EENED: End-to-End Neural Epilepsy Detection based on Convolutional Transformer

Chenyu Liu[1] *Xinliang Zhou[1] *Yang Liu[2,1]
[1]School of Computer Science and Engineering, Nanyang Technological University
[2]Zhejiang Sci-Tech University
{chenyu003, xinliang001}@e.ntu.edu.sg, {yangliu}@ntu.edu.sg

***Abstract*—Recently Transformer and Convolution neural network (CNN) based models have shown promising results in EEG signal processing. Transformer models can capture the global dependencies in EEG signals through a self-attention mechanism, while CNN models can capture local features such as sawtooth waves. In this work, we propose an end-to-end neural epilepsy detection model, EENED, that combines CNN and Transformer. Specifically, by introducing the convolution module into the Transformer encoder, EENED can learn the time-dependent relationship of the patient's EEG signal features and notice local EEG abnormal mutations closely related to epilepsy, such as the appearance of spikes and the sprinkling of sharp and slow waves. Our proposed framework combines the ability of Transformer and CNN to capture different scale features of EEG signals and holds promise for improving the accuracy and reliability of epilepsy detection. Our source code will be released soon on GitHub.**

***Index Terms*—Epilepsy detection, Transformer, Convolution neural network, Electroencephalogram**

I. Introduction

Epilepsy is a chronic non-infectious disease caused by the paroxysmal abnormal hypersynchronous electrical activity of brain neurons that affects people of all ages [1], [2]. It is also one of the most common neurological diseases in the world. Due to brain abnormalities, there are differences in the starting position and transmission mode of electrical activity, and the clinical manifestations of epilepsy are characterized by diversification and complexity. Repeated epileptic seizures will cause persistent adverse effects on patients' mental and cognitive functions and even endanger their lives. Studies have shown that EEG signals in epileptic patients differ from non-epileptic subjects. Therefore, judging whether a subject suffers from epilepsy by identifying EEG signals has important clinical significance for diagnosing epilepsy.

Neural network models based on Transformer and CNN are widely used in epilepsy detection tasks via EEG. Transformer performs very well in fields such as natural language processing (NLP), and it can capture long-distance dependencies in temporal signals through self-attention mechanisms. Therefore, Transformer is also widely used in EEG signal processing, modeling EEG signals through the self-attention mechanism, and extracting spatio-temporal relationship features related to epilepsy in the signal. In contrast, CNN-based neural networks pay more attention to capturing local features when processing time-series data, such as signal waveforms of different frequencies of EEG signals. Through hierarchical convolution operations, the convolutional network extracts high-dimensional representations of EEG signals for epilepsy detection and classification. In summary, the neural networks based on Transformer and CNN have different advantages in processing time series signals and can extract features of different scales in epilepsy detection tasks.

Since global and local interactions are all essential for parameter efficiency, we propose an End-to-End neural network based on the convolutional transformer encoder structure, as shown on the left side of Fig 1. Inspired by [3], the encoder blocks of EENED follow the self-attention mechanism of the Transformer model to extract the time-dependent relationship in time-series EEG signals. Unlike the traditional Transformer, we removed the positional encoding [4] in the multi-head attention mechanism and divided the feed-forward layer into two parts to form a sandwich structure. At the same time, a convolution module is introduced in the encoder blocks to enhance the extraction of local features. This convolution module is a hybrid convolution module, including one-dimensional depth-wise convolution and one-dimensional point-wise convolution, which is used to analyze local features in EEG signals that may be related to epilepsy, such as abnormal waveforms and sudden changes in frequency and amplitude.

As proved by the experimental results of the Epileptic Seizure Recognition dataset [5], EENED shows higher accuracy than CNN and Transformer-based neural networks. While capturing the long-term dependence of temporal EEG signals and extracting local signal features through convolution modules, EENED has shown reliability and great potential in epilepsy detection tasks.

II. Methodology

A. Encoder Blocks

Inspired by Macaron-Net [6], the encoder blocks of EENED adopt a sandwich structure, which divides the feedforward layer in the Transformer encoder into two half-step residual feedforward units. A multi-head self-attention and convolution modules are included between the two feedforward modules,

*Corresponding Author: Yang Liu and Xinliang Zhou.

