

Enhancing Neurophysiological Analysis and Emotion Recognition with EEG Channel Spatial Metadata through Graph Neural Networks

Leonid Sidorov* and Archil Maysuradze†

Lomonosov Moscow State University, GSP-1, 1-52 Leninskiye Gory, Moscow 119991, Russia

e-mail: *leon.sidorov@gmail.com, †maysuradze@cs.msu.ru

Abstract. The proposed architecture leverages the recording device's metadata by encoding the initial position of the recording electrodes into a graph structure which is then processed by the corresponding neural network architecture. This approach has proven its merit in neurophysiological applications such as P300 pattern detection and emotion recognition. Our experiments highlight the crucial role of the coordinate graph within the algorithm, which drastically influences the performance and efficacy of the model. Additionally, the versatility of our model is showcased through its consistent performance across diverse tasks, confirming its potential as a robust framework for future EEG-based research. Further analysis reveals that the incorporation of graph-based data alongside advanced optimization strategies markedly enhances the model's ability to generalize, making it a valuable asset for the neuroscience community. © 2024 Journal of Biomedical Photonics & Engineering.

Keywords: multimodal data integration; multivariate time series; electroencephalogram; spatial data; deep learning; graph neural network; P300 wave; emotion recognition.

Paper #8978 received 12 May 2023; revised manuscript received 1 Mar 2024; accepted for publication 13 Mar 2024; published online 8 May 2024. [doi: 10.18287/JBPE24.10.020304](https://doi.org/10.18287/JBPE24.10.020304).

1 Introduction

Nowadays, analysis of multivariate time series is often reduced to the identification of so-called functional patterns [1], that is, features of time series behavior corresponding to some states of the system of interest to the researchers. Over the years, experts from various subject areas have already identified a number of functional patterns. The task of this paper is to prove the worth of utilizing the spatial information about the input device.

That information can be represented in the form of a graph containing data about the order relationship of the components of the multivariate time series, in our case it is the electrodes on the patient's head. Such an approach should improve the generalizing ability of the model under study. Using an architecture that incorporates separate temporal and spatial analysis could be useful for classifying electroencephalogram (EEG) data, as it allows the model to analyse both the spatial structure of

the brain activity and the temporal structure of the EEG data.

A common approach is to use a convolution neural network (CNN) architecture that incorporates separate temporal and spatial analysis, as it allows the model to analyze both the spatial structure of the brain activity and the temporal structure of the EEG data. The spatial convolutional layers could be used to extract features from the EEG data that reflect the activity of different electrodes, while the temporal convolutional layer could be used to extract features that reflect changes in brain activity over time. The fully-connected layer could then combine these features to produce a final classification decision. This approach with the current dataset was used in Ref. [2].

The main approach of CNN models was to apply a 1×1 convolution in order to get the weighted sum of the channels and work with the one-dimensional time series later. However, an alternative operation was proposed later. A graph convolution layer can be applied before the

weighted sum of the channels. Models with such processing blocks are called graph convolutional networks (GCNs). Their spatial processing block works with the graph data extracted from the input dataset. GCN applications to EEG data are still an active area of research, and there are a lot of modifications, which we will observe in this paper.

The GCN model could be trained on a dataset that includes both EEG data and measures of brain connectivity, such as functional connectivity (measures of the correlations between brain activity in different regions) or structural connectivity (measures of the physical connections between brain regions). The GCN could then be used to analyse this connectivity information and incorporate it into the model's prediction or classification decisions. Our approach is based on structural connectivity which could be extracted from spatial coordinates of electrodes. That information often accompanies the EEG datasets.

