Graph Attention Based Spatial Temporal Network for EEG Signal Representation

Msonda, James; He, Zhimin; Lu, Chuan

Publication date:
2023

Graph Attention Based Spatial Temporal Network for EEG Signal Representation

James Ronald Msonda[1,2] [0000-0002-0470-1053], Zhimin He [3] [0000-0001-9713-7457] and Chuan Lu [2] [0000-0002-4898-6679]

[1] Department of Computer Science, Aberystwyth University, Wales, UK
[2] Malawi University of Business and Applied Sciences, Blantyre, Malawi
[3] Department of Psychology, Aberystwyth University, Wales, UK
{jam139,zhh2,cul}@aber.ac.uk

Abstract. Graph attention networks (GATs) based architectures have proved to be powerful at implicitly learning relationships between adjacent nodes in a graph. For electroencephalogram (EEG) signals, however, it is also essential to highlight electrode locations or underlying brain regions which are active when a particular event related potential (ERP) is evoked. Moreover, it is often important to identify corresponding EEG signal time segments within which the ERP is activated. We introduce a GAT Inspired Spatial Temporal (GIST) network that uses multilayer GAT as its base for three attention blocks: edge attentions, followed by node attention and temporal attention layers, which focus on relevant brain regions and time windows for better EEG signal classification performance, and interpretability. We assess the capability of the architecture by using publicly available Transcranial Electrical Stimulation (TES), neonatal pain (NP) and DREAMER EEG datasets. With these datasets, the model achieves competitive performance. Most importantly, the paper presents attention visualisation and suggests ways of interpreting them for EEG signal understanding.

Keywords: EEG models, Electroencephalography, Graph neural networks, Attention mechanism, Interpretable machine learning

1 Introduction

The success of deep neural networks (DNN) at learning from data in areas such as image classification, natural language processing, audio classification and speech generation etc., has been attributed to the applicability of convolution operations to a common system of coordinates in the Euclidean space [1]. However, representing electroencephalogram (EEG) signals within the same n-dimensional linear space fails to capture vital information including strengths and directionality of relationships between electrode locations within and between underlying brain regions.

Geometric deep learning (GDL) was conceptualized to replicate the achievement of erstwhile vanilla deep learning in the non-Euclidean space, often dealing with graph structured data [2]. A graphical representation of EEG data encapsulates interlinks and structural organization between EEG electrodes (channels).

This paper introduces a GAT Inspired Spatial Temporal (GIST) network, which draws inspiration from the success of Graph Attention (GAT) network at expressing the strength of the connections between nodes through self-attention. The architecture also has a node attention layer to learn the importance of individual nodes (electrodes), followed by a temporal attention layer which focusses on informative time windows. This completes the EEG feature representation learning part of the architecture. For EEG signal classification, a multilayer perceptron is added to the top of the model. In this paper, we make the following contributions: 1) propose a novel nearly transparent graphical model for high level EEG signal feature learning; 2) assess the feature learning ability of the model by using 3 different EEG datasets; 3) suggest ways through which the learned attention weights can be leveraged for model diagnosis and interpretation; and 4) demonstrate the practical use of the model on real world problems (a) identifying regions associated with physical pain; and (b) channel selection for emotion classification.

2 Related Work

2.1 Graph Neural Networks

In a graph structure, the nodes, or vertices, are linked together by edges. Nodes and edges often have multi-dimensional features. While static forms of graphs provide informative spatial representations, better insight emanates from the time-varying structural changes of such graphs e.g., adding/removing a node and creating/modifying an edge or their features or weights.

Graph neural networks (GNN) were devised to bring convolutional neural network (CNN) like operations to the domain of geometric deep learning and thrive on the principle of message passing. In the message passing operation, node features of all of the target node's neighbours are aggregated to create new features for the said node. For the current node to learn from features of a node two hops away (neighbour of a neighbour), two iterations would be needed. Stacking these message passing layers enables the current node to learn even from the entire graph. It follows therefore, that the resulting embedding in a GNN encodes both the node features and existing node to node relations. Common GNN architectures include Graph Convolution Networks (GCN) [3] and Graph ATtention networks (GAT) [4].

2.2 Graph Attention Networks

Unlike GCN, where all neighbours of a node are given equal importance, in GAT, the features of the target node's neighbours are given learned weights before aggregation. Weighting is done through an attention mechanism.