Chenyu Liu, and Xinliang Zhou are with the School of Computer Science and Engineering, Nanyang Technological University, 639798, Singapore {chenyu003, xinliang001}@e.ntu.edu.sg

[2,1]Yang Liu is with Zhejiang Sci-Tech University, Hangzhou, 314423, China and the School of Computer Science and Engineering, 50 Nanyang Avenue, 639798, Singapore yangliu@ntu.edu.sg

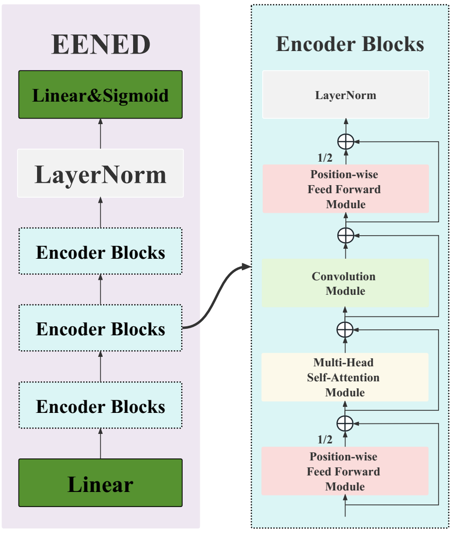

Fig. 1. **End-to-End Neural Epilepsy Detection model architecture.** EENED's structure contains several encoder blocks and linear layers. The encoder's multi-headed self-attention and convolution modules are sandwiched between two macaron-like positional-wise feed-forward layers with half-step residual connection

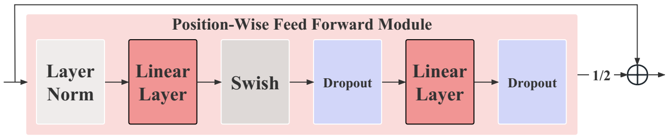

Fig. 2. **Position-wise feed-forward model.** Two linear layers increase and decrease the dimensionality of the data, respectively.

as shown in Fig 1. Mathematically, for given input $(f^t_{(e-1)}|t = 1, ..., T)$ to the e^{th} encoder block, the output $(f^t_{(e)}|t = 1, ..., T)$ of the block is:

$$f^{(e)}_{FF_1} = (f^t_{(e-1)}|t = 1, ..., T) + \frac{1}{2}\,\mathrm{PWFF}((f^t_{(e-1)}|t = 1, ..., T)) \quad (1)$$

$$f^{(e)}_{MHSA} = f^{(e)}_{PWFF_1} + \mathrm{MHSA}(f^{(e)}_{PWFF_1}) \quad (2)$$

$$f^{(e)}_{Conv} = f^{(e)}_{MHSA} + \mathrm{Conv}(f^{(e)}_{MHSA}) \quad (3)$$

$$(f^t_{(e)}|t = 1, ..., T) = \mathrm{LayerNorm}(f^{(e)}_{Conv} + \frac{1}{2}\,\mathrm{PWFF}(f^{(e)}_{Conv})) \quad (4)$$

Where $\mathrm{PWFF}()$ refers to the position-wise feed-forward module, $\mathrm{MHSA}()$ refers to the multi-head self-attention module, and $\mathrm{Conv}()$ refers to the convolution module.

1) Position-wise feed-forward module(PWFF): The Transformer architecture [7] employs a feed-forward module before and after the MHSA module, consisting of two linear layers and activation as shown in Fig 2. The e^{th} PWFF module transforms a sequence of input vectors $(f^t_{(e-1)}|t = 1, ..., T)$ from the previous encoder as follows:

$$F^{(e)} = \mathrm{LayerNorm}([f^{(e-1)}_t \cdots f^{(e-1)}_T]) \,\epsilon\, \mathbb{R}^{T\times D} \quad (5)$$

$$\bar{F}^{(e)} = \mathrm{Swish}(F^{(e)}W^{(e)}_1 + 1{b^{(e)}_1}^\top)W^{(e)}_2 + 1{b^{(e)}_2}^\top \,\epsilon\, \mathbb{R}^{T\times D} \quad (6)$$

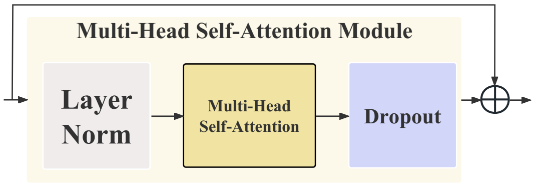

Fig. 3. **Multi-head self-attention module.** We use a multi-head self-attention similar to the transformer encoder but remove the relative positional embedding in this pre-norm residual unit.