2 Related Work

The prevailing method for processing EEG data via graph convolutional networks is through a graph learning technique [3, 4]. In this method, the graph is constructed concurrently with the training process. Typically, the adjacency matrix is parameterized using the following Eq.:

$$A_{ij} = \frac{\exp(-\text{ReLU}(\omega^T |\psi_i - \psi_j|))}{\sum_{k=1}^N \exp(-\text{ReLU}(\omega^T |\psi_i - \psi_k|))},$$

where ψ_i and ψ_j represent the node features, N denotes the total number of nodes in the graph, and ω is a learnable weight vector. To optimize the learning of this weight vector, it is common practice to modify the model's loss function to ensure desirable properties such as sparsity and smoothness in the adjacency matrix:

$$L_{\text{gcn}} = \lambda \sum_{j=1}^N \|\psi_i - \psi_j\|_2^2 A_{i,j} + \|A\|_F^2,$$

where $\|\cdot\|_F$ signifies the Frobenius norm.

This technique bears similarities to the attention mechanism and is designed to highlight the most relevant channels in the time series data. However, unlike our proposed method, the graph constructed in this manner does not provide new information beyond what is obtained during the model's training phase. It essentially represents a sophisticated parameterization of the input data. Furthermore, the evolving and highly sparse nature of this graph has a negative impact on the model's interpretability.

The study in Ref. [5] introduced the Pyramidal Graph Convolutional Network (PGCN), which leverages prior knowledge regarding the brain's mesoscopic regions and electrode positions. In their work, the brain is segmented into either 7 or 2 mesoscopic regions [6, 7], and an

adjacency matrix based on spatial proximity is learned along with the other parameters of the model.

The spatial adjacency matrix is defined as:

$$A_{ij} = \begin{cases} 1 & \text{if } A_{ij} \geq 1, \\ \frac{\delta}{d_{ij}^2} & \text{if } 0.1 \leq A_{ij} \leq 1, \\ 0.1 & \text{if } A_{ij} \leq 0.1, \end{cases} \quad (1)$$

where d_{ij} represents the Euclidean distance between nodes i and j and δ is a hyperparameter referred to as the sparsity factor. Subsequently, the authors proposed converting this adjacency matrix into a learnable parameter without any other modifications of the method. Our experiments indicate that allowing gradient flow through the adjacency matrix selectively emphasizes the most informative channels and results in a modest enhancement of the model's performance.

Our goal is to validate the utilization of spatial-based graphs in EEG applications and highlight the interpretability and performance benefits of such an approach. Unlike the PGCN paper [5], we will concentrate exclusively on spatial features and explore potential refinements to this methodology.

3 Considered Dataset

3.1 BCI Competition III Dataset II

This section describes the dataset used in the work and the corresponding tasks, the solution of which, as neurophysiologists already know today, is associated with the functional pattern "wave P300".

There were 185 tests for each of the two subjects, A and B. The authors of the study have already proposed a partition of 85+100. Due to this partitioning, the random guessing method has only 2.8% accuracy on the test sample. We will use the partitioning provided by the authors to be able to compare our results with other algorithms applied on the same dataset. All EEG signals were collected using a 64-electrode scalp, filtered in the range from 0.1 to 60 Hz and digitized at a frequency of 240 Hz. You can read more about this in [1].

According to research by neurophysiologists, it is assumed, but not formally guaranteed, that highlighting the target symbol gives a "P300 wave". Although we know when to expect the P300 signal, its appearance depends on the subject, and the subject himself does not control this process either. The production of the "P300 wave" is not a phenomenon of consciousness, it occurs due to external stimuli (flickering of rows and columns). In addition, the "P300 wave" interferes with many other signals, correspondingly, its severity can vary greatly.

The problem is to determine whether the target character is contained in the highlighted row or column or not. In this case, the problem can be considered as a binary classification. The input is an EEG

corresponding to the highlighting of one row or column, and the output is a binary class label (whether the current highlighting is the target). As mentioned before, a 600 ms record is allocated for each highlighting, which is longer than the duration of the highlighting itself. The last object of the epoch will not be cut off, because each recording ends with a pause of 2.5 s, during which the matrix goes out.

3.2 Seed

For additional architecture validation we used the SEED dataset [8] for the emotion recognition task. That task has the same input data in the form of a raw EEG signal.

