while $H^t \in \mathbb{R}^{Q \times N}$ represents the state of REST prior to the update. Inspired by He et al. (2016), REST uses a residual mechanism to update its latent state:

$$S^t = H^t + \delta S^t. \tag{3}$$

Here, S^t is the next state of the model and δS^t is the incremental update for the model's state. The critical aspect lies in extracting δS^t to align with the spatial changes in EEG dynamics at each time point. For this purpose, we utilize the graph convolution method introduced by Morris et al. (2019). We opt for this graph convolution because of its simple structure, which is suited for our application. The graph convolution is defined as follows:

$$O^t_{[:,i]} = \sigma\Big(\Theta_1 H^t_{[:,i]} + \Theta_2 \sum_{j \neq i} a_{ij} H^t_{[:,j]}\Big), \tag{4}$$

where $O^t \in \mathbb{R}^{Q \times N}$ is the output of the convolutional filter with Q features per node. $\Theta_1, \Theta_2 \in \mathbb{R}^{Q \times Q}$ parameterize the first and second convolutional filters, a_{ij} represents the edge (in this case, the adjacency matrix element) between node $i, j \in 1, \ldots, N$ and σ is the activation function. We denote the graph convolution in Equation (4) as $\mathcal{G}_\Theta(H^t)$. Note

that in Equation (4), the summation is performed over the neighbors of each node. Considering that for non-neighbor nodes, $a_{ij} = 0$, we can simplify the sum by taking it over all nodes, implicitly incorporating only the neighbor nodes.

The update for the state, δS^t, leveraging the graph convolution, is expressed as follows:

$$\delta S^t = \mathcal{G}_\Theta(H^t). \tag{5}$$

This approach aligns well with the spatial dynamics of EEG signals. We refer to the process of updating the state of our model using Equations (2), (3) and (5) as the update cell of REST (Figure 1 - c).

3.4. Binary Random Mask: Continuous Dropout during Inference

To combat overfitting in deep neural networks, Dropout is commonly employed, randomly selecting model parameters during training and retaining all parameters during test-time (Srivastava et al., 2014). Drawing inspiration from a similar concept in Mordvintsev et al. (2020), we introduce Binary Masking for state updates, preventing overfitting while enabling the model to learn random state updates. This approach prevent the model to overfit as well as accelerates inference during test-time by skipping computations related to zero-masked feature points in the update. The state update will simply change as follows:

$$S^t = H^t + \delta S^t \odot B. \tag{6}$$

Here, $\odot$ denotes the Hadamard product, and $B \in \mathbb{R}^{Q \times N}$ is the binary mask with $B_{ij} \sim \mathcal{B}(p)$ from the Bernoulli distribution, where B_{ij} takes the value 1 with a probability of p and can be treated as hyperparameter for the model (more details about the masking during inference is located at Appendix N).

3.5. Multiple Update Mechanism: Escaping the Memory Requirements of Stacked RNN Layers

As widely recognized in neural networks, increasing the depth enhances performance by enabling the extraction of more general and complex features (Nakkiran et al., 2021). However, this poses a challenge in RNNs, where each additional layer increases memory requirements, not only for storing extra weights but also for additional gates and states.

In our study, we tackle this challenge by modifying REST to employ identical weights for state updates, thus facilitating multiple state updates. Although the graph convolution layer appears repetitive, the effect of binary random mask allows REST to learn to update a new part of the state during each iteration. This adaptation allows REST to align itself with the nature of these random updates, contributing to increased performance and enhanced stability without affecting memory requirements.

Thus, the Equations (2), (5) and (6) will be modified as follows:

$$H_i^t = WX^t + US_i^t, \tag{7}$$

$$S_{i+1}^t = H_i^t + \delta S_i^t \odot B. \tag{8}$$

Here, the index i denotes the current iteration during which the model updates its state, and $\delta S_i^t = \mathcal{G}_\Theta(H_i^t)$. It is crucial to emphasize X^t as the feature input at time point t to prevent the model from diverging into a state and neglecting the input during multiple updates (additional details are provided in the Appendix G). To update the state for the next time point, the final state obtained after multiple updates becomes the initial state. For instance, after updating the model's state I times at time point t, the initial state for the next time point $t+1$ is set as the final state after the last update at time point t ($S_0^{t+1} = S_I^t$). This enables the model to effectively capture the temporal dynamics across different time points. The proposed framework for the update cell is illustrated in (Figure 1c).

Moreover, previous studies (Mordvintsev et al., 2020; Pajouheshgar et al., 2023) have demonstrated that recurrently updating the state of neural networks, similar to REST in structure, for image and texture generation contributes to improved stability. We hypothesize that a similar enhancement can be achieved for seizure detection and classification.

4. REST & RNNs

To better understand the memory efficiency and speed advantages of REST during inference, we compare REST with traditional RNNs. As mentioned in Related Work, RNNs map the input $x(t)$ to a hidden state $h(t)$ and update this state over time using the previous state $h(t-1)$ and the current input $x(t)$. We highlight the efficiency and connections between REST and other types of RNNs through the following comparisons:

Single Update REST vs. Single-Layer RNN: First we consider a single GRU as a representative of RNN models, which leverages gating mechanisms to mitigate gradient vanishing. For a simple GRU update, we have the following set of equations:

$$r(t) = \sigma(W_r \cdot [h(t-1), x(t)]), \tag{9}$$

$$z(t) = \sigma(W_z \cdot [h(t-1), x(t)]), \tag{10}$$

$$\tilde{h}(t) = \tanh(W_h \cdot [r(t) \odot h(t-1), x(t)]), \tag{11}$$

$$h(t) = (1 - z(t)) \odot h(t-1) + z(t) \odot \tilde{h}(t). \tag{12}$$

Here, $h(t)$ is the hidden state at time t, $x(t)$ is the input at time t, σ is the sigmoid activation function, $\odot$ denotes

Table 2. Summary of TUSZ v.2.0.0 Train and Evaluation sets used in this study. Columns represent (from left to right): 1) total number of EEG files 2) total Number of patients 3) total number of generalized non-specific (GN) 4) tonic-clonic (TC) 5) absence (AB) 6) focal (FN), and 7) complex parietal (CP) seizure types in train and evaluation sets.

	EEG-Files (% Seizures)	Patients (% Seizures)	Seizure Type Numbers (Seizure Type Sessions)				
			GN	TC	AB	FN	CP
Train	4664(5.34%)	579(36%)	335(152)	30(11)	50(15)	1516(496)	279(132)
Evaluation	881(5.82%)	43(79%)	185(54)	57(8)	50(1)	240(98)	108(32)

Table 3. Summary of CHB-MIT Train and Evaluation sets used in this study.

	Patients	Seizures	Recording (hours)
Train	18	154	732
Evaluation	3	19	91
Test	3	19	92.5

element-wise multiplication, $[a, b]$ denotes the concatenation of vectors a and b, and W_r, W_z, W_h represent the weight matrices.

These equations describe how the hidden state $h(t)$ is updated over time based on the input and the preceding state. Unlike REST, GRU relies on three different gates $(z(t), r(t), \tilde{h}(t))$ for each state update, requiring twice as much memory as REST, in addition to the storage required for the weights utilized in generating these gates.

Despite GRU's memory demands, it not only needs to compute the next state ($h(t)$), but also three additional gates $(z(t), r(t), \tilde{h}(t))$ as the next state depends on these gates. In contrast, REST relies solely on the update result (δS^t), enabling it to rapidly derive the next state by adding it to the previous state, without the need for additional gates.

Multi Random Update REST vs. Multi-Layer RNN:

The remarkable efficiency of REST becomes particularly evident when comparing it with multi-layer RNN. In the context of multi-layer GRU, reaching the final state involves computing a set of equations (Equations (9) to (12)) for each layer. This process introduces three times more latency per layer, as each layer has three gates that must be computed to obtain the next state. Furthermore, it requires additional memory to store the hidden state of each layer, especially since it is required for updating the final hidden state of the last layer.

Contrastingly, REST distinguishes itself by reusing the same set of weights for the update cell and state evolution. This eliminates the need to store the previous state, as it evolves a distinct state over iterations. Consequently, REST maintains the same memory requirements as a single update, while delivering more accurate results (as discussed in the next section). It is worth mentioning that in the context of EEG data, all fully connected layers will be replaced by graph convolutions for both REST and GRU. For example, the combination of GRU with diffusion graph convolution for a traffic forecasting problem was undertaken by Li et al. (2017). We also applied REST update techniques to traditional RNNs, with the results presented in Appendix K.