Where T is the length of time and D is the feature dimension. $W^{(e)}_1 \,\epsilon\, \mathbb{R}^{D\times d_{pwff}}$ and $b^{(e)}_1 \,\epsilon\, \mathbb{R}^{d_{pwff}}$ are the projection matrix and bias of the first linear layer, $1 \,\epsilon\, \mathbb{R}^T$ is an all-one vector. d_{pwff} is the dimension of hidden units. We also apply Swish activation [8] $\mathrm{Swish}(\cdot)$ and dropout [9] to help regularize the module. $W^{(e)}_2 \,\epsilon\, \mathbb{R}^{D\times d_{pwff}}$ and $b^{(e)}_2 \,\epsilon\, \mathbb{R}^{d_{pwff}}$ are the second linear projection matrix and bias. Following the pre-norm residual units [10], the final output of the PWFF module is computed as follows:

$$F^{(e)}_{PWFF} = F^{(e)} + \frac{\mathrm{Dropout}(\bar{F}^{(e)})}{2} \,\epsilon\, \mathbb{R}^{T\times D} \quad (7)$$

2) Multi-head self-attention module(MHSA): The structure of the multi-head self-attention module is shown in Fig 3. The MHSA module in the e^{th} encoder processes features from the PWFF module. The input features $F^{(e)}_{PWFF}$ is converted by layer normalization:

$$\bar{F}^{(e)} = \mathrm{LayerNorm}(F^{(e)}_{PWFF}) \,\epsilon\, \mathbb{R}^{T\times D} \quad (8)$$

Then, in the multi-head self-attention block, each attention head computes a pairwise similarity matrix $S^{(e)}_h$ using the dot products of query vectors $\bar{F}^{(e)}Q^{(e)}_h \,\epsilon\, \mathbb{R}^{T\times d}$ and key vectors $\bar{F}^{(e)}K^{(e)}_h \,\epsilon\, \mathbb{R}^{T\times d}$

$$S^{(e)}_h = \bar{F}^{(e)}Q^{(e)}_h(\bar{F}^{(e)}K^{(e)}_h)^\top \,\epsilon\, \mathbb{R}^{T\times T}(1 \leq h \leq H) \quad (9)$$

where H is the number of heads. $Q^{(e)}_h, K^{(e)}_h \epsilon \mathbb{R}^{D\times d}$ are query and key projection matrices for the h^{th} head. The pairwise similarity matrix $S^{(e)}_h$ is scaled by $1/\sqrt{D/H}$ and a softmax function is applied to form the attention weight matrix $A^{(e)}_h$:

$$A^{(e)}_h = \mathrm{Softmax}\left(\frac{S^{(e)}_h}{\sqrt{D/H}}\right) \,\epsilon\, \mathbb{R}^{T\times T} \quad (10)$$

Then the attention weight matrix $A^{(e)}_h$ is used to compute context vectors $C^{(e)}_h$ with the value vectors $\bar{F}^{(e)}V^{(e)}_h \,\epsilon\, \mathbb{R}^{T\times d}$:

$$C^{(e)}_h = A^{(e)}_h(\bar{F}^{(e)}V^{(e)}_h) \,\epsilon\, \mathbb{R}^{T\times d} \quad (11)$$

where $V^{(e)}_h \,\epsilon\, \mathbb{R}^{D\times d}$ is the value projection matrix. The final output feature of the multi-head self-attention module is computed by the concatenation of all heads' context vectors and an output projection matrix $O^{(e)} \epsilon \mathbb{R}^{D \times D}$:

$$\bar{F}^{(e)}_{MHSA} = [C^{(e)}_1 \cdots C^{(e)}_H] O^{(e)} \epsilon \mathbb{R}^{T \times D} \tag{12}$$

$$F^{(e)}_{MHSA} = \text{LayerNorm}(\bar{F}^{(e)} + \text{DropOut}(\bar{F}^{(e)}_{MHSA})) \epsilon \mathbb{R}^{T \times D} \tag{13}$$