The SEED dataset is a publicly available EEG emotion dataset primarily designed for discrete emotion models. Signal data was downsampled to 200 Hz and band-pass filtered within the range of 0 to 75 Hz. The dataset encompasses three emotional types: positive, neutral, and negative.

For the EEG signals elicited by emotional stimuli from each subject within the dataset, the initial step involves capturing non-overlapping EEG signals of duration 1 second. Subsequently, we extract Differential Entropy (DE) features within five frequency bands (namely, δ (1–3 Hz), θ (4–7 Hz), α (8–13 Hz), β (14–30 Hz), and γ (31–50 Hz) rhythms) from each time slice and map them into a spatial matrix [9]. If we denote raw EEG data as X and suggest that the signals of each subband approximately follow a Gaussian distribution $X \sim N(\mu, \sigma^2)$, the formula could be written as follows:

$$\begin{aligned} DE(X) &= \\ &= \int_{-\infty}^{\infty} \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) \log \frac{1}{\sqrt{2\pi\sigma^2}} \exp\left(-\frac{(x-\mu)^2}{2\sigma^2}\right) dx = \\ &= \frac{1}{2} \log(2\pi e \sigma^2), \end{aligned}$$

where π and e are constants, and σ^2 represents the variance of X .

4 Formula Problem Statement

P300 statement is a binary classification problem, where 1 means the presence of an impact, and 0 means its absence. One sample is a multivariate time series of fixed length, so each observation can be represented by a matrix of size $N \times T$, where N is the number of channels (electrodes), and T is the number of time intervals in which readings were recorded.

Analogically, *emotion recognition statement* is a classification problem with three classes. Negative, neutral and positive emotions stand for 1, 2, and 3 accordingly. In this setup authors also could not guarantee the appearance of a specific emotion from the subject. In that case the number of time intervals T represents the differential entropy for every frequency band δ , θ , α , β , and γ for every one-second slice of raw

EEG data. There is no neighbourhood relationship as well.

In the classic problem statement there is no neighbourhood relationship in the data. We can set some chain connection between the electrodes at best. This can be achieved with the channel sorting of the input matrix.

Having all the previously introduced terms at our disposal, we will rewrite the initial task as the graph neural networks statement. Let us denote the graph, which represent the initial neighborhood relationship, as G . G has a well-defined structure, and is also undirected, homogeneous, and static in time. Having a graph G with a one-dimensional time series associated with each vertex, we want to label it with one of two classes, where 1 represents the presence of an impact, and 0 represents its absence. That is, the problem is solved at the graph level with the teacher.

In our terminology, one sample is a graph. All objects have a structure G , but a unique one-dimensional time series is assigned to each vertex. Thus, as in the original problem statement, each observation is represented as a matrix of size $N \times T$, where N is the number of channels or electrodes, and T is the number of time intervals in which readings were recorded. However, this time a less trivial neighbourhood relationship G is set on the electrodes. Graph G is also fed to the algorithm. We will proceed to the construction of the graph G in the Section 6.

5 Proposed Architecture

As previously mentioned, we propose a model family that can be described with three types of modules: spacial processing block, temporal processing block and predictor block, which is a multilayer perceptron (MLP).

Our task is investigating of GCN capabilities, so the proposed architectures are quite simple. However, almost any existing EEG processing neural network could be expressed with such a structure.

The base model (Fig. 1) consists of 1×1 convolution as a spatial processing block and one-dimensional convolution as a temporal processing block. The predictor block is identical for all approaches and consists of multiplying the object vector by the matrix of the parameters being trained. Such an operation is defined only for vectors, but it will be applied at the last stage of object transformation, when both the graph in our problem and the matrix in the original one will be reduced to a one-dimensional embedding vector. Also, each convolution is accompanied by a nonlinear function ReLU [10].

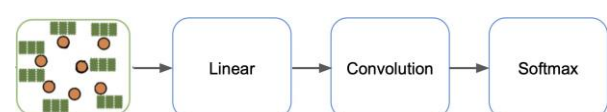


Fig. 1 Base CNN model.

