CS224W Report - Multiclass Seizure Classification from EEG with Graph Convolutional Recurrent Neural Networks

Siyi Tang
Department of Electrical Engineering
siyitang@stanford.edu

Kaylee Zhang
Department of Statistics
kayleez@stanford.edu

Louise Huang
Department of Statistics
qyhuang@stanford.edu

Abstract

Traditionally, seizure classification from electroencephalogram (EEG) is performed by clinical experts' visual analyses of EEG signals. Recent advances in deep learning have enabled the development of automated seizure classification frameworks. Most of previous efforts on modeling EEG with deep learning utilized convolutional neural networks (CNNs) and/or recurrent neural networks (RNNs), ignoring the inherent graphical structure in EEG. In this paper, we propose to model the spatiotemporal dependencies in EEG data with a graph convolutional recurrent neural network. Specifically, the spatial dependency is characterized by graph convolutions on the EEG electrode graph, and the temporal dependency is captured by a recurrent neural network. We explore various graph structures, and show that our graph-based approach improves multiclass seizure classification compared to RNN or CNN-based methods.

1 Introduction

Epileptic seizure is a transient occurrence of signs and symptoms due to abnormal excessive or synchronous neuronal activities in the brain. Around 50 million people worldwide have epilepsy, making it one of the most common neurological diseases globally. Due to its unpredictable nature, patients with epilepsy and their family often suffer from low quality of life, stigma, and discrimination. However, it is estimated that up to 70% of people living with epilepsy could live seizure-free if properly diagnosed and treated [1].

The traditional method for epilepsy diagnosis and treatment is based on a clinical expert's analysis of electroencephalogram (EEG) signals. However, seizure detection from time-series EEG signals is a challenging task that requires an experienced clinician's knowledge and substantial amount of time. Classifying seizures into finer classes is even more challenging. In fact, it could take an experienced neurologist several hours to evaluate a single patient's EEG record, not to mention the fact that EEG data is often collected over hours to days [2]. As such, algorithms to automatically classify seizure types would be beneficial for overcoming the bottleneck in diagnosis of epilepsy and thus accelerating the clinical workflows. Automatic seizure type classification may also allow remote monitoring of patients, as well as timely diagnosis and treatment to improve patient care [3]. Although much effort has gone into developing seizure detection algorithms [4–6, 2], little has been done for seizure classification [7, 8].

Feature extraction from EEG signals is the most crucial step in building an automatic seizure classification system. Downstream classification task will benefit if the model can successfully extract features that capture discriminative characteristics of the neural patterns of different types of seizures. Previous works have used recurrent neural networks (RNNs) and/or convolutional neural networks (CNNs) to model EEG data [8, 9]. However, such methods either ignore spatial dependencies or assumes regular grid structures in EEG signals. In contrast, EEG signals are recorded

from multiple electrodes placed on a patient's scalp, and thus have an inherent graph structure. Hence, we hypothesize that a graph convolutional recurrent neural network would be better at capturing the spatiotemporal dependencies in EEG signals than CNNs or RNNs.

In this study, we aim to develop a deep learning framework for multiclass seizure classification. We focus on the following seven common seizure classes: focal non-specific seizure (FN), generalized non-specific seizure (GN), simple partial seizure (SP), complex partial seizure (CP), absence seizure (AB), tonic seizure (TN) and tonic-clonic seizure (TC). We use graph convolutions [10, 11] to leverage the graph structure in EEG, and a recurrent neural network to capture the temporal dynamics. In the remainder of this paper, we use EEG electrodes, EEG channels and nodes interchangeably.

Our main contributions are as follows:

- We propose to model the EEG signals with a graph convolutional recurrent neural network, capturing the spatial and temporal dependencies in EEG.
- We investigate various graph structures, leveraging the physical structures in EEG electrodes and correlations in EEG signals.
- We show that our graph-based models out-perform RNN or CNN-based models on the multiclass seizure classification task, demonstrating the benefits of modeling EEG data with graphs.

2 Related Works

Convolutional neural networks have been generalized from regular grids to irregular structures. Defferrard et al. [11] presented mathematical foundations of extending CNNs to graphs in the context of spectral graph theory. They proposed to parameterize the spectral filters by Chebyshev polynomials of the graph Laplacian, which ensures that the filters are strictly localized within K hops from the central node. Furthermore, the proposed parameterization allows recursive computation of the spectral filters, and thus reduces the computational complexity from $\mathcal{O}(n^2)$ to linear time complexity.

To model spatiotemporal dependencies in time-series data, Li et al. [10] proposed diffusion convolutional recurrent neural network (DCRNN) in the context of traffic forecasting. The authors modeled traffic flows as a diffusion process, characterized by a bidirectional random walk on a *weighted directed* graph. In addition, they employed a sequence-to-sequence architecture with scheduled sampling to model the temporal dynamics. As proven in the paper, diffusion convolution is closely related to spectral graph convolution in [11] for undirected graphs. In comparison, Yu et al. [12] utilized complete convolutions on graphs for predicting traffic flows in real-time.