3) Convolution module: Similar to [11], the convolution module, as shown in Fig 4, takes $F^{(e)}_{MHSA}$ as input and starts with a point-wise convolution and a gated linear unit (GLU) [12]:

$$\bar{F}^{(e)} = \text{PWConv}(\text{LayerNorm}(F^{(e)}_{MHSA})) \epsilon \mathbb{R}^{T \times 2D} \tag{14}$$

$$\bar{F}^{(e)}_{glu} = (\hat{F}^{(e)} W^{(e)}_1 + b^{(e)}_1) \bigotimes \sigma(\check{F}^{(e)} W^{(e)}_2 + b^{(e)}_2) \epsilon \mathbb{R}^{T \times D} \tag{15}$$

where $\text{PWConv}(\cdot)$ is a one-dimensional point-wise convolutional layer with kernel size (1 X 1) and stride of 1. The output $\bar{F}^{(e)} \epsilon \mathbb{R}^{T \times 2D}$ could be divided into $\hat{F}^{(e)} \epsilon \mathbb{R}^{T \times D}$ and $\check{F}^{(e)} \epsilon \mathbb{R}^{T \times D}$, which are the first half and the second half of the output, respectively. $W^{(e)}_1 \epsilon \mathbb{R}^{D \times D}$, $b^{(e)}_1 \epsilon \mathbb{R}^{D}$,$W^{(e)}_2 \epsilon \mathbb{R}^{D \times D}$, $b^{(e)}_1 \epsilon \mathbb{R}^{D}$ are learned parameters, σ is the sigmoid function and $\bigotimes$ is the element-wise product. The output of GLU $\bar{F}^{(e)}_{glu}$ is processed with a $\text{DWConv}(\cdot)$:

$$\bar{F}^{(e)}_{DWConv} = W^{(e)}_{conv}\, \text{Swish}(\text{DWConv}(\bar{F}^{(e)}_{glu})) + b^{(e)}_{Conv} \epsilon \mathbb{R}^{T \times D} \tag{16}$$

where $W^{(e)}_{Conv} \epsilon \mathbb{R}^{d_{pwconv} \times D}$, $b^{(e)}_{Conv} \epsilon \mathbb{R}^{D}$ are learned linear parameters of Convolution module. $\text{Swish}(\cdot)$ is the activation function. $\text{DWConv}(\cdot)$ is a one-dimensional depth-wise convolutional layer with a kernel size of 15. d_{pwconv} is the dimension of depth-wise convolution layer output. The final output features $F^{(e)}_{Conv}$ of the convolution module are as follows:

$$F^{(e)}_{Conv} = F^{(e)}_{MHSA} + \text{Dropout}(\bar{F}^{(e)}_{DWConv}) \epsilon \mathbb{R}^{T \times D} \tag{17}$$

III. Experiments and Results

A. Dataset

The Epileptic Seizure Recognition dataset [5] contains EEG recordings from 500 subjects. The brain activity of each subject was recorded for 23.6 seconds. Since four of the five categories are unrelated to epilepsy, we reduced the labels to one category. The training set contains 7360 segments of EEG signal data, and the test set contains 1840 segments of EEG signal data, of which 1461 are non-epileptic EEG signals. We followed the data processing in [13].

B. Model configuration

1) EENED: EENED contains three Encoder blocks, and each Encoder layer contains a self-attention module and a convolution module. The attention mechanism consists of 8 attention heads, each with a dimension of 64. The convolution module uses one-dimensional convolution layers and GLU activation function; the convolution kernel size is 15, the step size is 1, the padding is 7, and the number of groups is 512. The model also contains two feed-forward layers and residual connections, and finally, it applies two linear layers and a sigmoid activation function to classify the features.

2) Dense-CNN: Dense-CNN [14] is comprised of multiple convolutional layers, with each layer having a different set of convolutional filters. The first is a linear layer that outputs a tensor of size 512. The subsequent layers use an architecture called Dense-Inception which is made up of several Inception modules. Each Inception module consists of multiple branches that process the input in parallel using different convolutional filters of different kernel sizes. The output of each branch is then concatenated along the channel dimension and fed into a 1x1 convolutional layer to reduce the number of channels. The output of one Inception module is then fed into the next Inception module, and the process is repeated until the final layer, which outputs a tensor of size 18.

