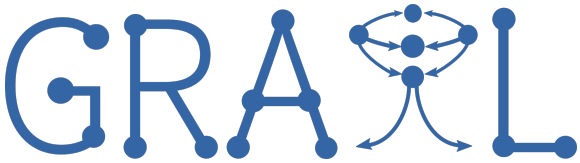
Proceedings of the 2nd Workshop on Graphs in Biomedical Image Analysis
MICCAI-GRAIL 2018

Enzo Ferrante[1], Sarah Parisot[2], Aristeidis Sotiras[3], Bartłomiej Papież[4]

[1] *CONICET / Universidad Nacional del Litoral*
[2] *AimBrain*
[3] *University of Pennsylvania*
[4] *University of Oxford*

20 September, 2018



Editorial

GRAIL 2018 is the 2nd International Workshop on Graphs in Biomedical Image Analysis, organized as a satellite event of the 21st International Conference on Medical Image Computing and Computer Assisted Intervention (MICCAI 2018) in Granada, Spain. After the success and positive feedback obtained last year, this is the second time we bring GRAIL to MICCAI, in the spirit of strengthening the links between graphs and biomedical imaging.

The workshop provides a unique opportunity to meet and discuss both theoretical advances in graphical methods, as well as the practicality of such methods when applied to complex biomedical imaging problems. Simultaneously, the workshop seeks to be an interface to foster future interdisciplinary research including signal processing and machine learning on graphs.

Graphs and related graph-based modelling have attracted significant research focus as they enable to represent complex data and their interactions in a perceptually meaningful way. With the advent of Big Data in the medical imaging community, the relevance of graphs as a means to represent data sampled from irregular and non-Euclidean domains is increasing, together with the development of new inference and learning methods that operate on such structures. There is a wide range of well-established and emerging biomedical imaging problems that can benefit from these advances; we believe that the research presented in this volume constitutes a clear example of that.

The GRAIL 2018 proceedings contain 6 high quality papers of 8 to 11 pages that were pre-selected through a rigorous peer review process. All submissions were peer-reviewed through a double-blind process by at least 2 members of the program committee, comprising 18 experts in the field of graphs in biomedical image analysis. The accepted manuscripts cover a wide set of graph based medical image analysis methods and applications, including neuroimaging and brain connectivity, graph matching algorithms, graphical models for image segmentation, brain modeling through neuronal networks and deep learning models based on graph convolutions. In addition to the papers presented in this LNCS volume, the workshop will comprise short abstracts and two keynote presentations from world renowned experts: Prof. Michael Bronstein and Prof. Dimitri Van De Ville. We hope this event will foster the development of more powerful graph-based models for the analysis of biomedical images.

We wish to thank all the GRAIL 2018 Authors for their participation and the members of the Program Committee for their feedback and commitment to the workshop. We are very grateful to our sponsors Entelai (https://entelai.com/) and the UK EPSRC-funded Medical Image Analysis Network (MedIAN - https://www.median.ac.uk/) for their valuable support.

The proceedings of the workshop are published as a joint LNCS volume alongside other satellite events organized in conjunction with MICCAI. In addition to the LNCS volume, to promote transparency, the papers' reviews and preprints are publicly available on the workshop website (http://grail-miccai.github.io/) together with their corresponding optional response to reviewers. In addition to the papers, abstracts, slides and posters presented during the workshop will be made publicly available on the GRAIL website.

Enzo Ferrante, Sarah Parisot, Aristeidis Sotiras, Bartłomiej Papież

Organization

Organizing Committee

- Enzo Ferrante, CONICET / Universidad Nacional del Litoral, Argentina
- Sarah Parisot, AimBrain, UK
- Aristeidis Sotiras, University of Pennsylvania, USA
- Bartłomiej Papież, University of Oxford, UK

Scientific Committee

- Kayhan Batmanghelich, University of Pittsburgh / Carnegie Mellon University, US
- Eugene Belilovsky, INRIA / KU Leuven, France
- Siddhartha Chandra, CentraleSupelec / INRIA, France
- Xin Chen, The University of Nottingham, UK
- Emilie Chouzenoux, INRIA Saclay, France
- Puneet K. Dokania, Oxford University, UK
- Ben Glocker, Imperial College London, UK
- Ali Gooya, University of Sheffield, UK
- Mattias Heinrich, University of Luebeck, Germany
- Lisa Koch, ETH Zurich, Switzerland
- Evgenios Kornaropoulos, University of Cambridge, UK
- Sofia Ira Ktena, Imperial College London, UK
- Georg Langs, University of Vienna / MIT, Austria, Austria
- Jose Ignacio Orlando, Medical University of Vienna, Austria
- Yusuf Osmanlioglu, University of Pennsylvania, US
- Yangming Ou, Harvard University, US
- Nikos Paragios, CentraleSupelec / INRIA, France
- Sotirios Tsaftaris, University of Edinburgh, UK
- Maria Vakalopoulou, Université Paris-Saclay / INRIA, France
- William Wells III, Harvard Medical School, US

Sponsors

ENTELAI

MedIAN
Medical Image Analysis
Network

Conference Program

Graph Saliency Maps through Spectral Convolutional Networks: Application to Sex Classification with Brain Connectivity

Salim Arslan, Sofia Ira Ktena, Ben Glocker, Daniel Rueckert

Biomedical Image Analysis Group, Department of Computing,
Imperial College London, UK

Abstract. Graph convolutional networks (GCNs) allow to apply traditional convolution operations in non-Euclidean domains, where data are commonly modelled as irregular graphs. Medical imaging and, in particular, neuroscience studies often rely on such graph representations, with brain connectivity networks being a characteristic example, while ultimately seeking the locus of phenotypic or disease-related differences in the brain. These regions of interest (ROIs) are, then, considered to be closely associated with function and/or behaviour. Driven by this, we explore GCNs for the task of ROI identification and propose a visual attribution method based on class activation mapping. By undertaking a sex classification task as proof of concept, we show that this method can be used to identify salient nodes (brain regions) without prior node labels. Based on experiments conducted on neuroimaging data of more than 5000 participants from UK Biobank, we demonstrate the robustness of the proposed method in highlighting reproducible regions across individuals. We further evaluate the neurobiological relevance of the identified regions based on evidence from large-scale UK Biobank studies.