To illustrate this point, assume that the input to a GAT layer is a collection of node features $\boldsymbol{h}=h_1, h_2, h_3, ..., h_N,\ h_i \in R^F$ where N denotes the total number of nodes in a graph, F the number of features available in each node. Equations (1)-(4) summarise how node embeddings $h^{(l+1)}$ are obtained given features h^l at a lower layer l.

$$z_i^{(l)} = W^{(l)} h_i^{(l)} \tag{1}$$

$$e_{ij}^{(l)} = \text{LeakyReLU}(\vec{a}^{(l)^T}(z_i^{(l)} \| z_j^{(l)})) \tag{2}$$

$$\alpha_{ij}^{(l)} = \frac{\exp(e_{ij}^{(l)})}{\sum_{k\in\mathcal{N}(i)} \exp(e_{ik}^{(l)})} \tag{3}$$

$$h_i^{(l+1)} = \sigma(\sum_{j\in\mathcal{N}(i)} \alpha_{ij}^{(l)} z_j^{(l)}) \tag{4}$$

As shown in (1), a learnable weight, $W^{(l)}$ is used to convert input features from a low level to a higher level to improve their expressivity. The outputs of (1) from two adjacent nodes, i and j are concatenated ($z_i^{(l)} \| z_j^{(l)}$), and an additive attention score $e_{ij}^{(l)}$ is obtained by taking the dot product of the result and a learnable weight, $\vec{a}^{(l)}$ before applying LeakyRelu activation function. In (3), SoftMax is used to normalize the scores attained in (2) across all single hop neighbours of i, $\mathcal{N}(i)$. Finally, normalized attention scores $\alpha_{ij}^{(l)}$ are used to provide weights to adjacent node embeddings ($z_j^{(l)}$) which are further aggregated as shown in (4), to get new embeddings for the target node where σ is an activation function such as Rectified Linear Unit (ReLU).

The attention score as calculated in GAT indicates the importance of a node to its neighbour, thus it can also be regarded as learned edge weight between concerned nodes. We refer to this as a form of edge attention. In case of multiple attention heads, node embeddings can be obtained by combining the outputs of the heads by concatenation or averaging, as shown in (5), with H as the number of heads.

$$h_i^{(l+1)} = \sigma(\frac{1}{H}\sum_{k=1}^{H} \sum_{j\in\mathcal{N}(i)} \alpha_{ijk}^{(l)} z_j^{(l)}) \tag{5}$$

2.3 EEG Graph Models

Zhang et al. [5] proposed a Graph based Hierarchical Attention Model (G-HAM) which encodes channel connectivity as either Euclidean distance between electrodes' spatial positions or structural neighbourhood. The node features are the raw signals. These node signals are sliced before conventional CNN is applied to them for feature extraction. This layer is followed by attention mechanisms which isolate important time slices and nodes. However, with the edge weights remaining constant across temporal slices and trials during graph formation, the model does not capture the dynamic nature of relations between brain regions, and there is no guarantee for a link between spatial and functional relationships.

Dynamical GCNs were proposed in [6] for EEG emotion classification, which adaptively adjust edge weights during model training. A similar approach is taken in [7] where a spatial temporal GCN is used to learn important edges and eventually estimate a latent graph structure. In [8], layers of GCN are used to extract features from temporal portions of EEG signals. Thereafter, long short-term memory (LSTM) is used to learn temporal changes across time slices.

As opposed to GAT process described in equations (1) to (4), GCN's convolution operation results in (6) below, where $c_{ij} = \sqrt{|\mathcal{N}(i)||\mathcal{N}(j)|}$. Clearly, c_{ij} is a function of the structural configuration of the graph. Thus, owing to their dependence on graph structure, GCN based methods suffer from limited generalisability.

$$h_i^{(l+1)} = \sigma(\sum_{j\in\mathcal{N}(i)} \frac{1}{c_{ij}} z_j^{(l)}) \tag{6}$$

On the other hand, GAT replaces c_{ij} with attention mechanism. This ensures that different edge weights are implicitly learned, which if applied to an EEG graph model would offer a good approximation of the functional brain connectome. Besides, the learned edge weights can be visualized for model interpretation. While interactivity between nodes is necessary, it is not sufficient for EEG signal understanding. In EEG signal localisation and lateralisation, for example, it is essential to isolate relevant nodes associated with a particular brain activation. Identifying a time window within which an event related potential (ERP) occurs is also relevant e.g., when measuring signal propagation speed or the time it takes for a brain to react to a painful stimulus.

We propose a GAT Inspired Spatial Temporal model which learns relationship between brain areas (edge attention), detects significant channels (node attention) and identifies relevant temporal segments (temporal attention) responsible for the ERP under study. This architecture allows for interpretability and explainability of brain activities following an ERP. Moreover, node attention can also be used for channel selection for production affordable portable devices for specific applications, e.g., seizure detection, neural marketing, emotion recognition etc.