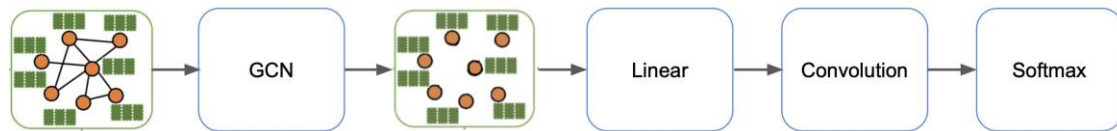


Fig. 2 GCN extended model.

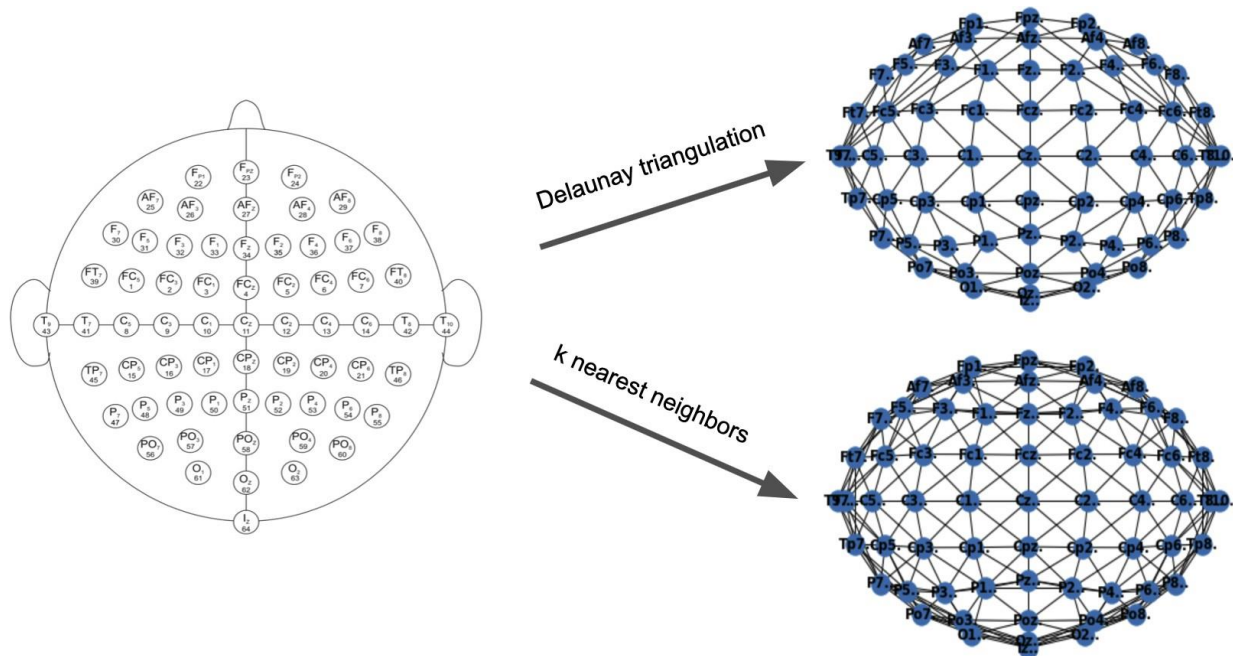


Fig. 3 10-10 mounting and obtained graphs.

Another approach is to add a graph convolution layer from GCN before the 1×1 convolution in the spatial processing block (Fig. 2). That could potentially improve the performance of the model by allowing it to capture more complex patterns of brain activity. The model can incorporate connectivity information into the input features. The GIN convolution from [11] was used because of its superior expressive ability.

6 Graph Construction

We will denote by G the graph representing the 10-10 standard. The vertices of this graph are electrodes, each vertex is encoded by a vector that contains the values of the potentials measured at each time. That is, each vertex of the graph is mapped to a time series recorded during the recognition of the P300.