1 Introduction

Graph convolutional neural networks (GCNs) have recently gained a lot of attention, as they allow adapting traditional convolution operations from Euclidean to irregular domains [1]. Irregular graphs are encountered very often in medical imaging and neuroscience studies in the form of brain connectivity networks, supervoxels or meshes. In these cases, applications might entail both node-centric tasks, e.g. node classification, as well as graph-centric tasks, e.g. graph classification or regression. While CNNs have redefined the state-of-the-art in numerous problems by achieving top performance in diverse computer vision and pattern recognition tasks, insights into their underlying decision mechanisms and the impact of the latter on performance are still limited.

Recent works in deep learning address the problem of identifying salient regions in 2D/3D images in order to visualise determinant patterns for classification/regression tasks performed by a CNN and obtain spatial information that might be useful for the delineation of regions of interest (ROI) [2]. In the field

of neuroscience, in particular, the identification of the exact locus of disease- or phenotype-related differences in the brain is commonly sought. Locating brain areas with a critical role in human behaviour and mapping functions to brain regions as well as diseases on disruptions to specific structural connections are among the most important goals in the study of the human connectome.

In this work, we explore GCNs for the task of brain ROI identification. As proof of concept, we undertake a sex classification task on functional connectivity networks, since there is previous evidence for sex-related differences in brain connectivity [3]. Characteristically, stronger functional connectivity was established within the default mode network of female brains, while stronger functional connectivity was found within the sensorimotor and visual cortices of male brains [4]. As a result, we consider this a suitable application to demonstrate the potential of the proposed method for delineating brain regions based on the attention/sensitivity of the model to the sex of the input subject's connectivity graph. More specifically, we show that spatially segregated salient regions can be identified in non-Euclidean space by using class activation mapping [5] on GCNs, making it possible to effectively map the most important brain regions for the task under consideration.

Related work: Graph convolutions have been employed to address both graph-centric and node-centric problems and can be performed in the spatial [6] or spectral domain [7,8]. In the latter case, convolutions correspond to multiplications in the graph spectral domain and localised filters can be obtained with Chebyshev polynomials [7] or rational complex functions [8]. [9] introduced adaptive graph convolutions and attention mechanisms for graph- and node-centric tasks, while in [10] attention mechanisms were employed to assign different weights to neighbours in node classification tasks with inductive and transductive inference. Although the latter works focus the attention of the network onto the most relevant nodes, they overlook the importance/contribution of different features/graph elements for the task at hand.

At the same time, visual feature attribution through CNNs has attracted attention, as it allows identifying salient regions in an input image that lead a classification network to a certain prediction. It is typically addressed with gradient and/or activation-based strategies. The former relies on the gradients of the prediction with respect to the input and attributes saliency to the regions that have the highest impact on the output [2]. Activation-based methods, on the other hand, associate feature maps acquired in the final convolutional layer with particular classes and use weighted activations of the feature maps to identify salient regions [5]. A recent work addresses the problem from an adversarial point of view and proposes a visual attribution technique based on Wasserstein generative adversarial networks [11]. While these methods offer promising results on Euclidean images, their application to graph-structured data is yet to be explored.

Contributions: We propose a visual feature attribution method for graph-structured data by combining spectral convolutional networks and class activation mapping [5]. Through a graph classification task, in which each graph

represents a brain connectivity network, we detect and visualise brain regions that are responsible for the prediction of the classifier, hence providing a new means of brain ROI identification. As a proof of concept, we derive experiments in the context of sex differences in functional connectivity. First, we train a spectral convolutional network classifier and achieve state-of-the-art accuracy in the prediction of female and male subjects based on their functional connectivity networks captured at rest. The activations of the feature maps are, then, used for visual attribution of the nodes, each of which is associated with a brain region. Using resting-state fMRI (rs-fMRI) data of more than 5000 subjects acquired by UK Biobank, we show that the proposed method is highly robust in selecting the same set of brain regions/nodes across subjects and yields highly reproducible results across multiple runs with different seeds.

2 Method

Fig. 1 illustrates the proposed method for identifying brain regions used by GCNs to predict a subject's sex based on its functional connectivity. Given an adjacency matrix that encodes similarities between nodes and a feature matrix representing a node's connectivity profile, the proposed method outputs the sex of the input subject and provides a graph saliency map highlighting the brain regions/nodes that lead to the corresponding prediction. Finally, we rank brain regions with respect to their contribution towards driving the model's prediction at subject level and compute a population-level saliency map by combining them across individuals.

Spectral graph convolutions: We assume n samples (*i.e.* subjects), $X = [X_1, \ldots, X_n]^T$, with signals defined on a graph structure. Each subject is associated with a data matrix $X_i \in \mathbb{R}^{d_x \times d_y}$, where d_y is the dimensionality of the node's feature vector (*i.e.* signal), and a label $y_i \in \{0, 1\}$. In order to encode the structure of the data, we define a weighted graph $G = (V, E, W)$ where V is the set of $d_x = |V|$ nodes (vertices), E is the set of edges (connections) and $W \in \mathbb{R}^{d_x \times d_x}$ is the weighted adjacency matrix, representing the weight of each edge, *i.e.* $W_{i,j}$ is the weight of the edge connecting $v_i \in V$ to $v_j \in V$.

A convolution in the graph spatial domain corresponds to a multiplication in the graph spectral domain. Hence, graph filtering operations can be performed in the spectral domain using the eigenfunctions of the normalised Laplacian of a graph [12], which is defined as $L = I_{d_x} - D^{-\frac{1}{2}} W D^{-\frac{1}{2}}$, where D is the degree matrix and I_{d_x} the identity matrix. In order to yield filters that are strictly localised and efficiently computed, Defferrard et al. [7] suggested a polynomial parametrisation on the Laplacian matrix by means of Chebyshev polynomials. Chebyshev polynomials are recursively computed using $T_k(L) = 2LT_{k-1}(L) - T_{k-2}$, with $T_0(L) = 1$ and $T_1(L) = L$.