Previous works on modeling EEG signals using deep learning have employed recurrent neural networks (RNNs) [9], convolutional neural networks (CNNs) [8] and graph convolutional neural networks [2, 13]. For example, SeizureNet [8] is an ensemble of three CNN-based classifiers on multi-spectral features extracted from raw EEG signals. In order to apply CNN, the multi-spectral features were resized to a dimension of $224 \times 224 \times 3$, which might cause loss of information in the features. In [13], the authors built a graph based on frequency band power features of EEG signals, where each node corresponds to a frequency band. This is different from our proposed method because we directly model EEG signals as a graph. In addition, [13] experimented on the following three ways to compute edge weights in the EEG graph: correlation-based, distance-based, and random. Note that the distance-based method is similar to our approach discussed in section 4.2.

A substantial amount of research has been conducted on developing automatic seizure detection (i.e. binary classification of seizure vs non-seizure) systems from EEG [4–6, 2]. [4] and [5] built classifiers based on handcrafted EEG features. More recently, [6, 2] utilized deep learning methods to automatically learn feature representations in EEG signals. In particular, Covert et al. [2] proposed temporal graph convolutional network (TGCN), which leverages spatial information in time-series data. TGCN consists of feature extractors that are localized and shared over the temporal and spatial dimensions of the data. Thus, TGCN is inherently invariant to when and where the patterns occur, and it also has the advantage of reducing the number of parameters and preventing over-fitting.

Multiclass seizure classification is much more challenging than binary seizure detection because (a) some seizure types are less common than others, (b) the inter-patient variability is generally higher than that in the binary seizure detection task. To the best of our knowledge, [7] and [8] are the only

works on multiclass seizure classification using deep learning techniques. However, both studies built models upon handcrafted EEG features, whereas we work on EEG signals directly.

3 Data

3.1 Dataset

We use the public Temple University Hospital EEG Corpus version v1.4.0 [14, 15]. The dataset contains a training set and a test set with non-overlapping patients. Table 1 shows the number of seizures in each class, as well as the number of patients having each seizure type in the entire dataset. Note that one patient might be associated with multiple seizure types. As can be seen in Table 1, the seven seizure classes are highly imbalanced, with FN seizure being the largest class, and SP seizure being the smallest class. Figure 1 shows a EEG segment with FN seizure in three EEG channels.

Seizure Type	#Seizures	#Patients
Focal non-specific seizure (FN)	992	109
Generalized non-specific seizure (GN)	415	44
Simple partial seizure (SP)	44	2
Complex partial seizure (CP)	342	34
Absence seizure (AB)	99	12
Tonic seizure (TN)	67	2
Tonic-clonic seizure (TC)	50	11

Table 1: Number of seizures and number of patients in each class

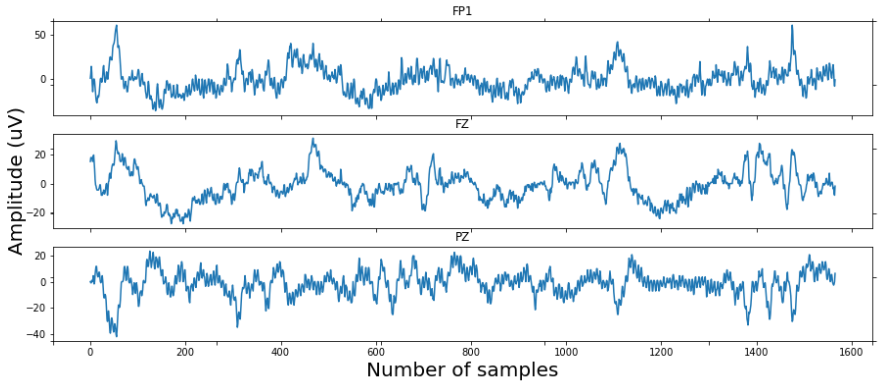

Figure 1: An example of raw EEG signals with FN seizure. Only three EEG channels are shown here (i.e. FP1, FZ and PZ). The seizure event spanned approximately 4 seconds. The x-axis shows the number of samples (sampling frequency = 400 Hz). The y-axis shows the magnitude of EEG signals (uV).

3.2 Data splits

As the official training set in the Temple dataset is relatively small (i.e. only 1327 seizures in the training set), we created our own train/dev/test splits by combining the official training and test sets and randomly splitting with ratio 80%/10%/10% in a cross-patient manner. In other words, one patient only belongs to one of the three splits. Note that only two patients have SP and TN seizures respectively (Table 1), and thus we assign one patient to either train or test splits. Therefore, the dev split does not contain SP or TN seizures.