In this paper, we obtain graph G in two ways: using Delaunay triangulation and the k nearest neighbours method, and compare the results (Fig. 3).

Medical data usually have fairly detailed specifications. For example, in our case, together with the data there was an article describing the procedure for collecting them, and a file with the coordinates of each electrode. It is the spatial information that we want to include in the neural network model in order to improve the quality of the received predictions.

Thus, by spatial coordinates we should get a graph G that characterizes the structure of the device used in

the experiments. We emphasize that the graph is built on a three-dimensional object, taking into account the form factor of the neurointerface, on a two-dimensional manifold.

We want to construct a graph with respect to the electrodes' neighbourhood, that is, there is an edge between two vertices if and only if they are neighbours in terms of coordinates. Now we will formalize the concept of a "neighbour".

Fortunately, humanity already knows how to solve such a problem, and we have considered two common solutions.

- The first approach is *Delaunay triangulation* [12]. This algorithm allows us to build a planar graph consisting only of triangles. Moreover, a circle can be described around each triangle, which will not include any other points of the triangulation, except for the vertices of this triangle. That is, the resulting graph will be sparse.

- The second approach is based on the classical *k nearest neighbours* algorithm, which calculates the distances to all objects in the dataset and searches for the k nearest among them. Since we are working with a real physical object, the Euclidean distance was chosen. The value of parameter k equal to 9 was selected experimentally.

Now, having a graph G , we can proceed to the formal formulation of the problem of pattern recognition.

7 Experiments

This section is devoted to the comparative analysis of selected approaches as applied to two datasets, which were detailed in Section 3. For the BCI Competition dataset, we adhere to the pre-defined split between training and testing sets. This standard enables us to benchmark our algorithms against existing strategies, including those developed in conjunction with the 2004 competition. The batch size is configured to 1024 samples. Adhering to the oddball paradigm, there are 2 positive class observations for every 12 objects, alongside 10 negative class observations, yielding a significant class imbalance of 1:6. As for the SEED dataset, it encompasses three balanced classes with a batch size set to 128, and the training partition is threefold larger than the testing partition.

We deploy consistent hyperparameters across both datasets: a learning rate of 10^{-4} and a weight decay of 10^{-2} . To bolster computational stability, we employed a learning rate scheduler with a *step size* of 5 and a γ parameter of 1. Both tasks converged within 500 epochs. For the “P300 wave” detection task we employed MSE loss, while for the SEED dataset CrossEntropy loss showed more promising results. With the training methodology established, we now embark on the empirical comparison of the proposed network architectures on the described datasets.

Initially, we evaluate our proposed models on the P300 dataset, aligning them with classical method of brain signal analysis such as gradient boosting.

For a point of reference, we utilize a baseline GCN model equipped with an identity matrix. Under this configuration, all models maintain an equivalent parameter count. However, the baseline model effectively translates to a graph void of connections. Additionally, our experiments incorporate Delaunay and KNN graphs. We also investigate a dense graph as described in Eq. (1), and a randomly initialized matrix drawn from a uniform distribution $\mathcal{U}(-\sqrt{k}, \sqrt{k})$, where $k = \frac{1}{N}$ and N represents the number of nodes. This is a standard practice for initializing the weights in the dense layers of neural networks.

For all architectures, excluding gradient boosting, we further employed gradient flow through the adjacency matrix to examine its effect as pointed out in Ref. [5].

Table 1 F1-score for binary classification problem.

Model	without gradient		with gradient	
	Subject A	Subject B	Subject A	Subject B
Gradient Boosting	28.03	24.04	—	—
Dense GCN	31.07	25.83	37.16	25.64
Baseline GCN	34.15	24.72	36.31	24.85
Delaunay GCN	34.93	25.25	37.20	25.26
KNN GCN	33.71	25.07	36.79	24.71
Random GCN	34.65	24.87	36.87	24.85

The F1-score was employed as the primary evaluation metric to accommodate the pronounced class imbalance observed within the first dataset. Insights drawn from Table 1 highlight that graph structure exerts a notable influence on model performance. While the KNN GCN showed relatively modest results, the Dense and Delaunay graphs consistently outperform the Baseline model across all experimental setups. As anticipated, gradient boosting yielded the lowest F1-scores.