A polynomial of order K yields strictly K-localised filters. Filtering of a signal x with a K-localised filter can, then, be performed using:

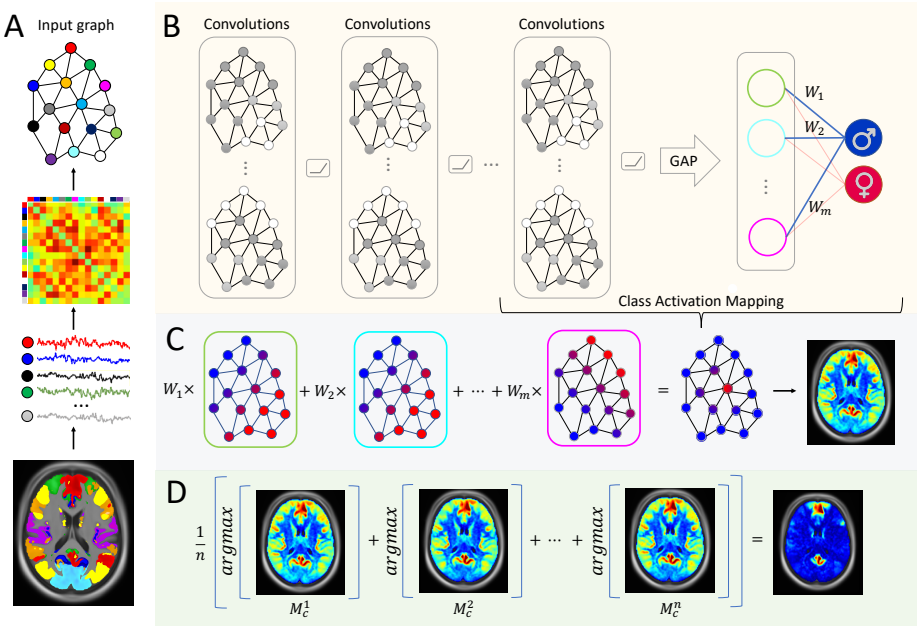


Fig. 1: Overview of the proposed approach. (A) The input graph is computed using a brain parcellation and rs-fMRI connectivity signals. (B) Graph convolutional network model. Convolutional feature maps of the last layer are spatially pooled via global average pooling (GAP) and connected to a linear sex classifier. (C) Class activation mapping procedure. (D) Generation of population-level saliency maps.

$$y = g_\theta(L) * x = \sum_{k=0}^{K} \theta_k T_k(\tilde{L}) x, \tag{1}$$

with $\tilde{L} = \frac{2}{\lambda_{max}} L - I_{d_x}$ and λ_{max} denoting the largest eigenvalue of the normalised Laplacian, L. The output of the l^{th} layer for a sample s in a graph convolutional network is, then, given by:

$$y_s^l = \sum_{i=1}^{F_{in}} g_{\theta_i^l}(L) x_{s,i}^l. \tag{2}$$

For F_{out} output filter banks and F_{in} input filter banks, this yields $F_{in} \times F_{out}$ vectors of trainable Chebyshev coefficients $\theta_i^l \in \mathbb{R}^K$ with $x_{s,i}^l$ denoting the input feature map i for sample s at layer l. Hence, at each layer the total number of trainable parameters is $F_{in} \times F_{out} \times K$.

Class activation mapping: Class activation mapping (CAM) [5] is a technique used to identify salient regions that assist a CNN to predict a particular class. It builds on the fact that, even though no supervision is provided on the

4

object locations, feature maps in various layers of CNNs still provide reliable localisation information [13], which can be captured via global average pooling (GAP) in the final convolutional layer [14]. Encoded into a class activation map, these "spatially-averaged" deep features not only yield approximate locations of objects, but also provide information about where the attention of the model focuses when predicting a particular class [5]. In the context of GCNs, CAM is used to localise discriminative nodes, each associated with a saliency score.

The process for generating class activation maps is illustrated in Fig 1. Given a typical GCN model which consists of a series of convolutional layers, a GAP layer is inserted into the network right after the last convolutional layer. The spatially-pooled feature maps are connected to a dense layer that produces the output for a classification task (Fig. 1B). We can then linearly map the weights of the dense layer onto the corresponding feature maps to generate a class activation map showing the salient nodes in the graph (Fig. 1C).

More formally, let $f_i(v)$ represent the activation of the ith feature map in the last convolutional layer at node v. For the feature map i, the average pooling operation is defined as $F_i = (1/d_z) \sum_v f_i(v)$, where $F_i \in \mathbb{R}$ and d_z is the number of nodes in the feature map. Thus, for a given class c, the input to the dense layer is $\sum_i w_i^c F_i$, where w_i^c is the corresponding weight of F_i for class c. Intuitively, w_i^c indicates the importance of F_i for class c, therefore, we can use these weights to compute a class activation map M_c, where each node is represented by a weighted linear sum of activations, *i.e.* $M_c(v) = \sum_i w_i^c f_i(v)$. This map shows the impact of a node v to the prediction made by the GCN model and, once projected back onto the brain, can be used to identify the ROIs that are most relevant for the specific classification task.

Population-level saliency maps: Although CAM provides graph-based activation maps at subject/class-level, population-level statistics about discriminative brain regions are also important. In order to combine class activation maps across subjects, we define a simple *argmax* operation that, for each subject, returns the index of the k top nodes with the highest activation. These are, subsequently, averaged across subjects and referenced as the population-level saliency maps as illustrated in Fig 1D.

Network architecture and training: The details of the GCN architecture are presented in Table 1 and summarised as follows. 5 convolutional layers, each succeeded by rectified linear (ReLU) non-linearity are used. No pooling is performed between consecutive layers, as empirical results suggest that reducing the resolution of the underlying graph does not improve performance. We apply zero-padding to keep the spatial resolution of the feature maps unchanged throughout the model. A dropout rate of 0.5 is used in the 2^{nd}, 4^{th}, and 5^{th} layers. The feature maps of the last layer are spatially averaged and connected to a linear classifier with softmax output. We employ global average pooling as it reflects where the attention of the network is focused and substantially reduces the number of parameters, hence alleviating over-fitting issues [14].

