CS63 Spring 2024
Local Spatial Relations in Time-Series EEG Data

Abdelrahman Abdelmonsef, Mehtap Yercel

May 6, 2024

1 Introduction

Convolutional Neural Networks (CNNs) are neural networks that use convolutional layers to extract and recognize common features in the input data. These networks are known to perform well on data exhibiting spatial locality relations, such as images, where the filters applied in the convolutional layers extract patterns from these relations (Meeden, Lisa, 2022; Mitchell, Ben, 2024).

Our project aims to investigate whether time-series data, denoting chronological data recorded over a time span (Stasko, John, 2013), exhibits spatial locality similar to images. Specifically, we aim to explore the spatial locality of electroencephalogram (EEG) signals. EEG measures brainwave activity via electrodes located on the participant's scalp, offering insights into brain function during specific tasks or conditions.

We will be using the EEGEyeNet dataset (Kastrati et al., 2021), which is a dataset containing EEG recordings from 356 adults between the ages of 18-80 that were asked to perform multiple eye-tracking tasks. For the purposes of our project, we will only be using data from the left/right antisaccade task. During this task, participants were instructed to look the opposite direction from a cue of either left or right that was displayed on a screen in front of them. The EEG net used in this task had 129 electrodes placed in a circular pattern around the participant's head, as shown in Figure 1, with each electrode recording EEG signals for a duration of the participants' performing the task (500 milli-seconds).

Our objective is to assess whether local spatial relationships persist within the relative positioning of electrodes around the participant's head and within the chronological sequence of EEG data. To this end, we'll create two additional versions of the dataset: electrode-shuffled and time-shuffled. The electrode-shuffled dataset involves rearranging data along the electrode axis, simulating a physical alteration in electrode placement within the EEG net. The time-shuffled dataset corresponds to shuffling the chronological ordering of data within EEG recordings, potentially distorting the representation of normal brain function.

By evaluating CNN performance across these datasets, we aim to validate or refute the presence of local spatial relations within time-series EEG data. A decline in the performance on shuffled datasets compared to the original dataset would imply the existence of disrupted spatial relations in the original data.
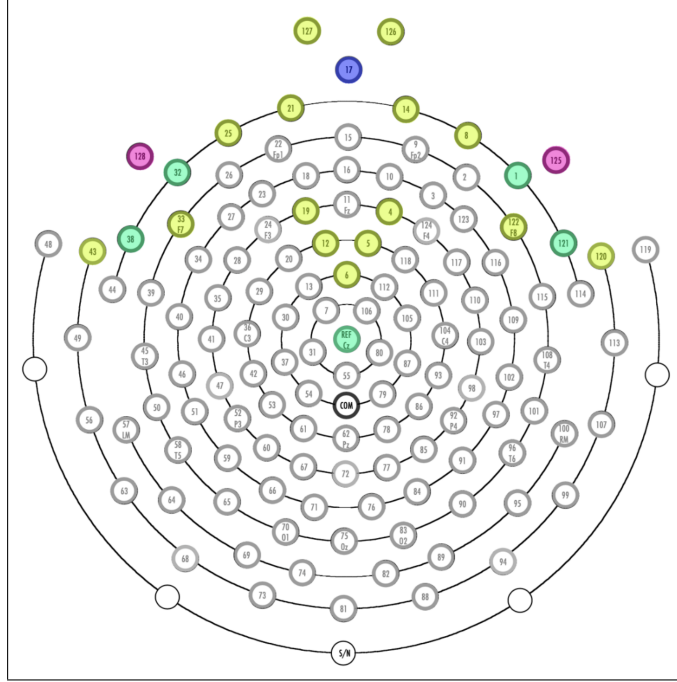
Figure 1: Layout of 129 electrodes on the EEG net (Kastrati et al., 2022).

2 Methods

In this section, we will detail the steps we took to select a CNN and prepare data for model training and evaluation.

2.1 Data Description

The EEGEyeNet dataset comprises 30,825 pre-processed EEG recordings, each containing 500 * 129 values. Each value corresponds to an EEG signal recorded from one of the 129 electrodes across 500 time points. The label for each recording is binary, denoted as 0 or 1, indicating whether the subject was looking left or right during the recording.