Employing the gradient-based enhancement significantly improves results across models, so the distinction between their performance gradually reduces. Incremental improvements are notably observed in the baseline model endowed with an identity matrix. Notably, the random graph model delivers satisfactory performance across the board.

For Subject B, all models display modest results, with the Dense model exhibiting a comparatively advantageous edge. The narrow range of results may possibly be explained by poor dataset targets and model overfitting.

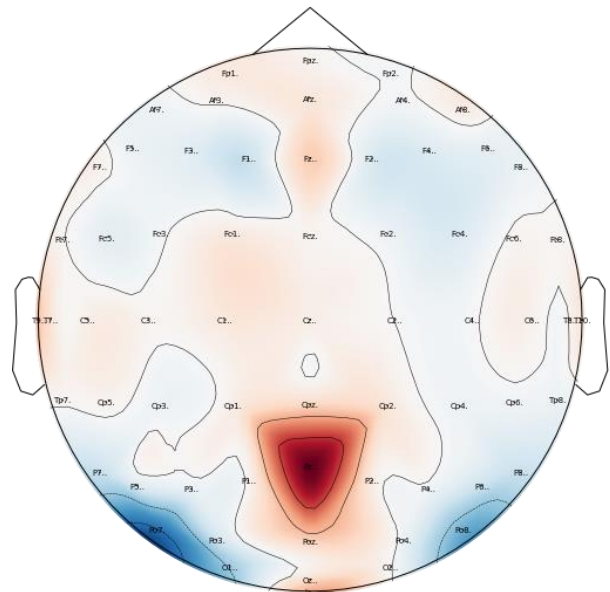


Fig. 4 The distribution of weights assigned to different features in the baseline model for the task of character recognition.

Table 2 Accuracy metric for emotion recognition problem.

Model	95% CI	
	without gradient	with gradient
Dense GCN	94.81 ± 3.74%	94.81 ± 3.74
Baseline GCN	83.83 ± 8.76%	90.37 ± 4.98%
Delaunay GCN	94.81 ± 3.74%	94.81 ± 3.74
KNN GCN	94.81 ± 3.74%	94.81 ± 3.74
Random GCN	87.41 ± 5.60%	94.81 ± 3.74
PGCN [5]	—	94.81 ± 3.74%

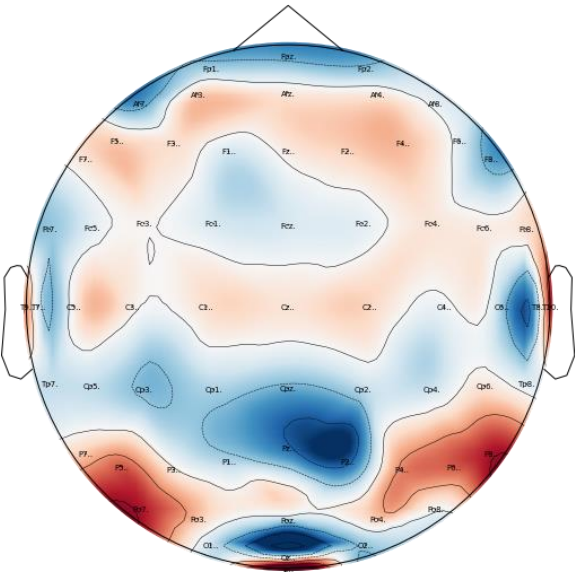


Fig. 5 The distribution of weights assigned to different features in the GNN model for the task of character recognition.