Connection of REST Update Cell to Gating Mechanism:

As shown in Equation (12), the state update of RNNs, such as GRU, can be expressed as:

$$h(t) = h(t-1) + z(t) \odot \left(\tilde{h}(t) - h(t-1)\right). \quad (13)$$

This update shares similarities with the REST cell update in Equation (6). Instead of learning both $\tilde{h}(t)$ and $h(t)$ separately, the REST update directly learns $\tilde{h}(t) - h(t-1)$ as the residual update δS^t. Additionally, the update gate vector $z(t)$ is replaced with binary random masking. This substitution reduces the computational and memory overhead required for building $z(t)$ from the input $x(t)$ and hidden state $h(t)$.

5. Empirical Results

5.1. Setup

Dataset: We used two extensive publicly available datasets for the seizure detection and classification task: the Temple University Hospital EEG Seizure Corpus (TUSZ) (Obeid & Picone, 2016; Shah et al., 2018) and the Children's Hospital Boston (Goldberger et al., 2000) dataset. Below is a detailed description of each dataset:

TUSZ This dataset includes a total of 5545 EEG files for training and evaluation. These files encompass five different seizure types. We incorporated all 19 channels for all patients in the standard 10-20 system (Figure 1a).

CHB-MIT This dataset comprises recordings from 24 patients, with each patient having data from 9 to 42 sessions, recorded at a sampling rate of 256Hz. The dataset contains a total of 192 seizures. For our study, we included all 19 channels in the standard 10-20 system for the majority of patients, and excluded sessions that had fewer or a higher number of channels.

Preprocessing: In line with previous studies (Tang et al.,

Table 4. Summary of models for seizure detection on the TUSZ dataset. AUROC of different models is represented along with their memory demands and inference times.

	Seizure Detection AUROC (%)						Model Efficiency		
Model	4-s	6-s	8-s	10-s	12-s	14-s	Size(MB)	#Param	Inference(ms)
LSTM	$75.5_{\pm 0.3}$	$76.1_{\pm 0.07}$	$80.1_{\pm 0.3}$	$70.43_{\pm 0.02}$	$77.9_{\pm 0.06}$	$74.24_{\pm 0.2}$	2.147	536K	3.254
GRU	$76.1_{\pm 0.02}$	$78.8_{\pm 0.03}$	$73.2_{\pm 0.04}$	$73.5_{\pm 0.02}$	80.1 ± 0.1	$77.9_{\pm 0.04}$	1.61	402K	2.12
ResNet-LSTM	$79.1_{\pm 0.05}$	$80.1_{\pm 0.2}$	$75.6_{\pm 0.07}$	$74.3_{\pm 0.04}$	$78.8_{\pm 0.1}$	$80.0_{\pm 0.08}$	27.6	6.9M	6.78
ResNet-Dilation-LSTM	$80.2_{\pm 0.08}$	$76.5_{\pm 0.12}$	$75.9_{\pm 0.06}$	$73.6_{\pm 0.03}$	$77.4_{\pm 0.15}$	$78.2_{\pm 0.07}$	27.6	6.9M	6.78
CNN-LSTM	$81.3_{\pm 0.1}$	$78.5_{\pm 0.05}$	$76.4_{\pm 0.01}$	$75.4_{\pm 0.05}$	$75.05_{\pm 0.1}$	$74.0_{\pm 0.03}$	22.8	6M	5.624
DCRNN	$79.7_{\pm 0.01}$	$82.1_{\pm 0.04}$	$80.1_{\pm 0.04}$	$80.0_{\pm 0.06}$	$82.5_{\pm 0.1}$	$80.12_{\pm 0.04}$	0.884	126K	9.670
DCRNN w/SS	$\mathbf{83.0_{\pm 0.08}}$	$81.8_{\pm 0.05}$	$\mathbf{82.7_{\pm 0.1}}$	$82.1_{\pm 0.03}$	$85.6_{\pm 0.2}$	$84.0_{\pm 0.01}$	1.319	330K	23.25
Transformer	$83.0_{\pm 0.02}$	$82.1_{\pm 0.03}$	$82.2_{\pm 0.04}$	$\mathbf{85.5_{\pm 0.07}}$	$\mathbf{86.0_{\pm 0.03}}$	$\mathbf{85.1_{\pm 0.02}}$	0.80	120.3K	2.5
REST$_{(DS)}$	$75.3_{\pm 0.2}$	$67.0_{\pm 0.03}$	$72.2_{\pm 0.07}$	$74.1_{\pm 0.1}$	$70.6_{\pm 0.04}$	$70.0_{\pm 0.04}$	**0.037**	**8.4K**	**0.615**
REST$_{(RS)}$	$79.4_{\pm 0.03}$	$81.1_{\pm 0.01}$	$81.0_{\pm 0.08}$	$81.8_{\pm 0.02}$	$80.1_{\pm 0.1}$	$78.1_{\pm 0.4}$	**0.037**	**8.4K**	**0.710**
REST$_{(RM)}$	$82.4_{\pm 0.04}$	$\mathbf{82.2_{\pm 0.05}}$	$\mathbf{82.7_{\pm 0.1}}$	$83.6_{\pm 0.2}$	$83.4_{\pm 0.09}$	$82.0_{\pm 0.1}$	**0.037**	**8.4K**	**1.292**

Table 5. Summary of models for seizure detection on the CHB-MIT dataset. AUROC of different models is represented along with their memory demands and inference times.

	Seizure Detection AUROC (%)					Model Efficiency		
Model	4-s	6-s	8-s	10-s	12-s	Size(MB)	#Param	Inference(ms)
LSTM	$85.5_{\pm 0.2}$	$84.1_{\pm 0.4}$	$81.0_{\pm 0.2}$	$75.2_{\pm 0.03}$	$73.5_{\pm 0.08}$	2.691	627K	3.56
GRU	$76.1_{\pm 0.3}$	$78.8_{\pm 0.03}$	$73.2_{\pm 0.4}$	$73.5_{\pm 0.01}$	$80.1_{\pm 0.2}$	1.92	553K	2.42
ResNet-LSTM	$77.6_{\pm 0.2}$	$82.1_{\pm 0.14}$	$79.9_{\pm 0.3}$	$76.8_{\pm 0.4}$	$81.4_{\pm 0.17}$	29.1	7.2M	6.84
ResNet-Dilation-LSTM	$78.2_{\pm 0.03}$	$79.8_{\pm 0.1}$	$82.3_{\pm 0.4}$	$77.6_{\pm 0.4}$	$81.2_{\pm 0.1}$	29.1	7.2M	6.84
CNN-LSTM	$86.2_{\pm 0.4}$	$84.9_{\pm 0.2}$	$80.4_{\pm 0.04}$	$80.35_{\pm 0.06}$	$77.6_{\pm 0.3}$	7.6M	30.23	6.432
DCRNN	$88.7_{\pm 0.3}$	$80.0_{\pm 0.02}$	$86.8_{\pm 0.06}$	$88.8_{\pm 0.3}$	$86.5_{\pm 0.3}$	0.591	147K	9.80
Transformer	$80.1_{\pm 0.2}$	$82.3_{\pm 0.6}$	$82.2_{\pm 0.04}$	$85.5_{\pm 0.01}$	$86_{\pm 0.17}$	0.25	52.4K	6.00
REST$_{(DS)}$	$89.1_{\pm 0.2}$	$88.5_{\pm 0.08}$	$90.1_{\pm 0.1}$	$86.3_{\pm 0.03}$	$87.8_{\pm 0.5}$	**0.037**	**9.3K**	**1.314**
REST$_{(RS)}$	$92.3_{\pm 0.1}$	$88.7_{\pm 0.06}$	$\mathbf{92.1_{\pm 0.03}}$	$\mathbf{93.5_{\pm 0.02}}$	$91.5_{\pm 0.02}$	**0.037**	**9.3K**	**1.314**
REST$_{(RM)}$	$\mathbf{96.7_{\pm 0.2}}$	$\mathbf{92.3_{\pm 0.04}}$	$91.4_{\pm 0.1}$	$89.2_{\pm 0.4}$	$\mathbf{91.6_{\pm 0.03}}$	**0.037**	**9.3K**	**1.314**

2021; Saab et al., 2020), we resample the EEG signals from TUSZ dataset into 200Hz (256Hz for CHB-MIT dataset) to have consistent sampling frequency among different EEGs. Then, we extract non-overlapping window sizes with length T leading to an EEG clip $X \in \mathbb{R}^{T \times L \times N}$ with $N = 19$ nodes, $L = 200$ ($L = 256$ for CHB-MIT dataset) features per node, and T time points. After applying the fast Fourier transform on the second dimension of the EEG clip and choosing the log amplitude of non-negative frequency components, the final EEG clip fused as the input to the models is $X \in \mathbb{R}^{T \times M \times N}$ where $M = 100$ ($M = 128$ for CHB-MIT dataset). Finally, the features for each node and time point are z-normalized using the mean and variance calculated from 100 (128 for CHB-MIT dataset) feature points along its axis. We examine the presence of a seizure within an EEG clip in the detection task. For classification, we start analyzing each clip 2 seconds before the seizure begins and evaluate the outcomes within a clip duration of $T = 10$ seconds. This approach aligns with the annotations of seizure onset, as demonstrated in previous works (Ahmedt-Aristizabal et al., 2020; Tang et al., 2021).