4 Methods

4.1 Preprocessing

In this study, we use signals measured from the following 19 common EEG channels: FP1, FP2, F3, F4, C3, C4, P3, P4, O1, O2, F7, F8, T3, T4, T5, T6, FZ, CZ and PZ. We perform the following preprocessing steps to convert raw EEG signals into EEG sequences. For each EEG signal segment containing a seizure event, (a) slide a t-second time window over the signals to obtain "clips" of t-second; (b) resample each "clip" to the same frequency f (e.g. $f = 200$ Hz; because the EEG data have different sampling frequencies), resulting in an input dimension of $P = t \times f$; (c) concatenate all the "clips" into a EEG sequence; (d) make all the sequences the same length T by padding with zeros (for sequences whose lengths $< T$) or truncating (for sequences whose lengths $> T$); (e) z-normalization with respect to the train split. Hence, we can now denote a preprocessed EEG sequence as $\mathbf{X} \in \mathbb{R}^{T \times N \times P}$, where N is the number of EEG channels and P is the input dimension.

4.2 Graph construction

We represent the EEG electrodes as a graph $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$, where $\mathcal{V}$ denotes the set of nodes (i.e. electrodes), $\mathcal{E}$ denotes the set of edges, and $\mathbf{W}$ is the adjacency matrix. We explore three ways of constructing the graph, which are explained in details in next sections. Figure 2 shows the resulting electrode graphs from the three methods. For better visualization, we do not show self-edges.

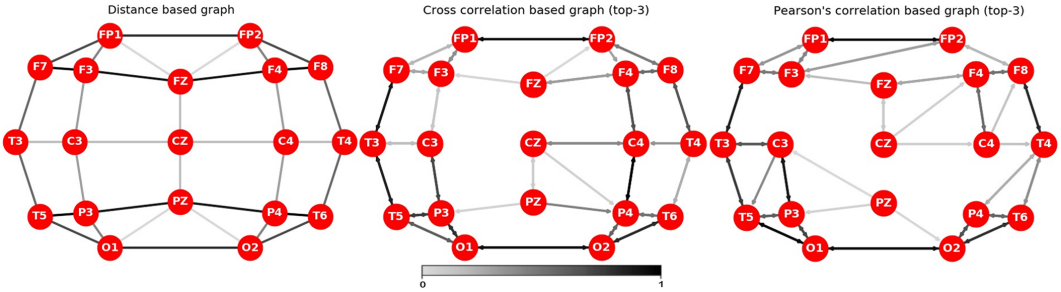

Figure 2: Graphs constructed using three different methods. From left to right: distance-based graph (threshold $k_{dist} = 0.05$), cross-correlation-based top-3 graph and Pearson correlation-based top-3 graph. The distance-based graph is undirected, while cross-correlation and Pearson correlation-based top-3 graphs are directed. Darker edge colors correspond to larger edge weights. For better visualization, we do not show self-edges here.

4.2.1 Distance-based graph

First, we construct the graph based on the physical distances between EEG electrode pairs. The edge weights are computed as follows using a thresholded Gaussian kernel [16]:

$$W_{ij} = \begin{cases} exp\left(- \frac{dist(v_i, v_j)^2}{\sigma^2} \right), & \text{if } W_{ij} \geq k_{dist} \\ 0, & \text{otherwise} \end{cases} \tag{1}$$

where $dist(v_i, v_j)$ is the physical distance in 3D space between electrodes v_i and v_j according to the standard 10-5 system [17], σ is the standard deviation of the distances, and k_{dist} is the threshold.

4.2.2 Cross-correlation-based (CC-based) graph

Secondly, we construct the graph based on the pairwise cross-correlations between signals in each EEG channel pair. Specifically, for each raw EEG signal, the edge weight W_{ij} of the associated graph is computed as the absolute value of the normalized cross-correlation between signals in channels v_i and v_j. This results in one graph for each input. Next, we combine the individual graphs by weighted averaging, giving higher weights to minority seizure classes. To introduce sparsity in the combined graph, we propose the following two approaches: (a) Keep the edges of top-k neighbors for each

node, which results in a *directed* graph. For instance, there is a directed edge from v_i to v_j if v_j is among the top-k neighbors of v_i. (b) Keep the edges whose weights are above a certain threshold k_{cc}, which results in an *undirected* graph.

4.2.3 Pearson correlation-based (PC-based) graph

Lastly, we construct the graph based on the pairwise Pearson correlations between features in each EEG channel pair. The features are computed as the average band powers of EEG signals in 5 main frequency bands: delta (1-4 Hz), theta (4-8 Hz), alpha (8-12 Hz), beta (12-30 Hz) and gamma (30-55 Hz). Similar to the CC-based graph, this approach also results in one graph for each input. We then employ the same methods described in the previous section to combine individual graphs.

4.3 Model

Figure 3 shows the overall architecture of our model.