3) Transformer: This network architecture is a Transformer-based classifier consisting of an upsample layer, two linear layers, Transformer-Encoder layers with 3 Transformer encoders, a fully connected layer, and a sigmoid activation function. The upsample layer maps the input sequence to a hidden state of size 512. The first linear layer maps the hidden state to a single value, which is then used to scale the output of the Transformer-Encoder layer. The Transformer-Encoder layer contains a self-attention mechanism and two linear and dropout layers. The self-attention mechanism uses 8 attention heads and an output projection matrix of size 512. The dropout probability is set to 0.1 for the attention and linear layers. The fully connected layer maps the output of the Transformer-Encoder to a single value, which is then passed through the sigmoid activation function to produce the final classification output.

4) CNN-LSTM: CNN-LSTM [15] comprises a convolutional neural network and a long and short-term memory (LSTM) layer. The convolutional neural network consists of two convolutional layers and a maximum pooling layer, and the output size of the fully connected layer is 512. The input of the LSTM layer is the output of the fully connected layer and contains two LSTM layers with a hidden state size of 128.

C. Result and Analysis

As shown in Fig 5, from the prediction result confusion matrix of the four models, the accuracy rates of Dense-CNN and Transformer are 0.942 and 0.943, respectively. The accuracy rates of CNN-LSTM and EENED are 0.958 and 0.982, respectively. Among them, EENED achieved the highest accuracy rate. When Dense-CNN is used to process time-domain signals, it cannot perform better due to the lack

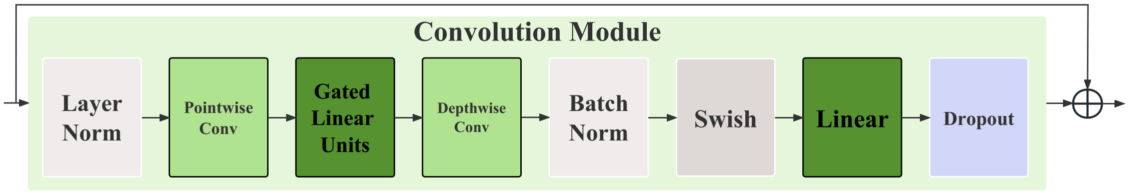

Fig. 4. **Convolution module.**The convolution module contains two convolutional layers of different scales, with Gated linear units used as the activation layer in the middle. The Swish activation function is used, followed by a linear layer.

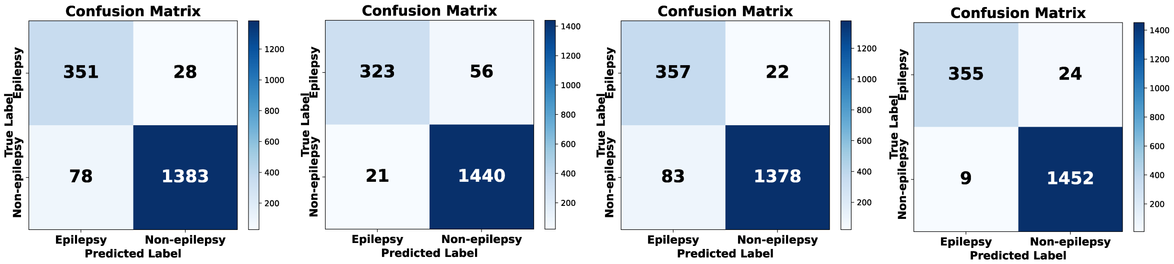

Fig. 5. Confusion matrix of the predicted results of the four models. From left to right: Dense-CNN, CNN-LSTM, Transformer, and EENED.

TABLE I
COMPARISON OF PERFORMANCE (F1 SCORE AND ACCURACY) OF FOUR NEURAL NETWORKS ON THE EPILEPTIC SEIZURE RECOGNITION DATASET.

Network	Accuracy	F1 Score
Dense-CNN [14]	0.942	0.963
Transformer	0.943	0.963
CNN-LSTM [15]	0.958	0.974
EENED	**0.982**	**0.989**

of a mechanism for processing sequence data. Transformer is good at global modeling of time-domain data. However, it is challenging to capture short sequence signal features related to epilepsy in EEG signals, such as the appearance of spikes and the sprinkling of sharp and slow waves. In contrast, CNN-LSTM integrates local and global feature extraction to a certain extent, which can effectively capture the temporal dependence of time series data. EENED has the highest accuracy rate in the experimental results. It combines the characteristics of CNN and Transformer. While learning the global temporal dependence of epilepsy-related EEG signals, it captures local signal features through the convolution module.