5

Table 1: Network architecture of the proposed model. * indicates the use of dropout for the corresponding convolutional layer.

Layer	Input	Conv	Conv*	Conv	Conv*	Conv*	GAP	Linear
Channels	55	32	32	64	64	128	128	2
***K*-order**	N/A	9	9	9	9	9	N/A	N/A
Stride	N/A	1	1	1	1	1	N/A	N/A

The loss function used to train the model comprises a cross entropy term and an L_2 regularisation term with decay rate of $5e^{-4}$. We use an Adam optimiser with momentum parameters $\beta = [0.9, 0.999]$ and initialise the training with a learning rate of 0.001. Training is performed for a fixed number of 500 steps (*i.e.* 20 epochs), in mini-batches of 200 samples, equally representing each class. We evaluate the model every 10 steps with an independent validation set, which is also used to monitor training. Based on this, the learning rate is decayed by a factor of 0.5, whenever validation accuracy drops in two consecutive evaluation rounds.

3 Data and Experiments

Dataset and preprocessing: Imaging data is collected as part of UK Biobank's health imaging study (http://www.ukbiobank.ac.uk/), which aims to acquire imaging data for 100,000 predominantly healthy subjects. The multimodal scans together with the vast amount of non-imaging data are publicly available to assist researchers investigating a wide range of diseases, such as dementia, arthritis, cancer, and stroke. We conduct our experiments on rs-fMRI images available for 5430 subjects from the initial data release. Non-imaging data and medical information are also provided alongside brain scans including sex, age, genetic data, and many others. The dataset used here consists of 2873 female (aged 40-70 yo, mean 55.38 ± 7.41) and 2557 male (aged 40-70 yo, mean 56.61 ± 7.60) subjects.

Details of data acquisition and preprocessing procedures are given in [15]. Standard preprocessing steps have been applied to rs-fMRI images including motion correction, high-pass temporal filtering, and gradient distortion correction. An independent component analysis (ICA)-based approach is used to identify and remove structural artefacts [15]. Finally, images go through visual quality control and any preprocessing failures are eliminated.

Brain parcellation and network modelling: A dimensionality reduction procedure known as "group-PCA" [16] is applied to the preprocessed data to obtain a group-average representation. This is fed to group-ICA [17] to parcellate the brain into 100 spatially independent, non-contiguous components. Group-ICA components identified as artefactual (*i.e.* not neuronally-driven) are discarded and the remaining $d = 55$ components are used to estimate a functional connectivity network for each subject by means of L_2-regularised partial

correlation between the ICA components' representative timeseries [18] Each connectivity network corresponds to the data matrix X_i, i.e. $d_x = d_y$ in our application, and their average across training subjects is used to define the weighted graph W, in which only the $k = 10$ nearest neighbours are retained for each node, so that the local connectivity structure in the graph is effectively represented.

Experimental setup: We use stratified 10-fold cross-validation to evaluate the model with split ratios set to 0.8, 0.1, and 0.1 for training, validation, and testing, respectively. Cross-validation allows to use all subjects for both training/validation and testing, while each subject in the dataset is used for testing exactly once. To further evaluate how the performance varies across different sets of subjects and how robust the identified salient regions are, we repeat cross-validation 10 times with different seeds.

4 Results and Discussion

In Table 2 we provide classification results obtained with the GCN classifier. The presented accuracy rates correspond to the results of all 10 folds for each run. On average, we achieve a test accuracy of 88.06% across all runs/folds, with low standard deviation for each run, indicating reproducible classification performance. While classification is not the end goal of the proposed method, a high accuracy rate is a prerequisite for robust and reliable activation maps. Yet, the average performance of our classifier is slightly higher than the state-of-the-art sex classification accuracy with respect to functional connectivity [19,20].

Fig. 2 shows the sex-specific activations for all nodes. As illustrated, the GCN focuses on the same regions for both classes, with one class (*i.e.* female) consistently yielding higher activation than the other (*i.e.* male). This can be attributed to the fact that a binary classifier only needs to predict one class, while every other sample is automatically assigned the remaining class label. The most important nodes, in descending order, are 21, 5, 13, and 7. As indicated by the size of their markers, these four nodes are almost always ranked within the top $k = 3$ of all nodes with respect to their activations, meaning that all subjects but few are consistently classified according to the connections of these nodes. While we only provide results for $k = 3$, the same regions are identified for lower/higher values of k, with only minimal changes in their occurrence rate, as shown in Supplementary Fig. 1.

In order to explore the neurobiological relevance of these results, we refer to the UK Biobank group-averaged functional connectome [15], which maps

Table 2: Average sex classification accuracy rates (in %) for each run.

Run	1	2	3	4	5	6	7	8	9	10	Avr
Acc	88.51	88.27	87.64	87.94	87.84	88.01	88.51	88.08	88.05	87.77	**88.06**
Std	1.57	1.25	1.88	1.30	1.73	1.66	1.54	1.34	1.93	1.06	**1.57**

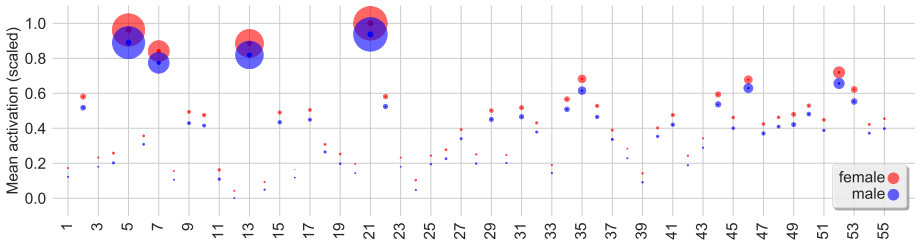


Fig. 2: Sex-specific class activations for all nodes averaged across subjects and runs. Mean activations are scaled to $[0, 1]$ for better visualisation. The size of the markers indicates the number of times a node is ranked within the top $k = 3$ most important, summed across subjects and runs.

the functional interactions between the 55 brain regions clustered into six resting state networks (RSNs) according to their average population connectivity (Fig. 3). RSNs comprise spatially segregated, but functionally connected cortical regions, that are associated with diverse functions, such as sensory/motor, visual processing, auditory processing, and memory. Our comparisons to the connectome revealed that regions 21, 5, 13, and 7 (as shown in Fig. 3) are part of the default mode network (highlighted with red), a spatially distributed cluster which is activated 'by default' during rest. A large-scale study on sex differences in the human brain [4] has also found evidence that functional connectivity is stronger for females in the default mode network, which might further indicate that the identified regions are neurobiologically relevant and reflect sex-specific characteristics encoded in functional connectivity.