We evaluate models' ability to perform detection tasks across a range of window sizes, spanning from {4,6,8,10,12,14} seconds for TUSZ and {4,6,8,10,12} seconds for CHB-MIT. This allows us to evaluate their performance in both short and long-term detection scenarios. For seizure detection task, we used both the seizure and background data, while for the classification task, only the seizure data were used (details in Appendix A).

Train-Evaluation Split: The original TUSZ Train-set was randomly split into training and validation sets with a ratio of 90/10. The TUSZ eval set served as a standardized evaluation set, consistent with previous studies Tang et al. (2021). Further details regarding the data split are provided in Table 2. For the CHB-MIT dataset, since predefined splits for training, evaluation, and testing are not provided, we randomly selected 80% of the data for training, 10% for evaluation, and 10% for testing. We ensured that patients in each set are unique, preventing the model from being tested on patients included in the training set (details at Table 3).

Baselines: To evaluate performance and runtime, we implemented three key baselines widely used in seizure analysis: **DCRNN** (Tang et al., 2021), with two versions of the model, with and without self-supervision; **CNN-LSTM** (Ahmedt-Aristizabal et al., 2020); **LSTM** (Hochreiter & Schmidhuber, 1997); **Transformer** (Vaswani et al., 2017); **GRU** (Cho et al., 2014); and two versions of the **ResNet-LSTM** model as described in Lee et al. (2022).

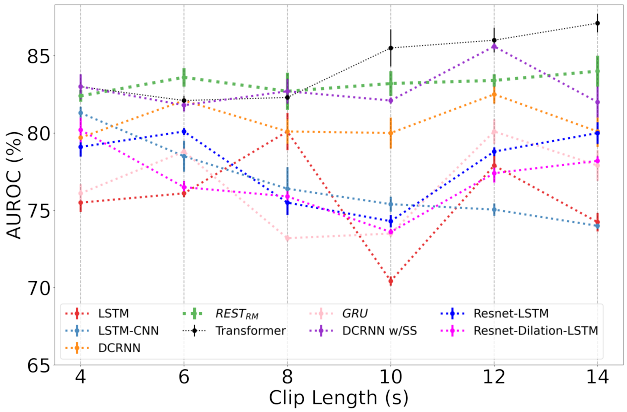

Figure 2. AUROC comparison among various models for seizure detection across different clip lengths on TUSZ dataset. A flatter line indicates more consistent performance, with error bars representing variation across five random seeds. Higher values on the y-axis correspond to increased accuracy. $\text{REST}_{(\text{RM})}$ is shown as bold green line to emphasise its stability.

REST architecture and training: REST was designed with two graph convolution layers for state updates, the first employing ReLU activation and the second utilizing a linear activation function (Figure 1c). We evaluate various versions of REST: a) $\text{REST}_{(DS)}$ with a single deterministic update without any masking, b) $\text{REST}_{(RS)}$ with a single random update (utilizing binary random masking), and c) $\text{REST}_{(RM)}$ with multiple random updates.

In the seizure detection task, both Binary Cross Entropy and Mean Squared Error (MSE) loss were employed, with MSE outperforming Binary Cross Entropy. This result stems from the observation that Binary Cross Entropy prevents residual updates from approaching zero (more details on Appendix E). For seizure classification, the Cross-Entropy loss was utilized.

Table 6. Classification Performance, model size and parameter count for different models under the clip length of 10-s.

Model	F1-Score	Size(MB)	Parameter(#)
LSTM	0.39	2.021	512K
GRU	0.44	1.92	553K
ResNet-LSTM	0.58	30.3	7.5M
ResNet-LSTM-Dilation	0.50	30.3	7.5M
CNN-LSTM	0.47	23.9	6M
DCRNN	0.54	0.506	126K
DCRNN w/SS	**0.62**	1.40	332K
Transformer	0.54	0.25	53K
$\text{REST}_{(\text{DS})}$	0.51	**0.034**	**8.6K**
$\text{REST}_{(\text{RS})}$	0.57	**0.034**	**8.6K**
$\text{REST}_{(\text{RM})}$	0.60	**0.034**	**8.6K**

We trained all models with 5 different random seeds and averaged the performance on evaluation set over different runs. We utilized ADAM (Kingma & Ba, 2014) to optimize the models' parameters, conducting training on a single NVIDIA A100 GPU with a batch size of 128 EEG clips. Training times for all models across various clip lengths can be found in the Appendix F.

Runtime Comparison: To ensure a fair comparison between different models, we adopted the following approach for each model: We selected the optimal set of hyperparameters for each clip length based on performance on the validation set. Here, inference time refers to the time required for each model to provide predictions for one sample of the test data, where each sample is an EEG clip with length $T \in \{4, 6, 8, 10, 12, 14\}$. We also attempted to shrink the baselines while maintain the same accuracy for both tasks and the details are reported in Appendix I.

5.2. Experimental Results

Seizure Detection and Classification Accuracy: We evaluated the performance of all baseline models and REST using the Area Under the Receiver Operating Characteristic Curve (AUROC) for seizure detection and Weighted F1-Score for seizure classification. Our model surpassed all baselines significantly on the CHB-MIT dataset for all different clip lengths. For the TUSZ dataset, it achieved very close detection AUROC scores for all clip lengths compared to DCRNN with self-supervision and the Transformer, while outperforming them at clip lengths of 6 and 8 seconds. Figure 2 suggests that multiple random updates improve the

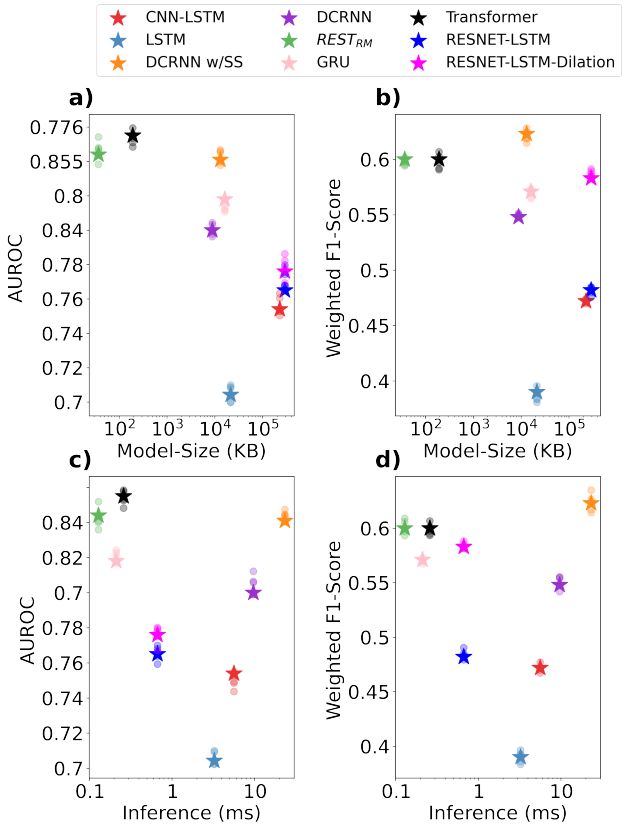

Figure 3. Performance comparison in seizure analysis across models on TUSZ dataset: **a)** Seizure detection AUROC vs. Model size. **b)** Seizure classification weighted F1-score vs. model size. **c)** Seizure detection AUROC vs. inference. **d)** Seizure classification weighted F1-score vs. inference. The •s represents the accuracy on evaluation set for different train/validation splits and ⋆s represent the mean accuracy across different train/validation splits.

stability of REST as it leads to higher and more consistent performance compared to other models. According to Figure 2, REST$_{(RM)}$ and DCRNN with self supervision exhibit more stable performance over time across clip lengths, yielding consistent results. Interestingly, CNN-LSTM achieved higher performance in a small clip size of 4s, surpassing DCRNN with graph convolution layers.