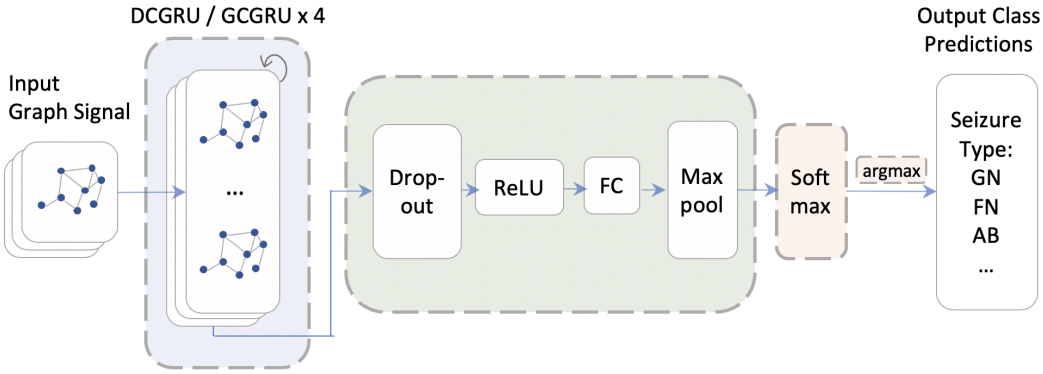

Figure 3: Model architecture includes the input layer, four diffusion convolutional GRU (DCGRU) or ChebNet graph convolutional GRU (GCGRU) layers, a dropout layer followed by ReLU non-linear activation, and a fully connected (FC) layer followed by max-pooling. Softmax function outputs the predicted probability of each seizure class.

Let $\mathcal{G} = \{\mathcal{V}, \mathcal{E}, \mathbf{W}\}$ be the graph constructed using any of the the methods described in section 4.2. For *directed* graphs (i.e. CC-based and PC-based top-k graphs), we employ the *diffusion convolution* proposed in [10] to model the *spatial* dependencies. More specifically, we model EEG signals as a diffusion process, characterized by a bidirectional random walk on the graph $\mathcal{G}$. The resulting diffusion convolution operation is as follows:

$$\mathbf{X}_{:,p*\mathcal{G}} f_\theta = \sum_{k=0}^{K-1} \left(\theta_{k,1}(\mathbf{D}_O^{-1}\mathbf{W})^k + \theta_{k,2}(\mathbf{D}_I^{-1}\mathbf{W}^\mathbf{T})^k\right)\mathbf{X}_{:,p} \text{ for } p \in \{1, ..., P\} \tag{2}$$

where $\mathbf{X} \in \mathbb{R}^{N \times P}$ is the graph signal with N nodes and P input features, f_θ is the convolution filter with parameters $\theta \in \mathbb{R}^{K \times 2}$, $\mathbf{D}_O$ and $\mathbf{D}_I$ are the out-degree and in-degree diagonal matrices respectively, $\mathbf{D}_O^{-1}\mathbf{W}$ and $\mathbf{D}_I^{-1}\mathbf{W}^\mathbf{T}$ are the state transition matrices of the outward and inward diffusion processes respectively, and K is the number of maximum diffusion steps.

Whereas for *undirected* graphs (i.e. distance-based, CC-based and PC-based thresholded graphs), we model the *spatial* dependencies with *ChebNet graph convolution* [11, 10]. Let $\mathbf{L} = \mathbf{D}^{-\frac{1}{2}}(\mathbf{D} - \mathbf{W})\mathbf{D}^{-\frac{1}{2}} = \mathbf{\Phi}\mathbf{\Lambda}\mathbf{\Phi}^T$ be the normalized graph Laplacian. The convolutional filter f_θ is parameterized to be a K order polynomial of $\mathbf{\Lambda}$, and is computed using stable Chebyshev polynomial basis:

$$\mathbf{X}_{:,p*\mathcal{G}} f_\theta = \mathbf{\Phi}\left(\sum_{k=0}^{K-1} \theta_k \mathbf{\Lambda}^k\right)\mathbf{\Phi}^T\mathbf{X}_{:,p} = \sum_{k=0}^{K-1} \theta_k \mathbf{L}^k \mathbf{X}_{:,p} = \sum_{k=0}^{K-1} \tilde{\theta_k} T_k(\tilde{\mathbf{L}})\mathbf{X}_{:,p} \tag{3}$$

where $T_0(x) = 1, T_1(x) = x, T_k(x) = xT_{k-1}(x) - T_{k-2}(x)$ are the bases of the Chebyshev polynomial. $\tilde{\mathbf{L}} = \frac{2}{\lambda_{max}}\mathbf{L} - \mathbf{I}$ is the scaled graph Laplacian so that the eigenvalues are mapped from $[0, \lambda_{max}]$ to $[-1, 1]$. As shown in [10], the output of ChebNet graph convolution is similar to the output of diffusion convolution, up to a constant scaling factor for undirected graphs.

Next, similar to DCRNN [10], we use Gated Recurrent Unit (GRU), a variant of RNN, to model the *temporal* dependencies. The matrix multiplications in GRU are replaced with diffusion convolutions (for directed graphs; Equation 2) or ChebNet graph convolutions (for undirected graphs; Equation 3) , resulting in the following variant of GRU. For simplicity, we denote GRU with diffusion convolution as DCGRU, and GRU with ChebNet graph convolution as GCGRU.