5 Conclusion

In this paper, we have addressed the visual attribution problem in graph-structured data and proposed an activation-based approach to identify salient graph nodes using spectral convolutional neural networks. By undertaking a graph-centric classification task, we showed that a GCN model enhanced with class activation mapping can be used to identify graph nodes (brain regions), even in the absence of supervision/labels at the node level. Based on experiments conducted on neuroimaging data from UK Biobank, we demonstrated the robustness of the proposed method by means of highlighting the same regions across different subjects/runs using cross validation. We further validated the neurobiological relevance of the identified ROIs based on evidence from UK Biobank studies [15,4].

While the potential of the proposed method is demonstrated on functional networks with rs-fMRI, it can be applied to any graph-structured data and/or modality. However, its applicability might be limited by several factors, including the definition and number of nodes (*e.g.* brain parcellation), network modelling, as well as node signal choices. It is also important to assess the robustness of the identified regions by disentangling the effect of the graph structure and the node features. While the method can successfully localise the salient regions, its lack of

8

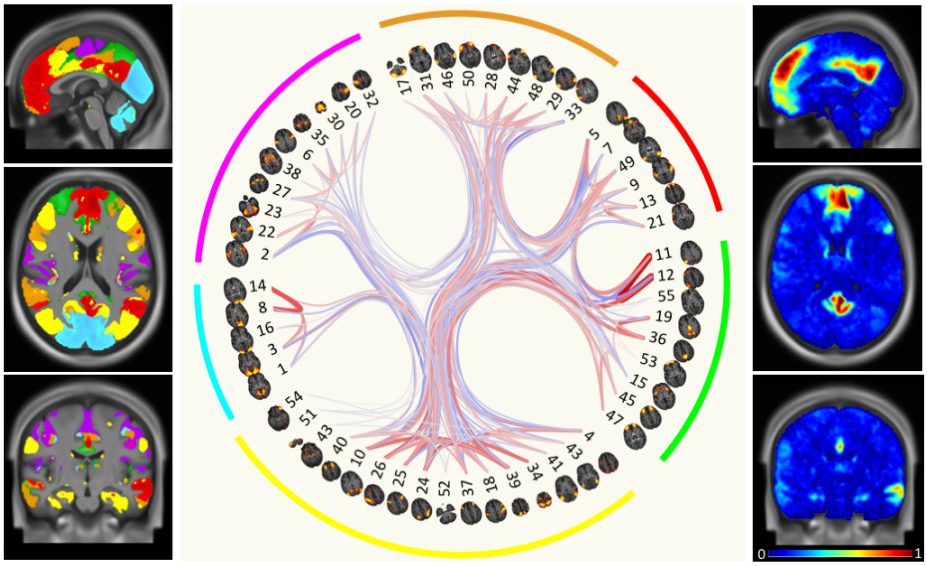


Fig. 3: *Left*: ICA-based brain parcellation shown in groups of six resting-state networks (RSNs), including the default mode network (red). The tree slices shown are, from top to bottom, sagittal, axial, and coronal, at indices 91, 112, and 91, respectively. *Middle*: Connectogram showing the group-averaged functional connectivity between 55 brain regions, which are clustered based on their average population connectivity. Strongest positive and negative correlations are shown in red and blue, respectively. Image is adapted from http://www.fmrib.ox.ac.uk/ukbiobank/netjs_d100/ and enhanced for better visualisation *Right*: Population-level saliency maps, combined for both genders.

ability to visualise the most important features remains as a limitation compared to classical linear models. Future work will focus on the applicability of the method to other graph-centric problems (*e.g.* regression). For instance, a GCN model can be trained for age prediction and consequently used to identify brain regions for which connectivity is most affected with ageing. Another interesting direction entails extending this work for directed/dynamic, *e.g.* time-varying, graphs, as well as using it for biomarker identification.

Acknowledgements. This research has been conducted using the UK Biobank Resource under Application Number 12579 and funded by the EPSRC Doctoral Prize Fellowship funding scheme.

References

1. Bronstein, M.M., Bruna, J., LeCun, Y., Szlam, A., Vandergheynst, P.: Geometric deep learning: going beyond euclidean data. IEEE Signal Processing Magazine