REST **Enjoys an Exponentially Smaller Size:** While maintaining high accuracy, REST exhibits a size that is 14× smaller than the smallest existing model for seizure detection and classification on TUSZ dataset. Table 4 highlights that REST requires 38× fewer parameters than state-of-the-art models (DCRNN w/SS) and over 697× fewer parameters than the deep CNN-LSTM model for seizure analysis.

Figure 3 a-b showcases REST's outstanding performance, achieving an AUROC of 83.6% for seizure detection with a clip length of 10 seconds. Additionally, REST secures the second-highest F1-Score for seizure classification, trailing only 2% below DCRNN w/SS but with a significantly smaller size than all other baselines. The substantial gap between REST's size and the sizes of other baselines, depicted on the logarithmic scale in Figure 3 a-b, underscores REST's remarkable size advantage and potential for implementation on edge devices. The graph convolution layers in REST efficiently capture both short and long-range communication between nodes, ensuring high accuracy with a compact model size. Moreover, using identical weights for multiple random updates eliminates the need for additional layers while enhancing the model's accuracy and memory efficiency.

Rapid Seizure Detection: REST$_{(RM)}$ achieves the fastest inference speed among all models, being 20× faster than DCRNN w/SS and 9× faster than DCRNN during inference, with only a minor AUROC drop of less than 2% for seizure detection across various clip lengths for TUSZ dataset. Moreover, REST, with multiple updates, requires only 1.292 ms for seizure detection, which is three times faster than the fastest baseline, LSTM, while being 13% more accurate in delivering predictions (at 10-s clip length). On the CHB-MIT dataset, REST outperforms all other baselines in the seizure detection task, being the only model with an AUROC higher than 90%. It also significantly outperforms other baselines for the short clip length of 4 seconds, which is crucial for real-time seizure detection (Zhu et al., 2021).

In seizure classification, REST$_{(RM)}$ secures the second-highest F1-Score (Table 6) and excels in providing the fastest classification result within 1.51 ms (Figure 3 c-d). Notably, it is three times faster than LSTM, while achieving 21% higher accuracy than LSTM. The swift prediction capability of our model is attributed to its efficient design. REST relies on a single affine mapping into the state space, complemented by two computationally lightweight graph convolutions.

6. Conclusion

In this work, we propose REST, a graph-based residual state update mechanism for efficient seizure detection and classification tasks. Our model effectively captures both spatial and temporal behaviors of EEG signals, achieving state-of-the-art performance in seizure detection and classification. With its shallow structure, REST boasts a fast inference speed, making it 9 times faster than current models with a comparable performance. Furthermore, REST exhibits remarkable efficiency, requiring only 37KB of memory, which is 14 times smaller than smallest existing models for seizure analysis tasks. These advancements position REST as a promising model for implementation on small, low-power edge devices, particularly for applications in epilepsy treatments like DBS and RNS.

Impact Statement

The EEG Seizure Corpus from Temple University Hospital, utilized in our research, is anonymized and publicly accessible with IRB approval (Obeid & Picone, 2016; Shah et al., 2018). The authors declare no conflicts of interest, and the seizure detection and classification models presented in this study do not provide any harmful insights. Although our model has demonstrated accuracy in real-time seizure analyses, further experiments are essential for real-world application and implementation on edge devices, as demonstrated in a number of recent systems (Shoaran et al., 2018; Shin et al., 2022; Shaeri et al., 2024). These evaluations should encompass testing with diverse datasets from various patient populations and hospitals. Additionally, assessing the model's energy efficiency is crucial to ensure its safety for chronic use, along with obtaining neurologists' approval regarding its neurological aspects for deployment in such devices.

Acknowledgements

This work was supported in part by the Swiss State Secretariat for Education, Research and Innovation under Contract number SCR0548363, in part by the Wyss project under contract number 532932, in part by Hasler Foundation Program: Hasler Responsible AI project number 21043, in part by the Army Research Office under grant number W911NF-24-1-0048, and in part by the Swiss National Science Foundation (SNSF) under grant number 200021_205011. Moreover, we appreciate the reviewers for their insightful feedback, which has significantly enhanced the robustness and clarity of our results.

References

Acharya, J. N., Hani, A. J., Thirumala, P., and Tsuchida, T. N. American clinical neurophysiology society guideline 3: a proposal for standard montages to be used in clinical eeg. *The Neurodiagnostic Journal*, 56(4):253–260, 2016.

Ahmedt-Aristizabal, D., Fernando, T., Denman, S., Petersson, L., Aburn, M. J., and Fookes, C. Neural memory networks for seizure type classification. In *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*, pp. 569–575. IEEE, 2020.

Asif, U., Roy, S., Tang, J., and Harrer, S. Seizurenet: Multispectral deep feature learning for seizure type classification. In *Machine Learning in Clinical Neuroimaging and Radiogenomics in Neuro-oncology: Third International Workshop, MLCN 2020, and Second International Workshop, RNO-AI 2020, Held in Conjunction with MICCAI 2020, Lima, Peru, October 4–8, 2020, Proceedings 3*, pp. 77–87. Springer, 2020.

Beghi, E., Giussani, G., Nichols, E., Abd-Allah, F., Abdela, J., Abdelalim, A., Abraha, H. N., Adib, M. G., Agrawal, S., Alahdab, F., et al. Global, regional, and national burden of epilepsy, 1990–2016: a systematic analysis for the global burden of disease study 2016. *The Lancet Neurology*, 18(4):357–375, 2019.

Cho, K., Van Merriënboer, B., Gulcehre, C., Bahdanau, D., Bougares, F., Schwenk, H., and Bengio, Y. Learning phrase representations using rnn encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078*, 2014.

Christou, V., Miltiadous, A., Tsoulos, I., Karvounis, E., Tzimourta, K. D., Tsipouras, M. G., Anastasopoulos, N., Tzallas, A. T., and Giannakeas, N. Evaluating the window size's role in automatic eeg epilepsy detection. *Sensors*, 22(23):9233, 2022.

Covert, I. C., Krishnan, B., Najm, I., Zhan, J., Shore, M., Hixson, J., and Po, M. J. Temporal graph convolutional networks for automatic seizure detection. In *Machine Learning for Healthcare Conference*, pp. 160–180. PMLR, 2019.

Fisher, R. S. and Velasco, A. L. Electrical brain stimulation for epilepsy. *Nature Reviews Neurology*, 10(5):261–270, 2014a.

Fisher, R. S. and Velasco, A. L. Electrical brain stimulation for epilepsy. *Nature Reviews Neurology*, 10(5):261–270, 2014b.

Goldberger, A. L., Amaral, L. A., Glass, L., Hausdorff, J. M., Ivanov, P. C., Mark, R. G., Mietus, J. E., Moody, G. B., Peng, C.-K., and Stanley, H. E. Physiobank, physiotoolkit, and physionet: components of a new research resource for complex physiologic signals. *circulation*, 101(23): e215–e220, 2000.

Gotman, J. Automatic seizure detection: improvements and evaluation. *Electroencephalography and clinical Neurophysiology*, 76(4):317–324, 1990.

Harrer, S., Shah, P., Antony, B., and Hu, J. Artificial intelligence for clinical trial design. *Trends in pharmacological sciences*, 40(8):577–591, 2019.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 770–778, 2016.

Ho, T. K. K. and Armanfard, N. Self-supervised learning for anomalous channel detection in eeg graphs: application to

seizure analysis. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 37, pp. 7866–7874, 2023.

Hochreiter, S. and Schmidhuber, J. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

Iešmantas, T. and Alzbutas, R. Convolutional neural network for detection and classification of seizures in clinical data. *Medical & Biological Engineering & Computing*, 58:1919–1932, 2020.

Jasper, H. H. Ten-twenty electrode system of the international federation. *Electroencephalogr Clin Neurophysiol*, 10:371–375, 1958.