3 GIST Network Architecture

The GIST network architecture, shown in Figure 1, takes its input as windowed signals and outputs predicted labels. Before the classifier, there are three attention blocks namely: edge attention, node attention and temporal attention. This section discusses these building blocks.

3.1 Input Segmentation

The input EEG signals are segmented into a fixed number of time windows, Q. Slicing signals enables learning from the temporal dynamics of the recorded signal and apportioning importance values to each slice which facilitates identification of a window within which relevant ERPs occurred. One graph is created from each time segment. Thus, there are Q graphs per recording.

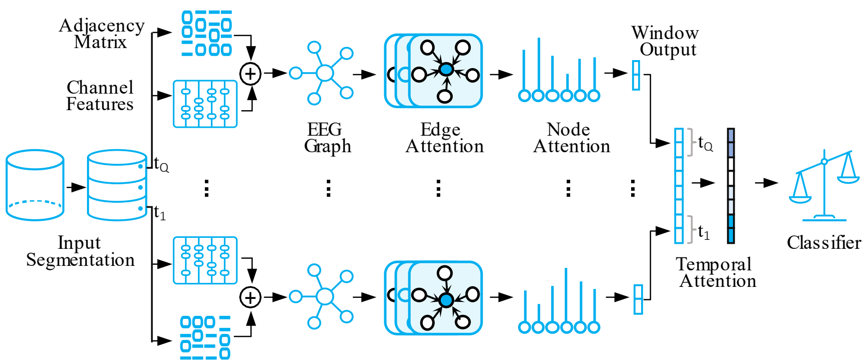

Fig. 1. GIST Network Architecture

3.2 Graph Representation

To formulate a graph, each electrode position (channel) in a temporal slice becomes a node. A set of F features is generated from the raw signal at each of the N nodes. Connectivity between nodes is encoded using a so-called adjacency matrix, A, such that a 1 indicates that an edge exists between nodes i and j. Otherwise, a 0 is inserted.

Adjacency Matrix. For the human brain, it is believed that measures of temporal and/or spectral oscillatory synchrony of recorded EEG signals define the functional connectivity between its regions. Common measures include coherence, transfer entropy, phase locking value, phase-slope index, and Granger causality [9]. Of these methods, coherence is the most popular because it is easy to interpret [10]. It reveals the magnitude of oscillatory frequency coupling between signals. To produce an adjacency matrix from coherence scores between signals in a time window, a threshold (k) value was used to determine if an edge existed between nodes of concerned signals.

Features. Node features extracted from raw EEG signals ranged from conventional statistical features (mean, kurtosis etc.) to nonlinear and nonstationary measures of entropy and fractal dimension. Details of these are available in [11].

3.3 Edge Attention

For edge attention, multi-layer GAT network was used to learn the interdependence between nodes. As it has been highlighted in section 2.3, in a GAT network, attention is defined as the importance of adjacent nodes to a central node - in a way, quantifying the strength of the edges between nodes.

It must be noted that at any given point, there are multiple electrical activities taking place in the human brain. Thus, the adjacency matrix cannot distinguish between brain connectivity related to an ERP and that due to other background activities. The role of this layer is, therefore, to tune edge weights in response to a target ERP. As the model learns to classify whether a desired event took place or not, edge attention weights are automatically adjusted accordingly. Given that in the architecture, we have chopped our signal into Q time windows, the multi-head node update function in (5) can be modified to factor in this temporal slice (t) element as shown in (7),

$$h_i^{\{t,(l+1)\}} = \sigma\left(\frac{1}{H}\sum_{k=1}^{H}\sum_{j\in\mathcal{N}(i)} \alpha_{ijk}^{(t,l)} z_j^{(t,l)}\right) \quad (7)$$

where $h_i^{\{t,(l+1)\}}$ is the output of ith node for tth time window, whose value depends on $z_j^{(t,l)}$, the output of its connected node (with index j) at layer l and tth time window, as well as the corresponding edge attentions $\alpha_{ijk}^{(t,l)}$ for kth attention head. σ is an activation function usually a LeakyReLU.