34(4) (2017) 18–42
2. Simonyan, K., Vedaldi, A., Zisserman, A.: Deep inside convolutional networks: Visualising image classification models and saliency maps. arXiv preprint arXiv:1312.6034 (2013)
3. Satterthwaite, T.D., Wolf, D.H., et al.: Linked sex differences in cognition and functional connectivity in youth. Cerebral cortex **25**(9) (2014) 2383–2394
4. Ritchie, S.J., Cox, S.R., Shen, X., et al.: Sex differences in the adult human brain: Evidence from 5,216 uk biobank participants. bioRxiv (2017)
5. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Learning deep features for discriminative localization. In: Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on, IEEE (2016) 2921–2929
6. Monti, F., Boscaini, D., Masci, J., Rodola, E., Svoboda, J., Bronstein, M.M.: Geometric deep learning on graphs and manifolds using mixture model cnns. In: Proc. CVPR. Volume 1. (2017) 3
7. Defferrard, M., Bresson, X., Vandergheynst, P.: Convolutional neural networks on graphs with fast localized spectral filtering. In: Advances in Neural Information Processing Systems. (2016) 3844–3852
8. Levie, R., Monti, F., Bresson, X., Bronstein, M.M.: Cayleynets: Graph convolutional neural networks with complex rational spectral filters. arXiv preprint arXiv:1705.07664 (2017)
9. Zhou, Z., Li, X.: Convolution on graph: A high-order and adaptive approach. (2018)
10. Veličković, P., Cucurull, G., Casanova, A., Romero, A., Liò, P., Bengio, Y.: Graph attention networks. arXiv preprint arXiv:1710.10903 (2017)
11. Baumgartner, C.F., Koch, L.M., Tezcan, K.C., Ang, J.X., Konukoglu, E.: Visual feature attribution using wasserstein gans. arXiv preprint arXiv:1711.08998 (2017)
12. Shuman, D.I., Narang, S.K., Frossard, P., Ortega, A., Vandergheynst, P.: The emerging field of signal processing on graphs: Extending high-dimensional data analysis to networks and other irregular domains. IEEE Sig Proc Mag **30**(3) (2013) 83–98
13. Zhou, B., Khosla, A., Lapedriza, A., Oliva, A., Torralba, A.: Object detectors emerge in deep scene cnns. arXiv preprint arXiv:1412.6856 (2014)
14. Lin, M., Chen, Q., Yan, S.: Network in network. arXiv preprint arXiv:1312.4400 (2013)
15. Miller, K.L., Alfaro-Almagro, F., Bangerter, N.K., et al.: Multimodal population brain imaging in the uk biobank prospective epidemiological study. Nature neuroscience **19**(11) (2016) 1523
16. Smith, S.M., Hyvärinen, A., Varoquaux, G., Miller, K.L., Beckmann, C.F.: Group-pca for very large fmri datasets. Neuroimage **101** (2014) 738–749
17. Beckmann, C.F., Smith, S.M.: Probabilistic independent component analysis for functional magnetic resonance imaging. IEEE TMI **23**(2) (2004) 137–152
18. Smith, S.M., Miller, K.L., Salimi-Khorshidi, G., Webster, M., Beckmann, C.F., Nichols, T.E., Ramsey, J.D., Woolrich, M.W.: Network modelling methods for fmri. Neuroimage **54**(2) (2011) 875–891
19. Ktena, S.I., Parisot, S., Ferrante, E., Rajchl, M., Lee, M., Glocker, B., Rueckert, D.: Metric learning with spectral graph convolutions on brain connectivity networks. NeuroImage **169** (2018) 431 – 442
20. Arslan, S., Ktena, S.I., Makropoulos, A., Robinson, E.C., Rueckert, D., Parisot, S.: Human brain mapping: A systematic comparison of parcellation methods for the human cerebral cortex. NeuroImage **170** (2018) 5 – 30 Segmenting the Brain.

A Graph Representation and Similarity Measure for Brain Networks with Nodal Features

Yusuf Osmanlıoğlu[1(✉)], Birkan Tunç[2], Jacob A. Alappatt[1], Drew Parker[1], Junghoon Kim[3], Ali Shokoufandeh[4], and Ragini Verma[1]

[1] Center for Biomedical Image Computing and Analytics, Department of Radiology, University of Pennsylvania, Philadelphia, USA
yusuf.osmanlioglu@uphs.upenn.edu
[2] Center for Autism Research, Childrens Hospital of Philadelphia, Philadelphia, USA
[3] CUNY School of Medicine, The City College of New York, New York, USA
[4] Department of Computer Science, Drexel University, Philadelphia,USA

Abstract. The human brain demonstrates a network structure that is commonly represented using graphs with pseudonym connectome. Traditionally, connectomes encode only inter-regional connectivity as edges, while regional information, such as centrality of a node that may be crucial to the analysis, is usually handled as statistical covariates. This results in an incomplete encoding of valuable information. In order to alleviate such problems, we propose an enriched connectome encoding regional properties of the brain network, such as structural node degree, strength, and centrality, as node features in addition to representing structural connectivity between regions as weighted edges. We further present an efficient graph matching algorithm, providing two measures to quantify similarity between enriched connectomes. We demonstrate the utility of our graph representation and similarity measures on classifying a traumatic brain injury dataset. Our results show that the enriched representation combining nodal features and structural connectivity information with the graph matching based similarity measures is able to differentiate the groups better than the traditional connectome representation.

Keywords: annotated brain networks, brain graphs, multi-feature representation, graph matching

1 Introduction

Connectomes can be described as a graph of organized regions and their connections that putatively have foundational roles in emerging functional and cognitive outcomes [1]. Hence, many analyses in cognition, learning, and brain diseases and disorders investigate the organization of the brain [2]. Graph theoretical approaches such as complex network analysis provide powerful tools to study structural and functional characteristics of the brain without losing its organizational features [3].

In traditional connectomes, when representing the brain as a network, the nodes of the network correspond to the brain regions, and the edges between the nodes correspond to connections between those regions. In this approach, networks encode only inter-regional connectivity. The regional information such as degree, strength, or centrality that may be crucial to the analysis are usually treated as confounding factors or covariates. This hinders interpretations regarding regional changes due to, for instance, an underlying pathology. However, graph theory facilitates a principled methodology to combine regional characteristics (node features) with interactions between regions (edge features), by means of annotating nodes of the network [4]. Hence, the first contribution of this study is to provide a rich brain network representation, an enriched connectome, that enables inclusion of such nodal features when modeling brain connectivity.

Such a rich representation of brain organization including nodal features requires a new set of tools such as a similarity measure between these networks (graphs) which is essential for classification, clustering, or regression tasks [5, 6]. As a second contribution, we propose a graph matching algorithm that provides a similarity measure between brain networks with nodal features. Among several approaches proposed in the literature to calculate graph similarity over brain data such as seeded graph matching [7] and graph embedding [8], graph edit distance (GED) is arguably the most effective method with promising results [9, 10]. However, high running time complexity of GED requires use of approximation techniques such as Hungarian algorithm in [11] and hinders a detailed analysis of edge features [12]. We approach the graph matching problem as an instance of the *metric labeling problem* [13] and provide an efficient approximation algorithm using the primal-dual scheme [14] by extending our previous study [15]. Our graph matching method achieves two goals simultaneously: finding a mapping between brain regions of different graphs and computing a similarity score. The enriched connectome along with the graph-based similarity measure facilitates its use in classification of samples and we demonstrate its effective application on a traumatic brain injury (TBI) dataset. Results show that our enriched connectome along with the proposed matching algorithm provides better classification between the groups than the traditional connectivity based connectome representation.