Kingma, D. P. and Ba, J. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*, 2014.

Lee, K., Jeong, H., Kim, S., Yang, D., Kang, H.-C., and Choi, E. Real-time seizure detection using eeg: a comprehensive comparison of recent approaches under a realistic setting. *arXiv preprint arXiv:2201.08780*, 2022.

Li, Y., Yu, R., Shahabi, C., and Liu, Y. Graph convolutional recurrent neural network: Data-driven traffic forecasting. *arXiv preprint arXiv:1707.01926*, 7(8), 2017.

Li, Z., Hwang, K., Li, K., Wu, J., and Ji, T. Graph-generative neural network for eeg-based epileptic seizure detection via discovery of dynamic brain functional connectivity. *Scientific Reports*, 12(1):18998, 2022.

Loshchilov, I. and Hutter, F. Sgdr: Stochastic gradient descent with warm restarts. *arXiv preprint arXiv:1608.03983*, 2016.

Mordvintsev, A., Randazzo, E., Niklasson, E., and Levin, M. Growing neural cellular automata. *Distill*, 5(2):e23, 2020.

Morris, C., Ritzert, M., Fey, M., Hamilton, W. L., Lenssen, J. E., Rattan, G., and Grohe, M. Weisfeiler and leman go neural: Higher-order graph neural networks. In *Proceedings of the AAAI conference on artificial intelligence*, volume 33, pp. 4602–4609, 2019.

Nakkiran, P., Kaplun, G., Bansal, Y., Yang, T., Barak, B., and Sutskever, I. Deep double descent: Where bigger models and more data hurt. *Journal of Statistical Mechanics: Theory and Experiment*, 2021(12):124003, 2021.

Obeid, I. and Picone, J. The temple university hospital eeg data corpus. *Frontiers in neuroscience*, 10:196, 2016.

O'Shea, A., Lightbody, G., Boylan, G., and Temko, A. Neonatal seizure detection from raw multi-channel eeg using a fully convolutional architecture. *Neural Networks*, 123:12–25, 2020.

Pajouheshgar, E., Xu, Y., Zhang, T., and Süsstrunk, S. Dynca: Real-time dynamic texture synthesis using neural cellular automata. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pp. 20742–20751, 2023.

Pascanu, R., Mikolov, T., and Bengio, Y. On the difficulty of training recurrent neural networks. In *International conference on machine learning*, pp. 1310–1318. Pmlr, 2013.

Randazzo, E., Mordvintsev, A., Niklasson, E., Levin, M., and Greydanus, S. Self-classifying mnist digits. *Distill*, 5(8):e00027–002, 2020.

Saab, K., Dunnmon, J., Ré, C., Rubin, D., and Lee-Messer, C. Weak supervision as an efficient approach for automated seizure detection in electroencephalography. *NPJ digital medicine*, 3(1):59, 2020.

Shaeri, M. A., Shin, U., Yadav, A., Caramellino, R., Rainer, G., and Shoaran, M. 33.3 mibmi: A 192/512-channel 2.46 mm^2 miniaturized brain-machine interface chipset enabling 31-class brain-to-text conversion through distinctive neural codes. In *2024 IEEE International Solid-State Circuits Conference (ISSCC)*, volume 67, pp. 546–548. IEEE, 2024.

Shah, V., Von Weltin, E., Lopez, S., McHugh, J. R., Veloso, L., Golmohammadi, M., Obeid, I., and Picone, J. The temple university hospital seizure detection corpus. *Frontiers in neuroinformatics*, 12:83, 2018.

Shin, U., Ding, C., Zhu, B., Vyza, Y., Trouillet, A., Revol, E. C., Lacour, S. P., and Shoaran, M. Neuraltree: A 256-channel 0.227-μj/class versatile neural activity classification and closed-loop neuromodulation soc. *IEEE Journal of Solid-State Circuits*, 57(11):3243–3257, 2022.

Shoaran, M., Shahshahani, M., Farivar, M., Almajano, J., Shahshahani, A., Schmid, A., Bragin, A., Leblebici, Y., and Emami, A. A 16-channel 1.1 mm 2 implantable seizure control soc with sub-μw/channel consumption and closed-loop stimulation in 0.18 μm cmos. In *2016 IEEE Symposium on VLSI Circuits (VLSI-Circuits)*, pp. 1–2. Ieee, 2016.

Shoaran, M., Haghi, B. A., Taghavi, M., Farivar, M., and Emami-Neyestanak, A. Energy-efficient classification for resource-constrained biomedical applications. *IEEE Journal on Emerging and Selected Topics in Circuits and Systems*, 8(4):693–707, 2018.

Shuman, D. I., Narang, S. K., Frossard, P., Ortega, A., and Vandergheynst, P. The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. *IEEE signal processing magazine*, 30(3):83–98, 2013.

Siddiqui, M. K., Morales-Menendez, R., Huang, X., and Hussain, N. A review of epileptic seizure detection using machine learning classifiers. *Brain informatics*, 7(1): 1–18, 2020.

Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., and Salakhutdinov, R. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

Strein, M., Holton-Burke, J. P., Smith, L. R., and Brophy, G. M. Prevention, treatment, and monitoring of seizures in the intensive care unit. *Journal of Clinical Medicine*, 8 (8):1177, 2019.

Sun, F. T. and Morrell, M. J. The rns system: responsive cortical stimulation for the treatment of refractory partial epilepsy. *Expert review of medical devices*, 11(6):563–572, 2014.

Tang, S., Dunnmon, J. A., Saab, K., Zhang, X., Huang, Q., Dubost, F., Rubin, D. L., and Lee-Messer, C. Self-supervised graph neural networks for improved electroencephalographic seizure analysis. *arXiv preprint arXiv:2104.08336*, 2021.

Thodoroff, P., Pineau, J., and Lim, A. Learning robust features using deep learning for automatic seizure detection. In *Machine learning for healthcare conference*, pp. 178–190. PMLR, 2016.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention is all you need. *Advances in neural information processing systems*, 30, 2017.

Voulodimos, A., Doulamis, N., Doulamis, A., Protopapadakis, E., et al. Deep learning for computer vision: A brief review. *Computational intelligence and neuroscience*, 2018, 2018.

Yan, J., Li, J., Xu, H., Yu, Y., and Xu, T. Seizure prediction based on transformer using scalp electroencephalogram. *Applied Sciences*, 12(9):4158, 2022a.

Yan, J., Li, J., Xu, H., Yu, Y., and Xu, T. Seizure prediction based on transformer using scalp electroencephalogram. *Applied Sciences*, 12(9):4158, 2022b.

Zhu, B., Farivar, M., and Shoaran, M. Resot: Resource-efficient oblique trees for neural signal classification. *IEEE Transactions on Biomedical Circuits and Systems*, 14(4):692–704, 2020.

Zhu, B., Shin, U., and Shoaran, M. Closed-loop neural prostheses with on-chip intelligence: A review and a low-latency machine learning model for brain state detection. *IEEE transactions on biomedical circuits and systems*, 15 (5):877–897, 2021.

Appendix Introduction

The Appendix is organised as followes:

- Preprocessing details are outlined in Appendix A.
- The mathematical proof addressing the avoidance of gradient vanishing in our model is provided in Appendix B.
- Seizure analyses results are presented in Appendix C.
- Hyperparameter selection and training details for all models are discussed in Appendix D.
- The impact of BCE and MSE loss on training REST is compared in Appendix E.
- Training times are documented in Appendix F.
- Details explaining how REST avoids overfitting are shown in Appendix G.
- Differences between various graph structures are explored in Appendix H.
- Information about baseline compression is provided in Appendix I.
- F1-scores for seizure detection are presented in Appendix J.
- The effectiveness of binary random masking on different RNN variants is shown in Appendix K.
- Size comparisons for models with the same number of neurons are provided in Appendix L.
- Real-time evaluations of different models with overlapping windows are detailed in Appendix M.
- An ablation study on the inference performance of REST with and without binary random masking is presented in Appendix N.