$$\mathbf{r}^{(t)} = \sigma\big(\mathbf{\Theta}_{r*\mathcal{G}}[\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}] + \mathbf{b}_r\big) \qquad \mathbf{u}^{(t)} = \sigma\big(\mathbf{\Theta}_{u*\mathcal{G}}[\mathbf{X}^{(t)}, \mathbf{H}^{(t-1)}] + \mathbf{b}_u\big) \tag{4}$$

$$\mathbf{C}^{(t)} = tanh\big(\mathbf{\Theta}_{C*\mathcal{G}}[\mathbf{X}^{(t)}, (\mathbf{r}^{(t)} \odot \mathbf{H}^{(t-1)}] + \mathbf{b}_C\big) \quad \mathbf{H}^{(t)} = \mathbf{u}^{(t)} \odot \mathbf{H}^{(t-1)} + (1 - \mathbf{u}^{(t)}) \odot \mathbf{C}^{(t)} \tag{5}$$

where $\mathbf{X}^{(t)}$, $\mathbf{H}^{(t)}$ denote the input and output of DCGRU (or GCGRU) at time t respectively, $\mathbf{r}^{(t)}$ and $\mathbf{u}^{(t)}$ denote reset gate and update gate at time t respectively, $*\mathcal{G}$ denotes the diffusion convolution (or ChebNet graph convolution), and $\mathbf{\Theta}_r$, $\mathbf{\Theta}_u$ and $\mathbf{\Theta}_C$ are the parameters for the corresponding convolutional filters.

Finally, as this is a sequence classification problem, our model consists of several layers of DCGRU (or GCGRU), followed by dropout, ReLU non-linear activation and a fully connected layer. We then apply pooling (max-pool or mean-pool) over the nodes and a Softmax function to obtain predicted probabilities of the seven seizure classes. Lastly, we train the model by minimizing the cross-entropy loss. For a single input (X, y), the cross-entropy loss is defined as follows:

$$L = \sum_{c=1}^{7} \Big(- y_c \log p_c \Big) \tag{6}$$

where y_c is 1 if the input belongs to class c, and 0 otherwise; p_c is the predicted probability for class c.

5 Experiments and Results

5.1 Experimental settings

We trained our model on the train split, and chose hyperparameters based on the dev split. We preprocessed EEG signals with time window $t = 1$ second, resampled frequency $f = 200$ Hz, maximum sequence length $T = 200$. Hence, each input EEG sequence has a dimension $200 \times 19 \times 200$ (see section 4.1). We used 4 layers of DCGRU (or GCGRU), each having 32 hidden units. The maximum diffusion step (i.e. K in Equations 2 & 3) was 2. The dropout probability was 0.5, and we used max-pooling at the end of the fully connected layer. We trained the model for 120 epochs with batch size 64 using the Adam optimizer. The initial learning rate was 8e-3 for undirected graphs, and 3e-3 for directed graphs.

Note that we also experimented with sequence-to-sequence architecture as that in the original DCRNN paper [10]. However, the model performance dropped significantly with sequence-to-sequence architecture. We think that this might be because certain seizure events are extremely short (e.g. 4 seconds in Figure 1), resulting in short EEG sequences. Thus, sequence-to-sequence architecture might not be suitable for our data.

In addition, our experiments showed that for CC and PC-based graphs, top-k graphs obtained better performance than thresholded graphs (see sections 4.2.2 & 4.2.3). Therefore, we report results of top-k graphs in next sections.

Finally, we experimented with max-pooling and mean-pooling in the pooling layer (see section 4.3 and Figure 3). Models with max-pooling achieved better performance than those with mean-pooling. This might be because certain seizures (e.g. focal non-specific seizure) start in a region in the brain, and thus max-pooling might be able to better capture the signals than mean-pooling.

5.2 Baselines

We compare our models with two baselines. The first baseline is a simple GRU with the same model architecture (e.g. number of GRU layers and hidden units etc.). The second baseline is SeizureNet [8], an ensemble of CNN-based models with multi-spectral features. Note that the authors applied 5-fold cross-validation in SeizureNet. Here, we implemented SeizureNet by ourselves and trained it from scratch on our own cross-patient data splits.

5.3 Model performances

As this is a multiclass classification problem with highly imbalanced classes, we use weighted F1 score, weighted precision and weighted recall scores as the main evaluation metrics. The weighted F1 score is calculated as follows:

$$F1_i = \frac{2 \times \text{precision}_i \times \text{recall}_i}{\text{precision}_i + \text{recall}_i} \text{ for } i \in \{1, ..., 7\} \tag{7}$$

$$\text{weighted } F1 = \sum_{i=1}^{7} F1_i \times w_i \tag{8}$$

where precision_i, recall_i and $F1_i$ are the precision, recall and F1 score for the i-th class respectively, and w_i is the weight of the i-th class depending on the number of positive examples in that class. Weighted precision and weighted recall are defined in the same way as weighted F1 score.

Table 2 shows the performances of baselines and our graph-based models on the test split. All of our graph-based models out-performed the baselines in terms of weighted F1 and recall scores, while SeizureNet achieved slightly better weighted precision score. These results suggest the effectiveness of modeling the spatiotemporal dependencies in EEG data with graphs.