3.4 Node attention

The output of edge attention layer at temporal window, t is an updated set of node embeddings $h^{\{t,(l+1)\}} \in R^{N \times F'}$, where the number of features, F' do not necessarily have to be the same as the original feature size, F. This becomes an input to the node attention layer within the same time slice, t. The purpose of node attention is to identify nodes whose electrical activities can be associated with the presence of a particular external stimulation. The node embedding vector is transposed to get an $F' \times N$ matrix, which is multiplied with a learned attention weight, $a_{node_i}^{(t)}$ as depicted in (8). After applying an activation function, the importance value, ${d_i}^{(t)}$ for node i in window t, is determined by taking the mean across the F' dimension. Again, to make these attention values comparable, we apply a softmax function (9) to normalize the values. The normalised attention is then used as weights for the node embeddings to produce the node attention block output ${h_i}'^{(t)}$ (10). For N nodes, the output is then $h'^{(t)} \in R^{N \times F'}$.

$$ {d_i}^{(t)} = \frac{1}{F'}\left\{LeakyReLU(a_{node_i}^{(t)} {h_i}^{\{t,(l+1)\}^T})\right\} \tag{8} $$

$$ \alpha_{node_i}^{(t)} = \frac{\exp({d_i}^{(t)})}{\sum_{i \in N^{(t)}} \exp({d_i}^{(t)})} \tag{9} $$

$$ {h_i}'^{(t)} = \alpha_{node_i}^{(t)} {h_i}^{\{t,(l+1)\}} \tag{10} $$

3.5 Temporal attention

The temporal attention block attempts to focus on certain time windows depending on their relevance to the end classification task. It also follows the pattern of additive attention as described above. Equations (11) – (15) specify how the final features of the GIST network are obtained. In this case, $a_{temp}^{(t)}$ is a learnable weight while $b^{(t)}$ and $\alpha_{temp}^{(t)}$ represent temporal slice importance and normalised attention score respectively.

$$ b^{(t)} = \text{LeakyReLU}(a_{temp}^{(t)} h'^{(t)}) \tag{11} $$

$$ \alpha_{temp}^{(t)} = \frac{\exp(b^{(t)})}{\sum_{t \in Q} \exp(b^{(t)})} \tag{12} $$

$$ o^{(t)} = \alpha_{temp}^{(t)} h'^{(t)} \tag{13} $$

$$ \boldsymbol{out} = \Vert_{t=1}^{Q}\, o^{(t)} \tag{14} $$

$$ \tilde{y} = MLP(\boldsymbol{out}) \tag{15} $$

For Q time windows, the operator $\Vert_{t=1}^{Q}\, o^{(t)}$ is used to chain together Q slice outputs to feed into the classification block. Here, $\Vert_{t=1}^{Q} o^{(t)} = (o^{(1)} \parallel o^{(2)} \parallel \ldots \parallel \mathrm{o}^{(Q)})$.

3.6 Classifier

The classification block comprises of a multilayer perceptron (MLP), or a fully connected feed forward neural network. This takes a flattened output of (14) to produce class predictions. In our experiments we empirically opted for an MLP comprising of 3 dense layers interleaved with dropout layers for regularisation and ReLU for activation.

4 Experiments

The main objective behind the GIST network is to model EEG signals in a way that facilitates understanding and interpretability. To this end three datasets were used: Transcranial Electrical Stimulation (TES) [12], Neonatal Pain (NP) and a Database for Emotion Recognition through EEG and ECG Signals (DREAMER) [13].

The TES dataset has known node positions which were stimulated and hence used to demonstrate the roles of edge and node attention blocks. NP signals were time locked to an ERP and hence, this is used to explore the capability of temporal attention in the network. We further investigate the usefulness of the model by applying it to two real world problems: brain regions associated with physical pain (NP), and emotion classification (DREAMER).

Thus, TES data was used to predict which part of the brain (frontal or motor region) was stimulated, while in NP the task was to classify if a given EEG record was done during painful heel lance or not. Finally, in DREAMER, the goal was to distinguish between emotions experienced by subjects while watching video clips on a binary scale of positive or negative valence. In TES and NP, model performance was evaluated by group stratified K-Fold validation where subjects formed the groups. On the other hand, leave one subject out (LOSO) cross validation was used in DREAMER. A summary of these datasets and time windows is shown in Table 1.

Table 1: Summary of datasets used to test GIST network.

Database	No. Subjects	Windows	No. Channels	Sampling Rate	Stimuli
TES	20 (7 F)	5	30	2000 Hz	30Hz 1mA current
NP	112 (52 F)	4	20	2000 Hz	Heel lance
DREAMER	23 (9 F)	12	14	128 Hz	Video clips

4.1 Experimental Settings

DGLGraph [14] package with PyTorch backend was used to create the models. The hyperparameter settings for our experiments were: GAT output feature size, 8; GAT hidden layer feature size, 8; number of GAT layers, 3; number of attention heads per layer, 3; attention drop rate, 0.1; LeakyReLU negative slope, 0.1; and 0.4 as the drop rate for the first two MLP layers. The learning rate was 0.001 using Adam optimiser with cross entropy as the loss function. A threshold, k = 0.6, was used to produce an adjacency matrix from coherence scores. These were chosen following a series of prior Bayesian optimisation experiments on subsets of the datasets. Bayesian optimisation is a relatively quick probabilistic method of progressively narrowing down hyperparameter choices based on previous evaluations.