A. Details of Preprocessing

We initially performed general preprocessing on the EEG data followed by specific steps for each detection and classification tasks:

A.1. TUSZ dataset

General Preprocessing: The EEG signals in the TUH EEG Corpus (TUSZ) dataset were initially sampled at various frequencies. As a part of the preprocessing pipeline, all signals were uniformly resampled to 200 Hz. Subsequently, EEG clips were extracted using the natural choice of one-second, non-overlapping windows, resulting in an EEG tensor $X \in \mathbb{R}^{T \times L \times N}$, where T represents clip lengths (ranging from 4, 6, 8, 10, 12, to 14 seconds), N is the number of electrodes (19), and L is the number of time samples (200). To harness the effectiveness of Fourier transform for neural EEG recordings, fast Fourier transform was applied to extract frequency components for each node at each time point. The log-amplitude of the frequencies was then computed and only non-negative frequency components were extracted similar to prior studies (Tang et al., 2021; Ahmedt-Aristizabal et al., 2020) leading to EEG clip tensor of $X \in \mathbb{R}^{T \times M \times N}$ with M=100. Last, we have z-normalized the EEG clips across their second dimension for further analyses.

Preprocessing for Seizure Detection: For seizure detection after extracting EEG clips from the entire training set consisting of 5545 sessions, a binary label was assigned, with $y = 1$ indicating the presence of at least one seizure within the clip and $y = 0$ otherwise. To handle the issue of a substantial number of background clips in the dataset, non-seizure clips were randomly selected to achieve a balanced representation with seizure clips in the training data. Also, the last clip was dropped for each EEG data if the recording ends before the clip could reach it's length.

Preprocessing for Seizure Classification: For seizure classification followed by Tang et al. (2021); Ahmedt-Aristizabal et al. (2020) we have removed the background data and only processed the seizure clips. We have started 2 seconds before the annotated seizure for tolerance in the annotations. Then we have labeled the clip $y = 0$ for general non-specific (GN), $y = 1$ for combined tonic (TC), $y = 2$ for absence (AB), $y = 3$ for focal, and $y = 4$ for complex parietal (CP) seizures.

Moreover, if seizure event is shorter than the clip length we have truncated the clip to avoid having multiple seizures in one clip. Also, it is noteworthy that while the training set included simple partial seizures, these seizure types were absent in the evaluation set. Therefore, we excluded simple parietal seizures from the classification task. Also, because the clips for seizure classification may have different lengths we pad 0's to the end of the clip to assure all samples share the same length.

A.2. CHB-MIT Dataset

For the CHB-MIT dataset, we randomly selected 18 patients for training, 3 for evaluation, and 3 for testing. We followed the same preprocessing pipeline as described for the TUSZ dataset, with the exception of maintaining a uniform sampling rate of 256Hz for all patients. For each 1-second time window, we have 256 samples of raw EEG data per channel. The number of channels is consistent with the TUSZ dataset, comprising 19 channels, and we excluded any sessions with a different number of channels.

We utilized the same frequency domain components for seizure detection. Unlike the TUSZ dataset, the CHB-MIT dataset does not include seizure types for classification. The results are reported based on five different random seeds for the train/test/evaluation splits (more details at Table 7).

Table 7. Details of sessions and number of seizures for each patient at CHB-MIT dataset.

Case	Number of Seizures	Number of Sessions	Age
1	7	24	11
2	3	36	11
3	7	38	14
4	4	42	22
5	5	39	7
6	10	18	1.5
7	3	19	14.5
8	5	20	3.5
9	4	19	10
10	7	25	3
11	3	35	12
12	27	24	2
13	10	33	3
14	8	26	9
15	20	40	16
16	8	19	7
17	3	21	12
18	6	36	18
19	3	30	19
20	8	29	6
21	4	33	13
22	3	31	9
23	7	9	6
24	16	22	Unknown

B. Preventing Gradient Vanishing with Residual Update

In equations Equations (3) to (5), the model's state is updated using a residual state update. When we take the derivative of S^{t-1} concerning the forward propagation of Equation (3), we get:

$$\frac{\partial \mathcal{L}}{\partial S^{t-1}} = \frac{\partial \mathcal{L}}{\partial S^{t}} \frac{\partial S^{t}}{\partial S^{t-1}} = \frac{\partial \mathcal{L}}{\partial S^{t}} \left(1 + \frac{\partial \delta S^{t}}{\partial S^{t-1}}\right) = \frac{\partial \mathcal{L}}{\partial S^{t}} + \frac{\partial \mathcal{L}}{\partial S^{t}} \frac{\partial \delta S^{t}}{\partial S^{t-1}}. \tag{14}$$

Here the $\mathcal{L}$ is the loss function to be minimized. This equation shows that the gradient of the previous state S^{t-1} always has a term $\frac{\partial \mathcal{L}}{\partial S^t}$ directly added. This helps prevent the gradients of $\frac{\partial \mathcal{L}}{\partial S^{t-1}}$ from becoming too small, even when the gradients of the previous updates are small, i.e., $\frac{\partial \mathcal{L}}{\partial S^t}\frac{\partial \delta S^t}{\partial S^{t-1}}$.

C. ROC Curves and Confusion Matrices for Different Clip Lengths

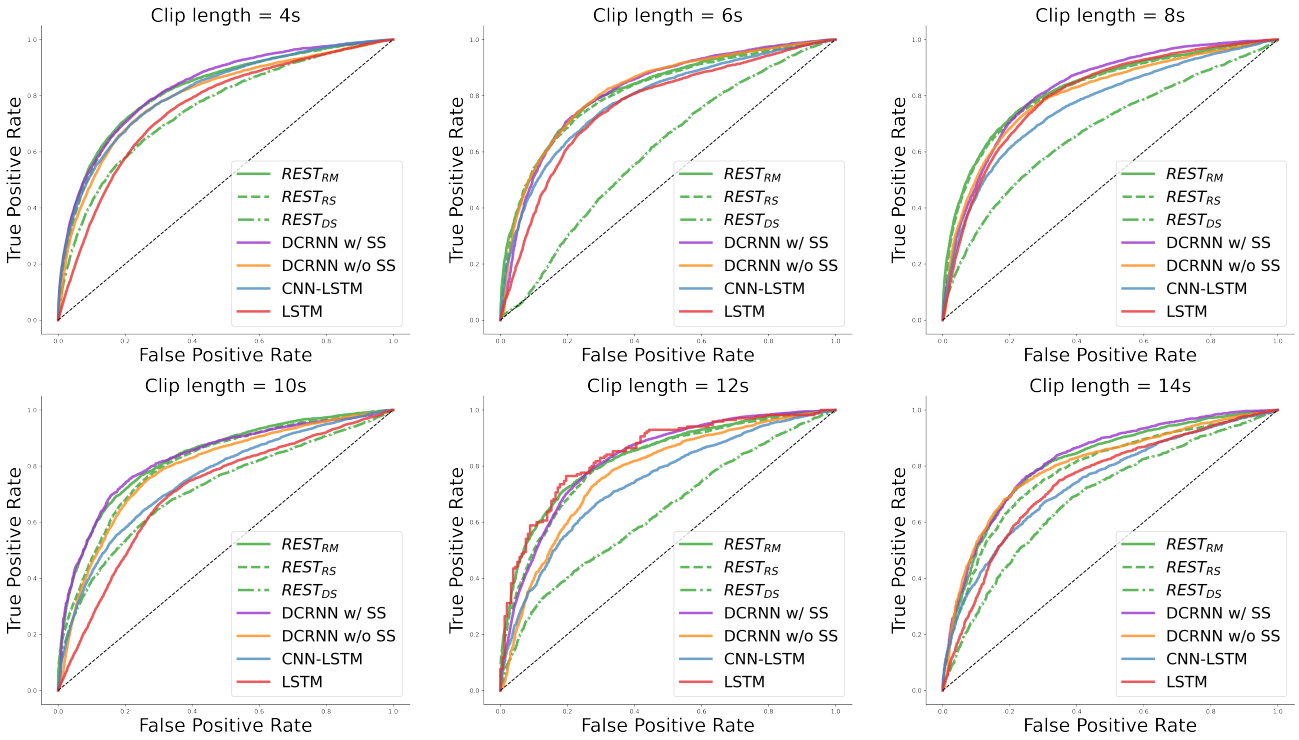

Figure 4. ROC curves for different clip lengths among REST and baselines for TUSZ dataset.