Model	Weighted F1	Weighted Precision	Weighted Recall
Baseline GRU	0.37	0.44	0.36
Baseline SeizureNet	0.54	**0.64**	0.51
Distance-based (GCGRU)	0.60	0.62	**0.62**
CC-based top-3 (DCGRU)	**0.62**	0.63	**0.62**
PC-based top-3 (DCGRU)	0.58	0.59	0.58

Table 2: Model performance on test split. For distance-based graph, threshold $k_{dist} = 0.05$. For CC-based and PC-based graphs, we only show the results of the best graph (i.e. graph constructed by keeping top-3 neighbors of each node, see sections 4.2.2 & 4.2.3).

Figure 4 shows the confusion matrices of our best model (i.e. CC-based top-3 graph) and SeizureNet on the test split. Both models failed to make a single correct prediction for the two minority classes: SP and TC seizures. In general, our model out-performed SeizureNet for most of the classes except for CP seizure. Notably, for TN seizure which only occurred in one patient in the train split (see section 3.2), SeizureNet completely failed and confused it with TC seizures. In contrast, our model was able to correctly classify some of the TN seizures. Our model also achieved a significantly higher accuracy in predicting FN and AB seizures. The confusion matrices further suggest that our graph-based models might be better at modeling EEG data compared to the CNN-based model. Nevertheless, there is still much room for improvement for our models, especially in classifying minority seizure classes.

5.4 Effect of spatial dependency modeling

To investigate the effect of *spatial* dependencies modeling with graphs, we compare our models with the GRU baseline. All of our graph-based models out-performed the GRU baseline by a large margin (Table 2), suggesting the benefits of modeling spatial dependencies in EEG data with graphs.

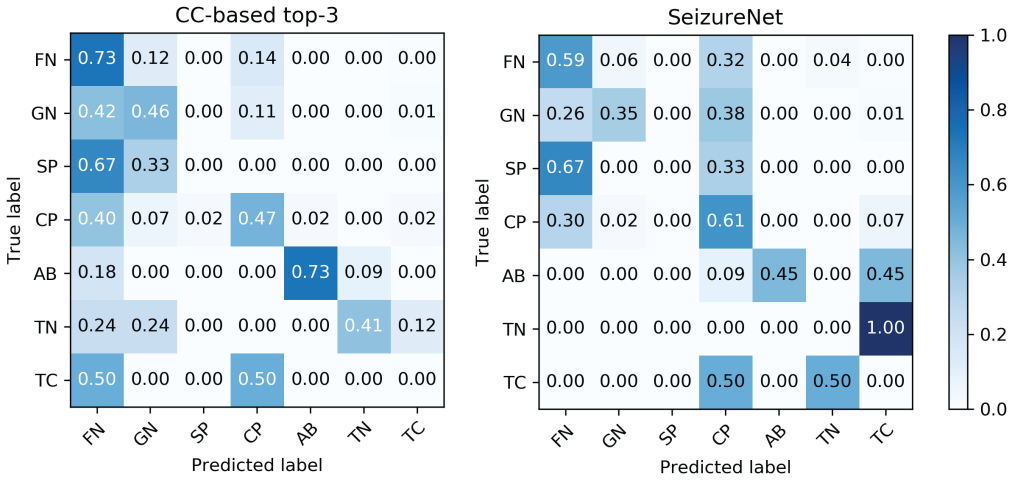

Figure 4: Confusion matrices of our model using CC-based top-3 graph and SeizureNet. Values shown here are normalized to be between 0 and 1.

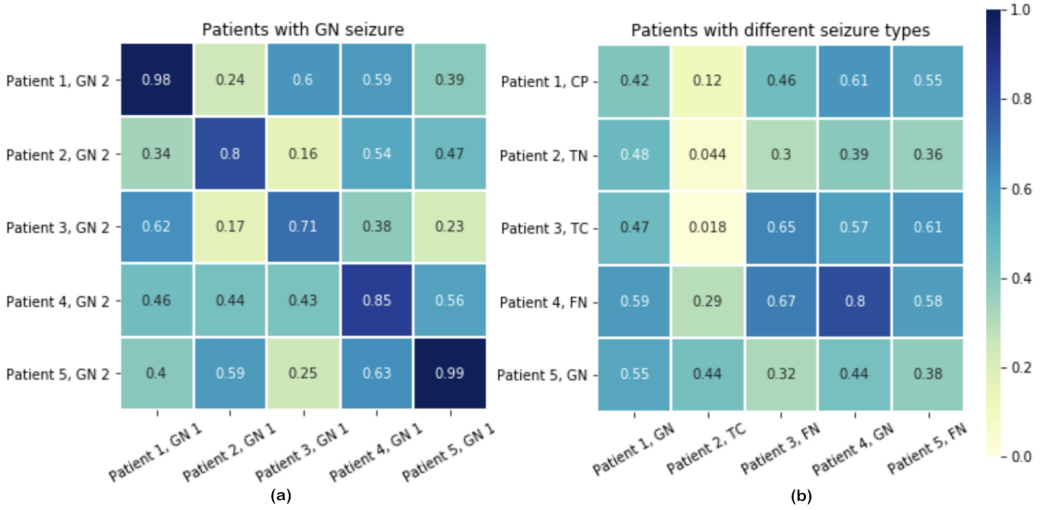

Figure 5: Consistencies of CC-based individual graphs of the same (a) or different (b) seizure types within- and across-patients. Values shown here are pairwise Pearson correlations between the graph adjacency matrices. (a) Correlations between graphs associated with five patients' two GN seizure events. We observe that graphs of the same seizure type are more consistent within-patient than across-patient. (b) Correlations between graphs associated with five patients' two different types of seizure events. We observe that graphs of different seizure types are less similar compared to graphs of the same seizure type.