It was observed that the most influential parameters were GAT output feature size (GOFS), GAT hidden layer feature size (GHLFS) and threshold (k). For GHLFS with possible values as 2,4 and 8, an increase in the value produced a corresponding improvement in the accuracy. On the other hand, with GOFS, lower values were better. An exploration of the threshold value revealed that accuracy was low when k was either too low (highly dense matrix) or too high (highly sparse matrix).

5 Results and Discussion

5.1 Feature Learning

A dimensionality reduction strategy called Uniform Manifold Approximation and Projection (UMAP) [15] was used to visualise how well feature learning took place across the GIST model.

Figure 2 shows a projection of the low-level input features and outputs of the three attention blocks onto two dimensional spaces. The plots demonstrate that there is increasing separability between classes moving across the edge attention, node attention and temporal attention blocks. Thus, a trained GIST model is good at feature transformation for classification purposes. It is also worth noting that for a simpler problem such as TES, it is possible to separate the classes at the node attention level which enables node attention to easily identify the active electrode positions.

5.2 The Role of Attention Blocks

From the temporal point of view, the datasets used are of three different characteristics: EEG recordings in which the time at which an ERP is induced is unknown e.g.,

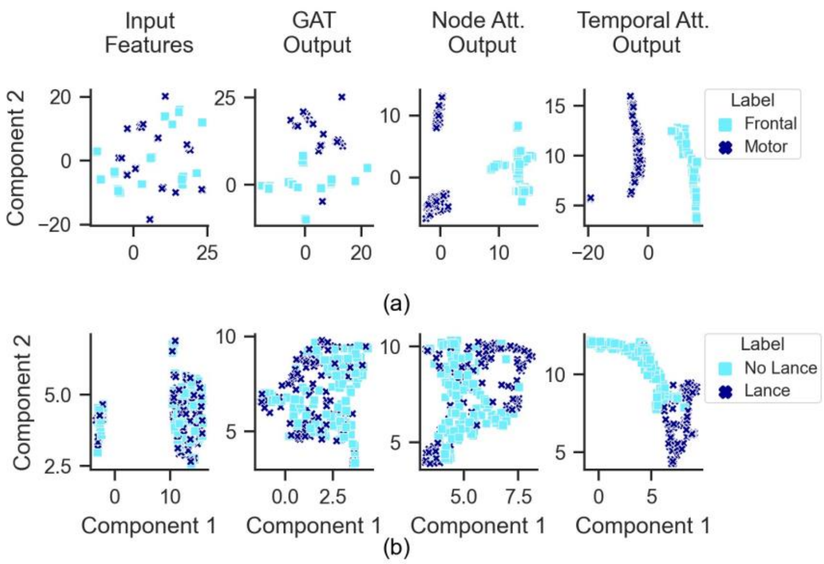

Fig. 2. UMAP visualisation of feature learning across the GIST network input, edge attention, node attention and temporal attention layers for (a) TES and (b) NP.

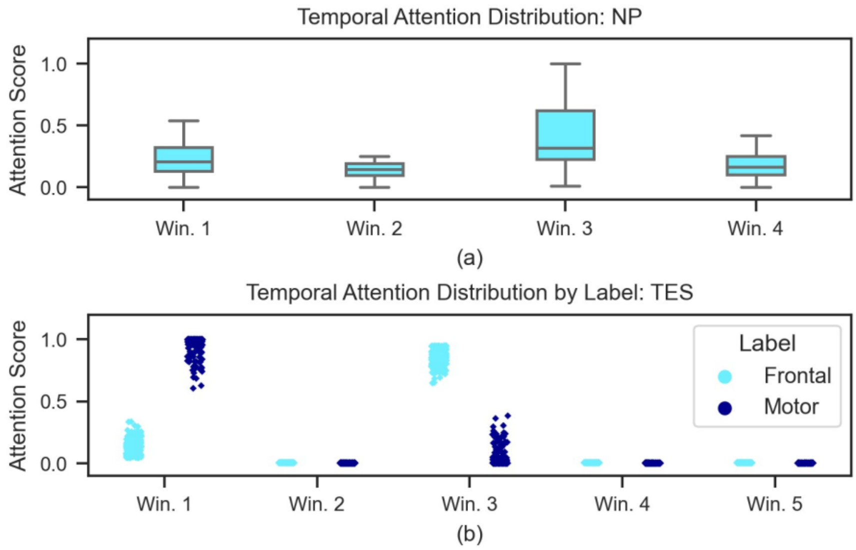

Fig. 3. (a) Temporal attention distribution box plots in TES and (b) Strip plots showing temporal attention by label in TES.