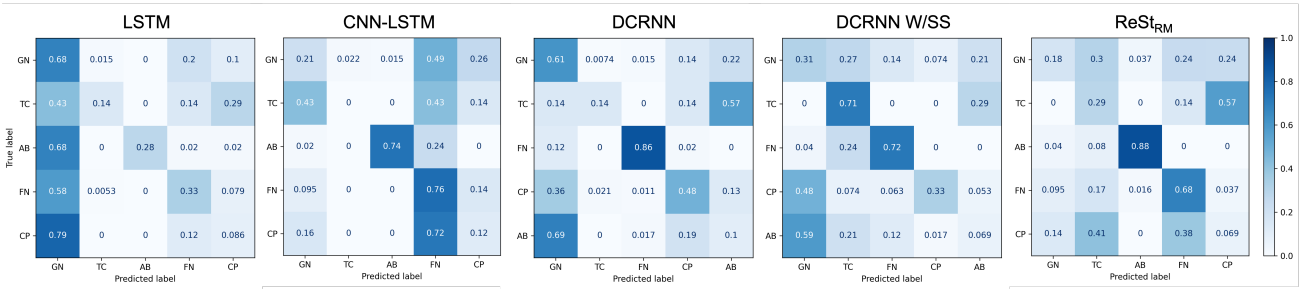

Figure 5. Confusion Matrices for seizure classification task among different models.

D. Model Training and Hyperparameter Selection Details

Here are the details of training and hyperparameter selection for REST and baselines:

REST **Hyperparameters:** We optimized the following hyperparameters for REST based on the lowest validation error: a) Number of neurons in each graph convolution layer within the range [16, 32, 64]; b) Initial learning rate within the range [5e-4, 1e-4]; c) Success probability of the random binary mask within [0.1, 0.3, 0.5, 0.7, 1]. For multi-update REST, the number of updates for each time point was randomly selected an integer from the interval [1, 10]. We conducted training for 500 epochs using a Multistep learning rate scheduler. Five experiments were run in PyTorch with different random seeds.

DCRNN: We followed the hyperparameter tuning strategy from the original paper (Tang et al., 2021) for both DCRNN with and without self-supervision tasks. The hyperparameter search on the validation set included: a) Initial learning rate within the range [5e-5, 1e-3]; b) Number of Diffusion Convolutional Gated Recurrent Units (DCGRU) layers within the range $\{2, 3, 4, 5\}$ and hidden units within the range $\{32, 64, 128\}$; c) Maximum diffusion step $K \in \{2, 3, 4\}$; d) Dropout probability in the last fully connected layer. For self-supervised pre-training, we utilized mean absolute error (MAE) as the loss function. The models underwent training for 350 epochs with an initial learning rate of 5e-4, employing a maximum diffusion step of 1 and 64 hidden units in both the encoder and decoder. Moreover, cosine annealing learning rate scheduler (Loshchilov & Hutter, 2016) was used as scheduler.

CNN-LSTM: For the baseline CNN-LSTM, we adopt the identical model architecture outlined in Ahmedt-Aristizabal et al. (2020). This configuration employes two stacked convolutional layers with 32 kernels of size 3×3, one max-pooling layer of size 2×2, one fully-connected layer with an output neuron count of 512, two stacked LSTM layers with a hidden size of 128, and one additional fully connected layer.

LSTM: We employed two stacked RNN layers, each with 64 hidden units, and an additional fully connected layer for the final prediction.

GRU: For the GRU model, we used same number of layers and hidden units as LSTM.

ResNet-LSTM: We followed two versions with and without dilation described at Lee et al. (2022).

Transformer: We implemented a two-layer multi-head attention mechanism with 64 embedding dimensions and 16 heads for the transformer architecture. Additionally, we utilized time positional encoding as introduced by Vaswani et al. (2017) for the original positional encoding.

For detection task for all models binary cross entropy loss was used exept for REST which MSE performs slightly higher during the validation step. For classification task weighted binary cross entropy was employed due to the highly imbalancy among different seizure types.

E. Comparison Between MSE and BCE loss for Training REST

REST was trained for seizure detection using both MSE and Binary Cross Entropy (BCE) loss functions. However, MSE outperformed BCE in terms of stability and accuracy. This advantage is attributed to BCE's tendency for unbounded growth in classification logits, hindering residual updates and message passing between graph nodes, particularly in multi-update scenarios, as discussed in Randazzo et al. (2020). As shown in Figure 4 MSE has less fluctuations and more stability in validation error during training compared to BCE loss when training REST with multiple updates.

Table 8. Detection Performance of REST with BCE and MSE loss functions for clip length of 10s on validation set.

Seizure Detection performance	
Loss Function	AUROC
REST BCE	80.4
REST MSE	**83.6**

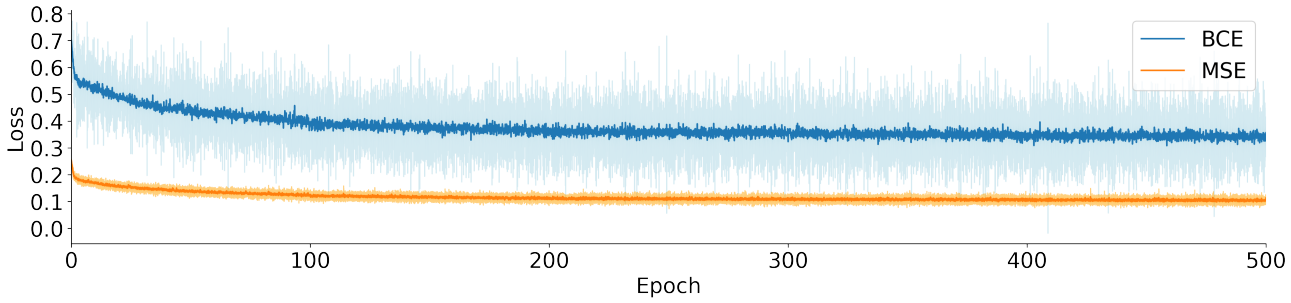

Figure 6. Validation loss of the model for BCE and MSE loss functions.

F. Training Time

Bellow we report the time needed for training each model (Table 9). All the models were trained on the same NVIDIA A100 GPU and the number of parameters and model size has reported at Tables 4 and 6. REST requires more training time to adopt itself and converge to a stable point specially to adapt its update cell with multiple random updates

Table 9. Train time for seizure detection and classification tasks for different models among different clip lengths. Times are reported in minutes.

	Seizure Detection						Seizure Classification
Model	4-s	6-s	8-s	10-s	12-s	14-s	10-s
LSTM	5	5	5	6	7	7	4
GRU	5	5	5	6	7	8	4
CNN-LSTM	8	8	8	9	9	10	5
ResNet-LSTM	9	9	10	10	12	12	6
ResNet-LSTM-Dilation	9	9	10	10	12	12	6
DCRNN	20	22	23	25	28	30	20
DCRNN w/SS	23	30	35	40	48	60	35
Transformer	12	12	13	14	14	16	8
REST$_{(DS)}$	45	47	50	53	55	60	10
REST$_{(RS)}$	45	47	50	53	55	60	10
REST$_{(RM)}$	70	75	80	90	95	100	25

G. REST Combat Forgetting at Each Time Point

While updating REST specially when the update cell includes multiple updates REST avoids forgetting the input by updating its state based on the affine mapping of the previous state and the input. As an example we consider two following setting:

Setting 1: Updating the state based on previous state only where first the state is initialized as $S_i^t = WX^t + US_i^t$ and then it will iteratively update the state S_i^t as follows:

$$\delta S_i^t = \mathcal{G}_\Theta(S_i^t), \tag{15}$$

$$S_{i+1}^t = S_i^t + \delta S_i^t \odot B. \tag{16}$$

Setting 2: Updating the state based on affine mapping of current input and previous state for iteratively update the state S_i^t as follows:

$$H_i^t = WX^t + US_i^t \tag{17}$$

$$\delta S_i^t = \mathcal{G}_\Theta(H_i^t), \tag{18}$$

$$S_{i+1}^t = H_i^t + \delta S_i^t \odot B. \tag{19}$$

In Setting 1, after mapping from the input to the state space, the state is updated only based on the previous state. This setup poses a risk of the model forgetting information from the current input, especially if the update cell iteratively modifies the state multiple times. This situation hinders state from converging to a stable point and simply diverges due to neglecting the input data. In Setting 2, represented by REST's update cell, the input plays a crucial role and is actively involved in the iterative update process, as shown in equations Equations (17) to (19). This design prevents the model from forgetting information from the current time input X^t, promoting convergence of the state to a more meaningful final state by utilizing the input's information throughout the updates.

As illustrated in Figure 7, Setting 1 fails to converge to a stable point, and the validation loss remains unchanged throughout the training process.