2 Materials and Method

2.1 Dataset

Participants: We use a traumatic brain injury dataset consisting 39 patients (12 female) with moderate-to-severe TBI examined at 3 months post injury and 30 healthy controls (8 female). Age of patients are in [18,65] years with a mean of 35 years and standard deviation of 14.7 years, while the age of healthy controls are in [20,56] years with a mean and standard deviation of 34.7 and 9.9 years, respectively. Duration of post-traumatic amnesia of patients, which can be considered as a measure of trauma severity, has a mean of 26.7 days with a standard deviation of 21.2 days.

Data Acquisition and Preprocessing: For each subject, DTI data was acquired on a Siemens 3T TrioTim scanner with a 8 channel head coil (single shot spin echo sequence, TR/TE = 6500/84 ms, $b = 1000$ s/mm^2 , 30 gradient directions). 86 region of interests from the Desikan atlas[16] were extracted to represent the nodes of the structural network. A mask was defined using voxels with an FA of at least 0.1 for each subject. Deterministic tractography was performed to generate and select 1 million streamlines, seeded randomly within the mask. Angle curvature threshold of 60 degrees, and a min and max length threshold of $5mm$ and $400mm$ were applied, resulting in an 86×86 adjacency matrix of weighted connectivity values, where each element represents the number of streamlines between regions.

2.2 Enriched Connectome

Given parcellation of the brain into 86 regions, we constructed a weighted undirected graph with 86 nodes corresponding to brain regions and weighted edges corresponding to the number of fibers connecting region pairs. We annotate each node with two set of features. First, we generated a 6 dimensional feature vector by calculating various graph theoretical measures for each node, namely degree, strength, betweenness centrality, local efficiency, participation coefficient, and local assortativity, using the Brain Connectivity Toolbox [17]. While calculating participation coefficient of nodes, we used association of structural regions with 7 functional systems as described in [18]. Second, we generated an 86 dimensional feature vector, representing the weighted connectivity of each node to the rest of the nodes in the graph, where we considered self edges to be nil. In summary, our graph representation, denoted *enriched connectome* hereby, incorporates graph theoretical measures of the connectome along with the weighted connectivity of the regions that are to be found in network representations. We normalized the values of each graph theory measure and the edge weights to [0,1] in order to make them comparable across subjects.

2.3 Graph Matching based Similarity Measure

We propose taking a graph matching approach to define a similarity measure between two enriched connectomes, while providing a mapping between their nodes. We note that since the brains are parcellated into a common atlas in our setup, mapping between the regions are known a priori. However, we expect to get several mismatching nodes between dissimilar enriched connectomes due to differences in the connectivity of the network, making the similarity of graphs and the ratio of mismatching nodes effective measures for identifying patients from controls.

To this end, we evaluate the graph matching as a special case of the metric labeling problem [13]. Translating the metric labeling into the domain of brain graphs, the problem reads as follows: Given two enriched connectomes A and B, find the optimal one-to-one mapping $f : \mathcal{A} \to \mathcal{B}$ between their nodes while

minimizing the following objective function:

$$\beta \sum_{a \in \mathcal{A}} c(a, f(a)) + (1 - \beta) \sum_{a, a' \in \mathcal{A}} w(a, a') \cdot d(f(a), f(a')). \tag{1}$$

The first summation term in (1) is regarded as the *assignment cost* with $c : \mathcal{A} \times \mathcal{B} \rightarrow \mathbb{R}$ that determines the cost of mapping a brain region $a \in \mathcal{A}$ to a region $b \in \mathcal{B}$, which we define as $||v1_a - v1_b||_2 + ||v2_a - v2_b||_2$ where $v1$ and $v2$ indicate the graph theoretical and edge weight based feature vectors described earlier, respectively. The second summation term stands for the *separation cost*, penalizing strongly connected brain regions $a, a' \in \mathcal{A}$ in getting mapped to loosely connected regions $b, b' \in \mathcal{B}$ with $w : \mathcal{A} \times \mathcal{A} \rightarrow \mathbb{R}$ indicating edge weights in $\mathcal{A}$ as a measure of connectivity strength and $d : \mathcal{B} \times \mathcal{B} \rightarrow \mathbb{R}$ indicating the distance between nodes of $\mathcal{B}$ as a measure of proximity between regions which is defined inversely proportional to the w in $\mathcal{B}$. Thus, the first half of the cost function encourages mapping each node of $\mathcal{A}$ to a node that resembles it most in $\mathcal{B}$ while the second half discourages two strongly connected regions in $\mathcal{A}$ getting mapped to two loosely connected regions in $\mathcal{B}$. The variable β in (1) is a balancing term to adjust the contribution of the assignment and separation costs to the objective function which takes values in [0,1]. Once optimized, summation of the costs in (1) defines a similarity score between the two graphs.

In their seminal paper, Kleinberg and Tardos presented the following quadratic optimization formulation for the metric labeling problem which they showed to be computationally intractable to solve [13]:

$$\begin{array}{lll} \min & \displaystyle\sum_{\substack{a \in \mathcal{A} \\ b \in \mathcal{B}}} c(a,b) \cdot x_{a,b} + \sum_{\substack{a, a' \in \mathcal{A} \\ b, b' \in \mathcal{B}}} w(a,a') \cdot d(b,b') \cdot x_{a,b} \cdot x_{a',b'} & \\ \text{s.t.} & \sum_{b \in \mathcal{B}} x_{a,b} = 1, & \forall a \in \mathcal{A} \\ & \sum_{a \in \mathcal{A}} x_{a,b} = 1, & \forall b \in \mathcal{B} \\ & x_{a,b} \in \{0,1\}, & \forall a \in \mathcal{A}, b \in \mathcal{B} \end{array} \tag{2}$$