The EEGEyeNet GitHub repository (Kastrati, Ard, 2021) provides detailed information on the data acquisition and pre-processing procedures. To prepare the dataset for our experiments, we executed the following processing steps:

1. Due to the dataset's substantial size (15GB), we truncated it to utilize only the first 10,000 records in our experiments. This reduction ensured compatibility with TensorFlow's computational and memory constraints in our experimentation environment.

2. Subsequently, we generated two additional versions of the truncated dataset: one with data shuffled along the electrode axis and another with data shuffled along the time axis.

   To create the electrode-shuffled version, we randomly permuted the numbers from 0 to 128 (inclusive). Then, for each record in the dataset, we reordered the 129 electrode readings based

on this permutation while preserving the time-ordering of the data per electrode, thereby shuffling the electrodes' relative positions.

To generate the time-shuffled version, we generated a random permutation of the numbers from 0 to 499 (inclusive). For each record in the dataset, we reordered the 500 data points per electrode based on this permutation, thus maintaining the relative positions of electrodes while shuffling the time ordering per electrode.

3. For each of the three datasets described above, we partitioned the data into training, validation, and testing sets using a 70-15-15 split.

2.2 Model Description

For experimentation, we used a standard one-dimensional CNN, based upon the CNN used in Kastrati et al., 2021. The original CNN contains 12 modules with an additive shortcut layer following each three modules. This is later followed by global average one-dimensional pooling and a final dense layer with an output shape of 1 and a sigmoid activation function. Each module has (1D-)convolution, batch normalization, ReLU activation, and max pooling. For convolutions, 16 filters of size 64 were used, and for the pooling, a kernel of size 2 and stride of 1 were used. Each shortcut layer performs a convolution followed by batch normalization.

Upon applying the original CNN on the truncated dataset, it very quickly led to over-fitting. As a result, we adjusted the number of modules in the CNN to only 4 with only one shortcut layer, keeping everything else the same. Although training using this reduced-size model still results in over-fitting, the model's generalization capability improved compared to the bigger model. Also, the model showed over-fitting effects later, relative to the original CNN.

2.3 Training Procedures

We conducted training and testing of the CNN on three distinct versions of the time-series dataset: the original dataset and the two shuffled versions. Accuracy served as the metric to gauge the model's performance. Here, accuracy denotes the proportion of correctly classified labels (left/right) by the model when applied to EEG signal samples in the testing split.

The training-testing procedure was iterated five times for each dataset condition, yielding a total of 15 scores. Table 1 outlines the parameter configurations employed across all experiments. Subsequently, the accuracies obtained from the 15 runs were aggregated to calculate the mean per dataset and used for further statistical analysis.

Parameters	Settings
Learning rate	1e-4
Batch Size	64
Kernel Size	64
Num Filters	16
Epochs	30
Loss Function	Binary Cross-Entropy Loss (bce)

Table 1: Model parameters for training.

2.4   Statistical Analysis

The purpose of the statistical analysis procedures is to investigate whether the variations in scores across the experiments within each dataset hold statistical significance. To achieve this, we employed an Analysis of Variance (ANOVA) test, a widely recognized statistical tool for discerning significant differences among group means.

First, we ran the data through the Shapiro-Wilk test for assessing normality and Levene's test for evaluating the homogeneity of variances. Passing both tests validated the suitability of employing ANOVA to gauge statistical significance. We then conducted the ANOVA test, where a significance level of 0.05 was adopted as the threshold for determining statistical significance.

3   Results & Discussion

Table 2 shows the obtained testing accuracies for 5 runs of training and testing the CNN model on each of the 3 datasets, along with the average performance for each.

Runs	Original	Electrode Shuffled	Time Shuffled
Run 1	70.63	62.90	54.56
Run 2	71.03	67.99	51.52
Run 3	68.45	66.34	54.03
Run 4	72.02	62.50	55.09
Run 5	70.30	68.19	55.36
Average Testing Accuracy (%)	**70.49**	**65.58**	**54.11**

Table 2: Testing accuracy of the CNN model over 5 runs on three different dataset conditions.