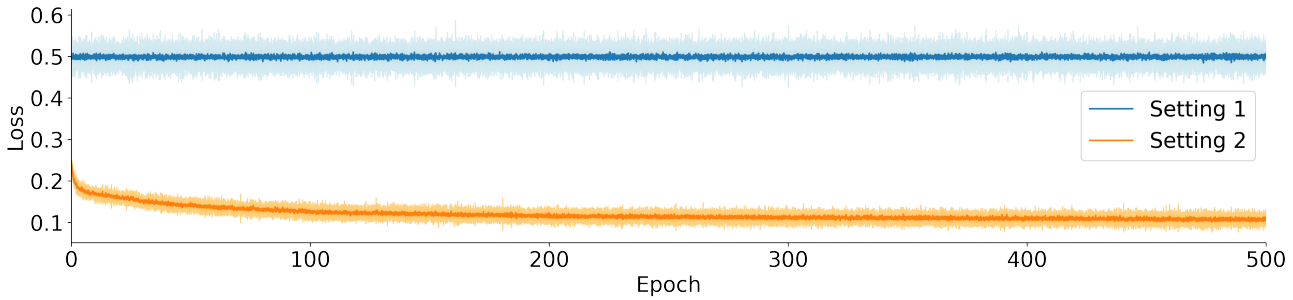

Figure 7. Validation loss of setting 1 and setting 2 mentioned in the Appendix F.

H. Comparison Between Different Gaussian Kernels Threshold for EEG Distance Graph

Here we illustrate different distance graph constructions based on different thresholds or the Gaussian kernel. The lower k values (i.e. 0.6) results in missing connection between nodes and large k thresholds results in connecting nodes which are far away. Similar to Tang et al. (2021) we also choose $k = 0.9$ as threshold which resembles the EEG montage (longitudinal bipolar and transverse bipolar) (Acharya et al., 2016) and results in a reasonable node connection.

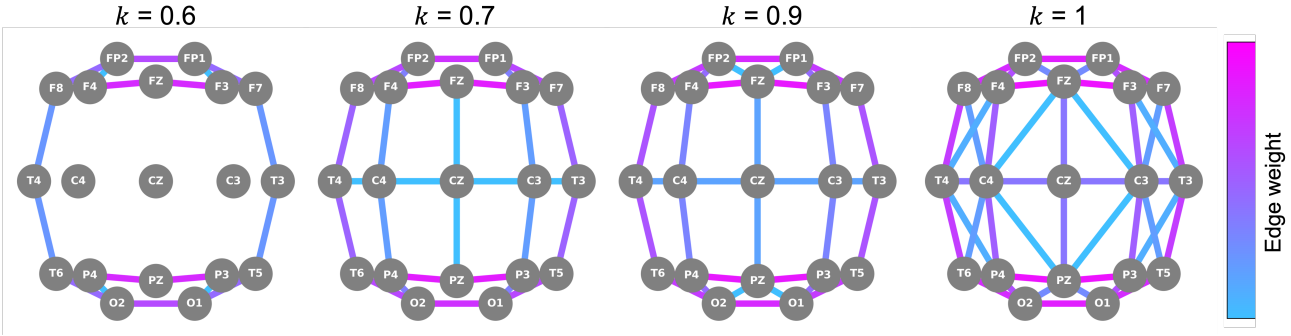

Figure 8. Illustration of distance based EEG graph constructed from different thresholds for the Gaussian kernel. k=0.9 was chosen for this study.

I. Compressing Baseline Models

We tried to compress existing models for seizure detection and classification, achieving performance comparable to those described in Tang et al. (2021). However, in case of LSTM and CNN-LSTM models shrinking the model size without a significant performance drop proved challenging. We matched the performance reported in Tang et al. (2021) for DCRNN and DCRNN w/SS models with only one diffusion convolution gated recurrent unit, and reduced the model size by half, from 2.7 MB to less than 1 MB for DCRNN. Furthermore, for the seizure detection task, we achieved the same accuracy with 126K parameters compared to the original paper's 168,641 parameters.

For the classification task, the original paper (Tang et al., 2021) reported 280,964 parameters for DCRNN and 417,572 parameters for DCRNN w/SS. In our compressed models, we achieved 126K parameters for DCRNN and 330K for DCRNN w/SS, successfully reducing the model size by a factor of 2 for DCRNN and a factor of 1.5 for DCRNN w/SS.

Despite successful reductions, the compressed models still possess a considerable number of parameters, especially in the presence of a gating mechanism, highlighting the non-parameter efficiency and memory demands associated with existing models for seizure detection.

J. F1-Score for Seizure detection

Below is the F1-score (weighted averaged) results for seizure detection task on TUSZ dataset.

Table 10. F1-Score for seizure detection across different time windows for various models.

Model	4-s	6-s	8-s	10-s	12-s	14-s
LSTM	82.3	69.9	79.5	80.5	72.7	73.2
CNN-LSTM	70.1	69.5	75.3	73.5	68.3	67.5
GRU	**82.7**	69.9	81.6	80.5	**81.0**	71.3
RestNet-LSTM	79.7	78.2	80.1	75.1	77.0	76.3
RestNet-Dilation-LSTM	80.5	80.4	79.0	76.6	75.0	74.6
Transformer	78.45	79.3	78.5	**82.0**	79.1	**79.2**
DCRNN	81.2	80.2	81.6	80.0	74.2	72.0
DCRNN W/SS	75.2	**81.1**	81.2	81.0	75.7	76.0
Rest(RS)	69.5	68.4	78.3	79.1	74.7	74.1
Rest(RM)	81.0	75.2	**83.2**	81.0	75.7	76.2

K. Binary Random Masking and Multiple Updates for Other RNNs

We conducted an ablation study to evaluate the performance of RNN baselines with single and multiple random updates, as shown in Table 11.

Table 11. AUROC for seizure detection on a window size of 10s for the TUSZ dataset. Vanilla models are RNN variants without any update techniques, while RM and RS are REST update strategies.

Model	Vanilla	RS	RM
RNN	77.3	80.1	**80.8**
GRU	73.5	72.8	**73.6**
LSTM	70.4	74.5	**74.7**

As shown, the RNN variants can improve their performance in seizure detection tasks using REST update techniques.

L. Size Comparison with 64 Number of Neurons for all Models

Table 12. Size comparison of different models using an equal number of neurons (64). The table indicates the model size and parameter count for REST and baseline models.

Model	Parameters (#)	Size (MB)
DCRNN w/SS	330K	1.319
DCRNN	126K	0.884
Transformer	48.3K	0.193
GRU	402K	1.61
ResNet-LSTM	7.5M	30.3
ResNet-LSTM-Dilation	7.5M	30.3
LSTM	536K	2.147
CNN-LSTM	6M	22.8
REST(DS)	27K	0.051
REST(RS)	27K	0.051
REST(RM)	27K	0.051

M. More Evaluation for Real-Time Detection

We followed the real-time seizure detection framework described by Lee et al. (2022), using a 4-second clip length for seizure detection with a 3-second overlap between consecutive clips. We measured both the inference time and latency, the latter being the delay between the actual onset of a seizure and the model's detection. Low latency is crucial to avoid

late detection of seizure events. As shown in Table 13, REST achieves the lowest latency alongside the Transformer model, while also maintaining significantly lower inference times compared to all other baselines.

Table 13. Comparison of different models' performance on real-time seizure detection with overlapping windows

Model	AUCROC	Latency (s)	Inference (ms)
LSTM	75.5	0.31	3.254
GRU	76.1	0.4	2.12
RestNet-LSTM	79.1	0.3	6.78
RestNet-Dilation-LSTM	80.2	0.34	6.78
CNN-LSTM	81.3	0.26	5.624
DCRNN	79.7	0.25	9.67
Transformer	**83**	**0.2**	2.5
REST(DS)	75.3	0.23	**0.615**
REST(RS)	79.4	**0.2**	0.71
REST(RM)	82.4	0.25	1.29

N. REST W/O Binary Random Mask during Inference

We evaluated REST performance with and without masking over the inference in which similar to Srivastava et al. (2014) strategy the mask was removed and the incremental state update was scaled using the success probability of the binary mask (p).

Table 14. Performance metrics for seizure detection on 10-s clip length with and without inference mask

Model	W/ Inference Mask	W/O Inference Mask
REST (RS)	**81.8**	81.5
REST (RM)	**83.6**	82.9