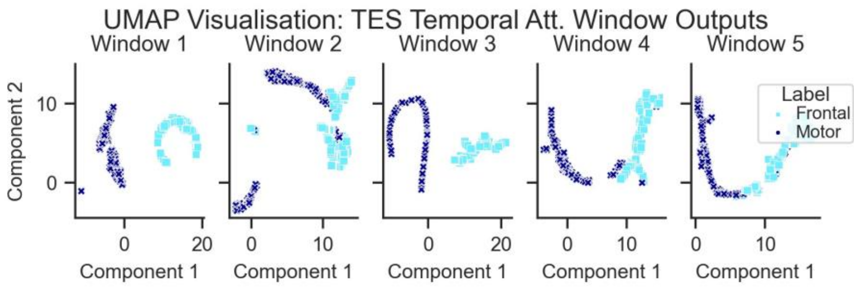

Fig. 4. UMAP visualisation of TES of the temporal attention block output of each individual window

DREAMER where relevant parts of the clips at which emotions were evoked are not identified; synchronized data where the stimulation time point is fixed such as NP in which heel lance was done 2 seconds after the start of the recording; and a dataset like TES where a stimulus was applied in the entire duration of the recording.

From figure 3(a) we observe that highest temporal attention scores are in time windows 3. This is in line with what was expected since window 3 comes immediately after a heel lance and hence it being the most informative in as far as classification between lance and no lance is concerned. By extension, this also demonstrates that the brain's response to a noxious stimulus is within 1 second. Figure 3(b) displays strip plots of TES temporal attention by label. It can be observed that both windows 1 and 3 can distinguish between frontal and motor stimulation. This is also supported by a temporal output visualisation of the 5 windows in figure 4 where windows 1 and 3 show data points in nearly perfect clusters.

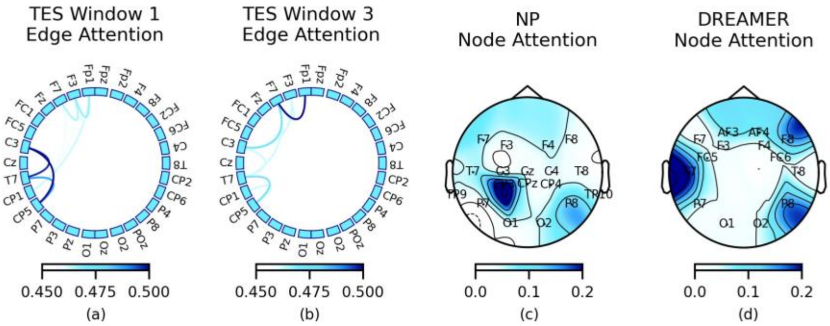

Fig. 5. Connectivity plots for (a) TES window 1 edge attention and (b) TES window 3 edge attention, and topographic plots for (c) NP node attention (d) DREAMER node attention

Moreover, if we consider the edge attention at these two windows as depicted in figures 5(a) and (b), it is evident that the two windows attend to different brain regions. Window 1 focusses on the sensorimotor region while window 2 concentrates on the frontal area. Thus, for this data type, where the entire duration of EEG recording comprised of ERP, the time segments act in a manner similar to multi-head attention layers.

The node attention distribution information conveyed through topographic plots in 5(c) and (d) can be interpreted as either highlighting relevant electrode positions for classification or associating concerned brain areas with certain functions e.g. pain, 5(c) and emotion processing, 5(d). From 5(c) the highest intensity is around the premotor and sensorimotor region, followed by the parietal region. This observation is supported by Tayeb et al. who established that 'noxious stimulation activates the pre-motor (Cz electrode) and moderately intense stimulation was found in the parietal lobe (P2, P4, and P6 electrodes)'[16]. 5(d) suggests that the frontal, temporal, and parietal regions are active during emotion processing. This is a view also shared by [17–19] among others. It follows therefore, that node attention in figures 5(d) can potentially be used to select channels for emotion recognition. However, applying this model for channel selection purposes must be done with extreme caution. This is because information contained in a node at the node attention layer is an aggregation of feature data from neighbouring nodes. The number of message passing rounds (how far wide the node goes to fetch for information) is controlled by the number of GAT layers.