5.5 Individual graph structures

Here, we look at the individual graph structures constructed using the CC-based method (section 4.2.2). First, to investigate the consistency of graphs of the *same* seizure type within- and across-patients, we randomly selected 5 patients who have at least 2 occurrences of GN seizure, and calculated the Pearson correlation between the adjacency matrices (whose values are the normalized cross-correlations) of the graphs associated with these seizure events. Figure 5a shows the resulting correlation matrix. For example, the first row represents the correlations between the graph associated with patient 1's second GN seizure event and the graphs associated with other patients' first GN seizure events. The diagonal values (i.e. within-patient) are much higher than the off-diagonal values (i.e. across-patient), indicating that individual graphs of the same seizure type are more consistent

within-patient than across-patient. Such inter-patient variability also indicates that the classification task is challenging by nature.

Next, to compare the graphs of *different* seizure types within- and across-patients, we randomly selected 5 patients who have at least 2 different seizure types, and calculated the Pearson correlation the same way as above. Figure 5b shows the resulting correlation matrix. For instance, the second row represents the correlations between the graph associated with patient 2's TN seizure event and the graphs associated with other patients' other types of seizure events. Compared to Figure 5a, we have two observations: (a) The diagonal colors in Figure 5b are in general much lighter, suggesting that graphs of different seizure types exhibit different structures, even within the same patient. (b) Graphs of the same seizure type are generally more similar compared to those of different seizure types. Take the last row as an example, graph associated with patient 5's GN seizure has a correlation of 0.55 with that for patient 1's GN seizure (i.e. same seizure type, different patients), while only has a correlation of 0.38 with the graph for his/her own FN seizure (i.e., different seizure types, same patient).

5.6 Visualization of learned localized filters

In this section, we visualize the learned diffusion convolutional filters $f_\theta \in \mathbb{R}^{K\times 2}$ in Equation 2 for the CC-based top-3 graph. Briefly, we compute feature maps of a graph input $\mathbf{X} \in \mathbb{R}^{N\times P}$ by convolving it with f_θ. Figure 6 visualizes examples of learned filters centered at different nodes for $K = 2$. Colors of nodes denote the values of the feature maps, and gray nodes are excluded because they are not among the top-2 neighbors of the center node. Looking at the magnitude of the feature maps (i.e. darkness of colors), we can observe that the feature maps are well localized around the center nodes, and diffuse to nodes that are further away.

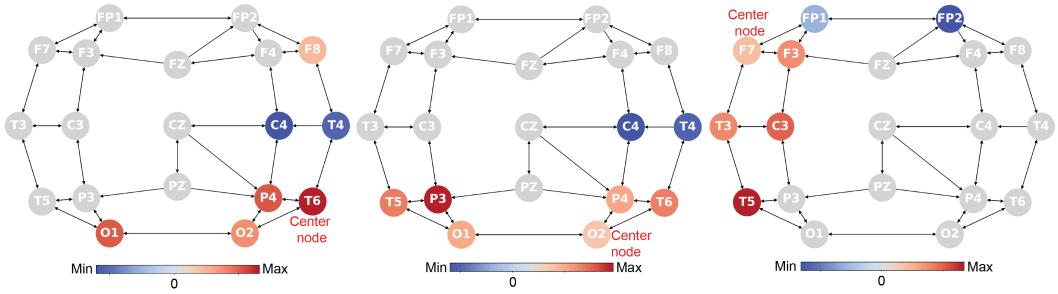

Figure 6: Visualization of learned diffusion convolutional filters centered at different nodes with $K = 2$ for the CC-based top-3 graph. Colors of nodes denote the values of the feature maps, and gray nodes are excluded from visualization. The feature maps are localized around the center, and diffuse away to nodes that are further away.

6 Conclusions and future work

In conclusion, we propose a novel way of modeling the spatiotemporal dependencies in EEG signals with graph convolutional recurrent neural networks. We model the spatial dependencies by employing the diffusion convolutions or ChebNet graph convolutions previously used for traffic forecasting, and the temporal dynamics by recurrent neural networks. In addition, we explored various ways of constructing graphs. Lastly, our extensive analyses showed that our graph-based models achieved better performance in multiclass seizure classification task compared to RNN or CNN-based models.