In addition to improving the quality of the model under consideration, the use of graph-based architecture allowed us to obtain more interpretable results. As stated in Ref. [13], the most informative electrodes in the recording device are Fz, Cz, P3, P4, Po7, Po8, Pz and Oz. In Fig. 4 we can notice that the basic architecture in our approach focused its attention on only three electrodes Pz, Po7, and Po8. While the graph neural network in Fig. 5 was able to recognize all of the above electrodes, except Fz and Cz.

The source of these visualizations is the 1×1 convolutional layer’s weight matrix, which yields higher absolute values to electrodes of greater importance during the weighted sum. In these figures, blue signifies negative values while red designates positive ones. However, our interest lies predominantly in their absolute values due to the inherent characteristics of the neural network training process.

The results in Table 2 were more unambiguous because the SEED dataset turned out to be too plain for

the considered model family. We used accuracy for multi-classification as the base metric for this task.

Distinct variations in model performance emerged with the baseline GCN, which lacked spatial information about channel structure, being an outlier. Noteworthy is the Random GCN, which achieved parity with the top-performing models with the gradient enhancement.

We further benchmarked our results against cutting-edge research in the domain of emotion recognition, notably [5]. The results show that our enhancements allow our models to rival state-of-the-art architectures such as PGCN in terms of accuracy, even though the high scores may suggest a saturation point for this particular dataset.

Inspection of the weight allocation maps further unraveled the models’ profound grasp of the data. There was a significant activation of nodes located in the temporal and occipital lobes and partial activation of nodes in the frontal lobe, aligning with the sources of different emotional impulses as stated in Ref. [8]. The spatial processing module in the GCN revealed activation of additional nodes, potentially contributing to the observed uptick in model accuracy.

8 Adjacency Gradient Interpretation

In this section, we will consider the adjacency gradient, which was derived through gradient flow from a random initialization. During the experiments, this was referred to as the “Random GCN”. To highlight the most substantial connections within the graph, we discretized the matrix values using the 95th percentile threshold.

As we can see from the Fig. 6, the matrix evolves over the course of learning. However, in the absence of explicit constraints, the matrix becomes asymmetric. Notably, the configuration that appears in the learned matrix is reminiscent of an attention mechanism, suggesting that the network identifies and prioritizes inter-channel connections that are important for the targeted task.

Dominating channels within the matrix, such as Fz, Po7, Po8, and Pz, coincide with four of the eight channels that neurophysiologists have pinpointed as crucial in the detection of the P300 wave [13], a finding that was also validated via analysis of the channel summation vector.

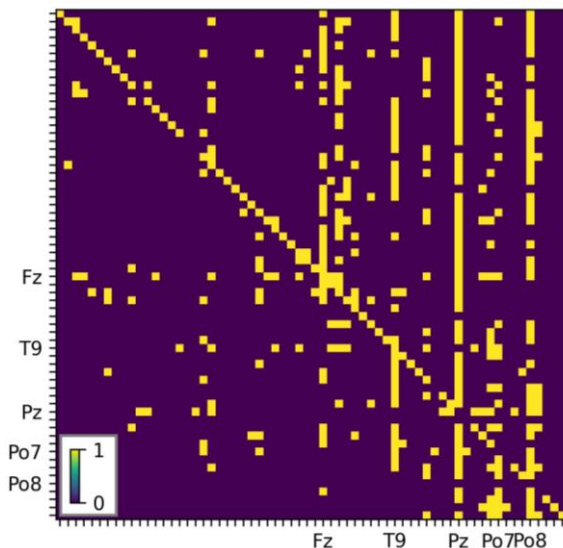


Fig. 6 Discretized version of the learned adjacency matrix highlighting connections between channels crucial for the task.

This congruence with neurophysiological understanding lends credence to the notion that the patterns discerned in the adjacency matrix are valid.

The autonomous functioning of this architecture segment as a distinct feature is promising. Future enhancements in parameterization and loss function adjustments may further refine the matrix structure, thereby augmenting the positive impact of this technique in the future research.