5.3 Classification Performance

The model's performance was assessed using accuracy, sensitivity, and specificity. Accuracy refers to the ratio of the number of correct predictions to the total number of predictions. Sensitivity, also called recall, hit rate, or true positive rate is calculated as the number of instances which were predicted to be positive which are truly positive divided by the number of all positive instances available in the dataset. The equivalent of sensitivity for the negative class is specificity (selectivity or true negative rate). Thus, it is the ratio of the number of true negatives to the sum of the true negatives and false positives.

Classification scores for TES, NP and DREAMER are provided in table 2 below. The performance of the model on TES was very good, with scores of 99.4%, 100% and 98.8% for accuracy, specificity, and sensitivity respectively. On the other hand, specificity for NP was 67% even though accuracy and sensitivity were relatively higher at 76% and 96% in that order.

It is only DREAMER which has been found to have been used in other studies, albeit with different cross validation strategies. Table 3 compares performance of GIST network with other models. The within subject (WS) protocol utilised subject dependent leave one session out cross validation strategy in which a subject's recordings are split such that one session is used for testing while the rest are used for training. Thus, an accuracy score of 79.18% obtained using GIST network compares fairly with other models even though LOSO, a more challenging cross validation strategy was used.

Table 2: Model classification performance

Database	Accuracy	Specificity	Sensitivity
NP	76%	67%	96%
TES	99.4%	100%	98.8%
DREAMER	79.18%	75.4%	80.2%

Table 3: Performance on DREAMER

Model	Validation	Accuracy
daSPDnet [20]	LOSO	67.99%
CNN [21]	LOSO	75.93%
DGCNN [6]	WS	86.23%
GCB-Net [22]	WS	86.99%
GIST Network	**LOSO**	**79.18%**

6 Conclusion and Future Work

In this paper, a GIST network has been presented. The graph model thrives on the principle of attention to create a rich feature representation for classification purposes. The additive multi-level attention mechanisms used in the network facilitates understanding of EEG signals. From the attention scores, it is possible to visualise interdependence between brain regions, importance of individual electrode positions and the significance of temporal slices.

Future work could include adding an automatic feature extraction block to the GIST model. Modification of the architecture to accept heterogeneous graphs could also make the model accommodate multimodal data. To improve the model's channels selection capability, a dedicated channel attention layer could be added to the architecture just before the edge attention layer.

References

1. Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., Vandergheynst, P.: Geometric Deep Learning: Going beyond Euclidean data. IEEE Signal Process. Mag. 34, 18–42 (2017). https://doi.org/10.1109/MSP.2017.2693418.