As shown, on average, the CNN model performed the best on the original dataset as compared to the electrode-shuffled and time-shuffled versions. A one-way ANOVA test was applied twice: one comparing the original scores to the electrode-shuffled and another comparing the original scores to the time-shuffled. The p-values obtained were $p < 0.01$ and $p < 0.0001$, respectively. Furthermore, a one-way ANOVA was used to observe that the difference between the performances of the two shuffled datasets is statistically significant with a p-value of $p < 0.0001$. Using a significance level of 0.05, we can say that both differences are statistically significant.

These results suggest that the dip in performance experienced when the ordering of the data was shuffled across both the electrode and time dimensions can't be attributed to random chance. Rather, these differences can be attributed to the disruptions of the patterns arising from the local spatial relations within the original ordering of the data. Electrode-shuffling the data equates to shuffling the locations of the electrodes on the EEG net. Given that the CNN performed worse on the electrode-shuffled data, this might indicate that there are spatial patterns that arise from the locations of the electrodes around the participant's head that are disrupted when the electrodes are shuffled, which may cause the CNN to perform worse. Similarly, time-shuffling the data shuffles around the datapoints within each EEG, which essentially creates a new EEG that may be unrepresentative of actual brain functions. Given that the CNN performed worse on the time-shuffled data, this may indicate that there are important patterns within an authentic EEG recording that are disrupted when the time data is shuffled, which would explain the CNN's even

worse performance on this dataset compared to electrode-shuffled.

4   Conclusions

In this project, we were interested in examining whether time-series EEG data exhibits local spatial relations that are present in the relative-ordering of the electrodes and chronological EEG recordings per electrode.

To do so, we trained a CNN model on three different versions of the same EEG dataset to examine whether disrupting the ordering of the data would disrupt local spatial relations, as indicated by the CNN's worsening performance. Our CNN model performed better on the original time-series data with the local spatial relations intact compared to the electrode-shuffled and time-shuffled datasets. This supports the idea that there may be local spatial relations in the time-series dataset that allows the CNN to perform well on the left/right antisaccade classification task. In addition, our CNN model also performed better on the electrode-shuffled dataset as opposed to the time-shuffled dataset, which indicates that shuffling the chronological time-series data disrupts local spatial relations more than shuffling electrodes do.

We were only able to perform 5 runs per dataset condition; however, further repetitions of training the CNN should be done to confirm the findings of this project. Furthermore, in our initial planning of this project, we were also interested in comparing the performance of a feed-forward neural network to the CNN. While we had to forego this idea due to computational constraints, it would be interesting to compare the performance of our CNN on the EEG dataset to that of a feed-forward neural network to see if the ability of the CNN to recognize features based on local spatial relations would allow it to perform better.

5   Acknowledgements

References

Kastrati, A., Płomecka, M. B., Küchler, J., Wattenhofer, R., & Langer, N. (2022). Electrode clustering and bandpass analysis of eeg data for gaze estimation. *Proceedings of Machine Learning Research*, *1*, 1–16. https://doi.org/10.48550/arXiv.2302.12710

Kastrati, A., Płomecka, M. B., Pascual, D., Wolf, L., Gillioz, V., Wattenhofer, R., & Langer, N. (2021). Eegeyenet: A simultaneous electroencephalography and eye-tracking dataset and benchmark for eye movement prediction. *NeurIPS Datasets and Benchmarks*. https://doi.org/10.48550/arXiv.2111.05100

Kastrati, Ard. (2021, November). Eegeyenet. https://github.com/ardkastrati/EEGEyeNet

Meeden, Lisa. (2022, October). Deep learning. https://www.cs.swarthmore.edu/swatonly/meeden/cs63/lecture_f22/w08b.pdf

Mitchell, Ben. (2024, March). Convolutional networks. https://edstem.org/us/courses/51793/resources?download=34033

Stasko, John. (2013, November). Time series data. https://faculty.cc.gatech.edu/~stasko/7450/13/Notes/time.pdf