9 Conclusions

In conclusion, our study demonstrated the effectiveness of a unified model architecture in independent detecting the P300 wave and classifying emotions using EEG data. The incorporation of additional graph-based data and the utilization of specialized optimization techniques significantly enhanced the model's generalization abilities, as evidenced by the spatial filter weights and improved performance metrics.

Our experiments confirmed that graph structures play a crucial role in the success of Graph Convolutional Networks, markedly influencing model outcomes. The use of the gradient trick, in combination with the proposed approach, resulted in notable gains in model performance. This improvement allowed our models to compete with existing state-of-the-art models, suggesting a promising direction for future research in this domain.

Acknowledgments

The research is supported by Scientific and educational school of Moscow State University "Brain, cognitive systems, artificial intelligence", research work of Moscow State University 5.1.21.

Disclosures

The authors declare no conflict of interest.

References

1. B Blankertz, KR Muller, DJ Krusienski, G Schalk, JR Wolpaw, A Schlogl, G Pfurtscheller, Jd. R. Millan, M. Schroder, and N. Birbaumer, "The bci competition iii: Validating alternative approaches to actual bci problems," *IEEE Transactions on Neural Systems and Rehabilitation Engineering* 14(2), 153–159 (2006).
2. H. Cecotti, A. Graser, "Convolutional neural networks for p300 detection with application to brain-computer interfaces," *IEEE Transactions on Pattern Analysis and Machine Intelligence* 33(3), 433–445 (2010).
3. P. Mathur, T. Mittal, and D. Manocha, "Dynamic graph modeling of simultaneous eeg and eye-tracking data for reading task identification," *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 1250–1254 (2021).
4. W. Ye, Z. Zhang, M. Zhang, F. Teng, L. Zhang, L. Li, G. Huang, J. Wang, D. Ni, and Z. Liang, "Semi-supervised dual-stream self-attentive adversarial graph contrastive learning for cross-subject eeg- based emotion recognition," *arXiv preprint arXiv:2308.11635* (2023).
5. M. Jin, E. Zhu, C. Du, H. He, and J. Li, "Pgcn: Pyramidal graph convolutional network for eeg emotion recognition," *arXiv preprint arXiv:2302.02520* (2023).
6. P. Hagmann, L. Cammoun, X. Gigandet, R. Meuli, C. J. Honey, V. J. Wedeen, and O. Sporns, "Mapping the structural core of human cerebral cortex," *PLoS Biology* 6(7), e159 (2008).
7. G. E. Bruder, J. W. Stewart, and P. J. McGrath, "Right brain, left brain in depressive disorders: clinical and theoretical implications of behavioral, electrophysiological and neuroimaging findings," *Neuroscience & Biobehavioral Reviews* 78, 178–191 (2017).
8. W. L. Zheng, B.-L. Lu, "Investigating critical frequency bands and channels for EEG-based emotion recognition with deep neural networks," *IEEE Transactions on Autonomous Mental Development* 7(3), 162–175 (2015).
9. R. N. Duan, J. Y. Zhu, and B. L. Lu, "Differential entropy feature for EEG-based emotion classification," *2013 6th International IEEE/EMBS Conference on Neural Engineering (NER)*, 81–84 (2013).
10. A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," *Communications of the ACM* 60(6), 84–90 (2017).

11. K. Xu, W. Hu, J. Leskovec, and S. Jegelka, “[How powerful are graph neural networks?](#)” arXiv preprint arXiv:1810.00826 (2018).
12. B. Delaunay, “[Sur la sphere vide](#),” Izvestia Akademicheskikh Nauk SSSR, Otdelenie Matematicheskii i Estestvennyka Nauk 7(793-800), 1–2 (1934).
13. F. Sharbrough, G. E. Chatrian, R. Lesser, H. Luders, M. Nuwer, and T. W. Picton, “[American electroencephalographic society guidelines for standard electrode position nomenclature](#),” Journal of Clinical Neurophysiology 8(2), 200–202 (1991).