2. Zhou, J., Cui, G., Hu, S., Zhang, Z., Yang, C., Liu, Z., Wang, L., Li, C., Sun, M.: Graph neural networks: A review of methods and applications. AI Open. 1, 57–81 (2020). https://doi.org/https://doi.org/10.1016/j.aiopen.2021.01.001.
3. Kipf, T.N., Welling, M.: Semi-supervised classification with graph convolutional networks. arXiv Prepr. arXiv1609.02907. (2016).
4. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Lio, P., Bengio, Y.: Graph attention networks. arXiv Prepr. arXiv1710.10903. (2017).
5. Zhang, D., Yao, L., Chen, K., Wang, S., Haghighi, P.D., Sullivan, C.: A Graph-Based Hierarchical Attention Model for Movement Intention Detection from EEG Signals. IEEE Trans. Neural Syst. Rehabil. Eng. 27, 2247–2253 (2019). https://doi.org/10.1109/TNSRE.2019.2943362.
6. Song, T., Zheng, W., Song, P., Cui, Z.: EEG Emotion Recognition Using Dynamical Graph Convolutional Neural Networks. IEEE Trans. Affect. Comput. 11, 532–541 (2020). https://doi.org/10.1109/TAFFC.2018.2817622.
7. Li, X., Qian, B., Wei, J., Li, A., Liu, X., Zheng, Q.: Classify EEG and Reveal Latent Graph Structure with Spatio-Temporal Graph Convolutional Neural Network. In: 2019 IEEE International Conference on Data Mining (ICDM). pp. 389–398 (2019). https://doi.org/10.1109/ICDM.2019.00049.
8. Yin, Y., Zheng, X., Hu, B., Zhang, Y., Cui, X.: EEG emotion recognition using fusion model of graph convolutional neural networks and LSTM. Appl. Soft Comput. 100, 106954 (2021). https://doi.org/10.1016/j.asoc.2020.106954.
9. Cao, J., Zhao, Y., Shan, X., Wei, H.-L., Guo, Y., Chen, L., Erkoyuncu, J.A., Sarrigiannis, P.G.: Brain functional and effective connectivity based on electroencephalography recordings: A review. Hum. Brain Mapp. 43, 860–879 (2022). https://doi.org/10.1002/hbm.25683.
10. Rocca, D. La, Campisi, P., Vegso, B., Cserti, P., Kozmann, G., Babiloni, F., Fallani, F.D.V.: Human Brain Distinctiveness Based on EEG Spectral Coherence Connectivity. IEEE Trans. Biomed. Eng. 61, 2406–2412 (2014). https://doi.org/10.1109/TBME.2014.2317881.
11. Msonda, J.R., He, Z., Lu, C.: Feature Reconstruction Based Channel Selection for Emotion Recognition Using EEG. In: 2021 IEEE Signal Processing in Medicine and Biology Symposium (SPMB). pp. 1–7 (2021). https://doi.org/10.1109/SPMB52430.2021.9672258.
12. Gebodh, N., Esmaeilpour, Z., Datta, A., Bikson, M.: Dataset of concurrent EEG, ECG, and behavior with multiple doses of transcranial electrical stimulation. Sci. Data. 8, 274 (2021). https://doi.org/10.1038/s41597-021-01046-y.
13. Katsigiannis, S., Ramzan, N.: DREAMER: A Database for Emotion Recognition Through EEG and ECG Signals from Wireless Low-cost Off-the-Shelf Devices. IEEE J. Biomed. Heal. Informatics. 22, 98–107 (2018). https://doi.org/10.1109/JBHI.2017.2688239.
14. Wang, M., Zheng, D., Ye, Z., Gan, Q., Li, M., Song, X., Zhou, J., Ma, C., Yu, L., Gai, Y., Xiao, T., He, T., Karypis, G., Li, J., Zhang, Z.: Deep Graph Library: A Graph-Centric, Highly-Performant Package for Graph Neural Networks, https://arxiv.org/abs/1909.01315, (2019). https://doi.org/10.48550/ARXIV.1909.01315.
15. McInnes, L., Healy, J., Melville, J.: UMAP: Uniform Manifold Approximation and Projection for Dimension Reduction, https://arxiv.org/abs/1802.03426, (2018). https://doi.org/10.48550/ARXIV.1802.03426.
16. Tayeb, Z., Bose, R., Dragomir, A., Osborn, L.E., Thakor, N. V., Cheng, G.: Decoding of Pain Perception using EEG Signals for a Real-Time Reflex System in Prostheses: A Case Study. Sci. Rep. 10, 4–8 (2020). https://doi.org/10.1038/s41598-020-62525-7.

17. Phillips, M.L., Drevets, W.C., Rauch, S.L., Lane, R.: Neurobiology of emotion perception I: The neural basis of normal emotion perception. Biol. Psychiatry. 54, 504–514 (2003). https://doi.org/10.1016/S0006-3223(03)00168-9.
18. Zheng, W.L., Lu, B.L.: Investigating Critical Frequency Bands and Channels for EEG-Based Emotion Recognition with Deep Neural Networks. IEEE Trans. Auton. Ment. Dev. 7, 162–175 (2015). https://doi.org/10.1109/TAMD.2015.2431497.
19. Zheng, W.: Multichannel EEG-Based Emotion Recognition via Group Sparse Canonical Correlation Analysis. IEEE Trans. Cogn. Dev. Syst. 9, 281–290 (2017). https://doi.org/10.1109/TCDS.2016.2587290.
20. Wang, Y., Qiu, S., Ma, X., He, H.: A prototype-based SPD matrix network for domain adaptation EEG emotion recognition. Pattern Recognit. 110, 107626 (2021). https://doi.org/https://doi.org/10.1016/j.patcog.2020.107626.
21. Pandey, P., Seeja, K.R.: A One-Dimensional CNN Model for Subject Independent Emotion Recognition Using EEG Signals. In: Khanna, A., Gupta, D., Bhattacharyya, S., Hassanien, A.E., Anand, S., and Jaiswal, A. (eds.) International Conference on Innovative Computing and Communications. pp. 509–515. Springer Singapore, Singapore (2022).
22. Zhang, T., Wang, X., Xu, X., Chen, C.L.P.: GCB-Net: Graph Convolutional Broad Network and Its Application in Emotion Recognition. IEEE Trans. Affect. Comput. 13, 379–388 (2022). https://doi.org/10.1109/TAFFC.2019.2937768.