There are multiple interesting directions we would like to explore as next steps. First, in CC and PC-based graphs, we combined individual graphs by weighted averaging, which might introduce loss of information. A potential direction would be to extend the model to allow varying graph structures. Secondly, as our analyses in section 5.5 suggests, graphs of the same seizure type are more similar than graphs of different seizure types. Therefore, a reasonable direction would be to use a uniform graph structure for the same seizure type. Thirdly, we would like to investigate other ways of constructing directed graphs, such as using directed information in information theories. Lastly, inspired by recent advancement in natural language processing (e.g. BERT, GPT-2 etc.), we

are interested in applying techniques such as self-supervised learning to train a general model. The trained model could then be fine tuned on multiple downstream tasks such as seizure prediction and multiclass seizure classification.

7 Individual contributions

Siyi: Problem formulation, preliminary data analyses, coding up baselines, adapting DCRNN code and coding up models using distance-based and PC-based graphs, running experiments, performing post-analyses, writing up the report.

Kaylee: Brainstorming and coding up models using CC-based graphs with Louise and Siyi, performing post-analyses and visualization, writing up the report.

Louise: Brainstorming and coding up models using CC-based graphs with Kaylee and Siyi, performing post-analyses, coding up the framework for graph visualization, writing up the report.

8 Acknowledgement

We thank Dr. Jared Dunnmon, Dr. Rose Yu, Dr. Christopher Lee-Messer, Dr. Bibek Paudel, Prof. Daniel Rubin and Khaled Saab for their insightful discussions and comments.

References

[1] "Epilepsy," *World Health Organization*, 2019.

[2] I. C. Covert, B. Krishnan, I. Najm, J. Zhan, M. Shore, J. Hixson, and M. J. Po, "Temporal graph convolutional networks for automatic seizure detection," in *Proceedings of the 4th Machine Learning for Healthcare Conference*, vol. 106, pp. 160–180, 2019.

[3] S. Harrer, P. Shah, B. Antony, and J. Hu, "Artificial intelligence for clinical trial design," *Trends in Pharmacological Sciences*, vol. 40, no. 8, pp. 577 – 591, 2019. Special Issue: Rise of Machines in Medicine.

[4] N. Ahammad, T. Fathima, and P. Joseph, "Detection of epileptic seizure event and onset using eeg," *BioMed research international*, vol. 2014, 2014.

[5] F. Fürbass, P. Ossenblok, M. Hartmann, H. Perko, A. Skupch, G. Lindinger, L. Elezi, E. Pataraia, A. Colon, C. Baumgartner, *et al.*, "Prospective multi-center study of an automatic online seizure detection system for epilepsy monitoring units," *Clinical Neurophysiology*, vol. 126, no. 6, pp. 1124–1131, 2015.

[6] M. Golmohammadi, S. Ziyabari, V. Shah, E. Von Weltin, C. Campbell, I. Obeid, and J. Picone, "Gated recurrent networks for seizure detection," in *2017 IEEE Signal Processing in Medicine and Biology Symposium (SPMB)*, pp. 1–5, IEEE, 2017.

[7] S. Roy, U. Asif, J. Tang, and S. Harrer, "Machine learning for seizure type classification: Setting the benchmark," *arXiv preprint arXiv:1902.01012*, 2019.

[8] U. Asif, S. Roy, J. Tang, and S. Harrer, "Seizurenet: A deep convolutional neural network for accurate seizure type classification and seizure detection," *arXiv preprint arXiv:1903.03232*, 2019.

[9] P. Bashivan, I. Rish, M. Yeasin, and N. Codella, "Learning representations from eeg with deep recurrent-convolutional neural networks," *ICLR*, 2016.

[10] Y. Li, R. Yu, C. Shahabi, and Y. Liu, "Diffusion convolutional recurrent neural network: Data-driven traffic forecasting," *ICLR*, 2018.

[11] M. Defferrard, X. Bresson, and P. Vandergheynst, "Convolutional neural networks on graphs with fast localized spectral filtering," in *Advances in neural information processing systems*, pp. 3844–3852, 2016.

[12] B. Yu, H. Yin, and Z. Zhu, "Spatio-temporal graph convolutional networks: A deep learning framework for traffic forecasting," in *Proceedings of the 27th International Joint Conference on Artificial Intelligence*, 2018.

[13] S. Jang, S.-E. Moon, and J.-S. Lee, "Eeg-based video identification using graph signal modeling and graph convolutional neural network," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 3066–3070, IEEE, 2018.

[14] V. Shah, E. von Weltin, S. Lopez, J. R. McHugh, L. Veloso, M. Golmohammadi, I. Obeid, and J. Picone, "The temple university hospital seizure detection corpus," *Frontiers in Neuroinformatics*, vol. 12, p. 83, 2018.

[15] I. Obeid and J. Picone, "The temple university hospital eeg data corpus," *Frontiers in Neuroscience*, vol. 10, p. 196, 2016.

[16] D. I. Shuman, S. K. Narang, P. Frossard, A. Ortega, and P. Vandergheynst, "The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains," *IEEE signal processing magazine*, vol. 30, no. 3, pp. 83–98, 2013.

[17] R. Oostenveld and P. Praamstra, "The five percent electrode system for high-resolution eeg and erp measurements," *Clinical Neurophysiology*, vol. 112, no. 4, pp. 713 – 719, 2001.