IV. CONCLUSION

EENED achieves higher accuracy in epilepsy detection tasks than other transformer and CNN-based neural networks. The experimental results show that EENED can combine the ability of the self-attention mechanism to build the long-term dependence of EEG signals and the characteristics of the convolution module to extract local EEG signal features, which is crucial for using EEG to detect epilepsy as a reference for medical diagnosis.

V. ACKNOWLEDGMENTS

This research is supported by National Research Foundation (NRF) Singapore, NRF Investigatorship NRF-NRFI06-2020-0001.

REFERENCES

[1] Alexander J Casson, David C Yates, Shelagh JM Smith, John S Duncan, and Esther Rodriguez-Villegas, "Wearable electroencephalography," *IEEE engineering in medicine and biology magazine*, vol. 29, no. 3, pp. 44–56, 2010.

[2] Xinliang Zhou, Chenyu Liu, Liming Zhai, Ziyu Jia, Cuntai Guan, and Yang Liu, "Interpretable and robust ai in eeg systems: A survey," *arXiv preprint arXiv:2304.10755*, 2023.

[3] Anmol Gulati, James Qin, Chung-Cheng Chiu, Niki Parmar, Yu Zhang, Jiahui Yu, Wei Han, Shibo Wang, Zhengdong Zhang, Yonghui Wu, et al., "Conformer: Convolution-augmented transformer for speech recognition," *arXiv preprint arXiv:2005.08100*, 2020.

[4] Yusuke Fujita, Naoyuki Kanda, Shota Horiguchi, Yawen Xue, Kenji Nagamatsu, and Shinji Watanabe, "End-to-end neural speaker diarization with self-attention," in *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*. IEEE, 2019, pp. 296–303.

[5] Ralph G Andrzejak, Klaus Lehnertz, Florian Mormann, Christoph Rieke, Peter David, and Christian E Elger, "Indications of nonlinear deterministic and finite-dimensional structures in time series of brain electrical activity: Dependence on recording region and brain state," *Physical Review E*, vol. 64, no. 6, pp. 061907, 2001.

[6] Yiping Lu, Zhuohan Li, Di He, Zhiqing Sun, Bin Dong, Tao Qin, Liwei Wang, and Tie-Yan Liu, "Understanding and improving transformer from a multi-particle dynamic system point of view," *arXiv preprint arXiv:1906.02762*, 2019.

[7] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[8] Prajit Ramachandran, Barret Zoph, and Quoc V Le, "Searching for activation functions," *arXiv preprint arXiv:1710.05941*, 2017.

[9] Nitish Srivastava, Geoffrey Hinton, Alex Krizhevsky, Ilya Sutskever, and Ruslan Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[10] Qiang Wang, Bei Li, Tong Xiao, Jingbo Zhu, Changliang Li, Derek F Wong, and Lidia S Chao, "Learning deep transformer models for machine translation," *arXiv preprint arXiv:1906.01787*, 2019.

[11] Zhanghao Wu, Zhijian Liu, Ji Lin, Yujun Lin, and Song Han, "Lite transformer with long-short range attention," *arXiv preprint arXiv:2004.11886*, 2020.

[12] Yann N Dauphin, Angela Fan, Michael Auli, and David Grangier, "Language modeling with gated convolutional networks," in *International conference on machine learning*. PMLR, 2017, pp. 933–941.

[13] Emadeldeen Eldele, Mohamed Ragab, Zhenghua Chen, Min Wu, Chee Keong Kwoh, Xiaoli Li, and Cuntai Guan, "Time-series representation learning via temporal and contextual contrasting," *arXiv preprint arXiv:2106.14112*, 2021.

[14] Khaled Saab, Jared Dunnmon, Christopher Ré, Daniel Rubin, and Christopher Lee-Messer, "Weak supervision as an efficient approach for automated seizure detection in electroencephalography," *NPJ digital medicine*, vol. 3, no. 1, pp. 59, 2020.

[15] David Ahmedt-Aristizabal, Tharindu Fernando, Simon Denman, Lars Petersson, Matthew J Aburn, and Clinton Fookes, "Neural memory networks for seizure type classification," in *2020 42nd Annual International Conference of the IEEE Engineering in Medicine & Biology Society (EMBC)*. IEEE, 2020, pp. 569–575.