where $x_{a,b}$ is an indicator variable taking value 1 if a is mapped to b and 0 otherwise. They also presented a linear programming formulation of the problem along with an approximation algorithm using hierarchically well-separated trees (HST), which was inefficient due to the computational time it takes to build the HSTs and to solve the linear program. Using another integer linear programming formulation of the problem along with a primal-dual approximation scheme [14], we recently presented an efficient approximation algorithm for the traditional metric labeling problem [15]. Here, we extend the latter study by altering the constraints of the metric labeling to account for the particular case of matching the enriched connectomes. Traditional metric labeling formulation allows many-to-one matching of the nodes between graphs, that is, several nodes of the first graph can be mapped to the same node in the second graph. In the setup of enriched connectomes where the brains are registered to a common atlas and parcellated into the same number of regions across subjects, it is known a priori that there should be a one-to-one mapping between the nodes of

the graphs. Motivated by this observation, we impose additional constraints to the metric labeling formulation to enforce a one-to-one mapping between graphs. Our extended version of the metric labeling with the integer linear programming formulation is as follows:

$$
\begin{array}{lll}
\min & \displaystyle\sum_{\substack{a\in\mathcal{A}\\ b\in\mathcal{B}}} c(a,b)\cdot x_{a,b} + \sum_{\substack{a,a'\in\mathcal{A}\\ b,b'\in\mathcal{B}}} w(a,a')\cdot d(b,b')\cdot x_{a,b,a',b'} & \\
\text{s.t.} & \sum_{b\in\mathcal{A}} x_{a,b} = 1, & \forall a\in\mathcal{A} \\
& \sum_{a\in\mathcal{B}} x_{b,a} = 1, & \forall b\in\mathcal{B} \\
& \sum_{a'\in\mathcal{A}} x_{a,b,a',b'} = x_{a,b}, & \forall a\in\mathcal{A}, b, b'\in\mathcal{B} \\
& \sum_{b'\in\mathcal{B}} x_{a,b,a',b'} = x_{a,b}, & \forall a, a'\in\mathcal{A}, b'\in\mathcal{B} \\
& x_{a,b,a',b'} = x_{a',b',a,b}, & \forall a\neq a'\in\mathcal{A}, b\neq b'\in\mathcal{B} \\
& x_{a,b}\in\{0,1\}, x_{a,b,a',b'}\in\{0,1\}, & \forall a, a'\in\mathcal{A}, b, b'\in\mathcal{B}.
\end{array}
\tag{3}
$$

Note that, the formulation (3) replaces the quadratic term $x_{a,b}\cdot x_{a',b'}$ in (2) with the indicator variable $x_{a,b,a',b'}$, introducing $O(n^4)$ new variables and $O(n^3+n^4)$ additional constraints relative to the linear programming formulation. Despite the increase in the size of the problem, this formulation allows applying the primal-dual scheme to efficiently achieve approximate results.

In order to get a primal-dual approximation algorithm for solving the metric labeling in its extended version in (3), we first state the dual of the formulation as follows:

$$
\begin{array}{lll}
\max & \displaystyle\sum_{a\in\mathcal{A}} y_a + \sum_{b\in\mathcal{B}} y_b & \\
\text{s.t.} & y_a + y_b - \sum_{a'\in\mathcal{A}} y_{a,b,a'} - \sum_{b\in\mathcal{B}} y_{a,b,b'} \leq c_{a,b}, & \forall a\in\mathcal{A}, b\in\mathcal{B} \\
& \left.\begin{array}{l} y_{a,b,a'} + y_{a,b,b'} + y_{a,b,a',b'} \leq w_{a,a'}\cdot d_{b,b'} \\ y_{a',b,a} + y_{a,b,b'} - y_{a,b,a',b'} \leq w_{a',a}\cdot d_{b,b'} \end{array}\right\}, & \forall a, a'\in\mathcal{A}, b, b'\in\mathcal{B} \\
& y_p\ ,\ y_{a,b,a'}\ ,\ y_{a,b,b'}\ ,\ y_{a,b,a',b'} \text{ unrestricted}, & \forall a, a'\in\mathcal{A}, b, b'\in\mathcal{B} \\
& y_a\geq 0, y_b\geq 0, & \forall a\in\mathcal{A}, b\in\mathcal{B}
\end{array}
\tag{4}
$$

Since the variables of type $y_{a,b,a',b'}$ appears as a summation and a subtraction in the second type of constraints of (4) which accounts for the balancing constraints in (3), strictly following the primal-dual method presented in [14] would require making assignments in tuples since it enforces dual feasibility throughout the algorithm, resulting in poor performance. As we previously suggested in [15], we relax the dual feasibility condition for the first type of the dual constraints that previously became tight and present an efficient primal-dual approximation algorithm for the problem in Algorithm 1.

The algorithm starts by initializing indicative variables $x_{a,b}$, set of unassigned nodes $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$, and an adjusted assignment cost function ϕ where the value of $\phi(a,b)$ is initially set to be the assignment cost of a to b (line 1). In each iteration of the loop in lines $2-7$, the algorithm maps a node a to a node b that minimizes the adjusted assignment cost function ϕ (lines $3-4$). Before proceeding to the next iteration, assigned nodes a and b are removed from the sets $\hat{\mathcal{A}}$ and $\hat{\mathcal{B}}$ (line 5) and ϕ function is updated for each of the unassigned nodes in the set $\hat{\mathcal{A}}$ by

Algorithm 1 A primal-dual approximation algorithm for approximating (3)

procedure Graph-match($\mathcal{P}, \mathcal{L}$)
1: $\forall a, a' \in \mathcal{A}, b \in \mathcal{B} : x_{a,b} \leftarrow 0, \hat{\mathcal{A}} \leftarrow \mathcal{A}, \hat{\mathcal{B}} \leftarrow \mathcal{B}$
$\phi(a, b) \leftarrow c_{a,b}$
2: **while** $\hat{\mathcal{A}} \neq \emptyset$ **do**
3: Find $a \in \hat{\mathcal{A}}$ that minimizes $\phi(a, b)$ for some $b \in \hat{\mathcal{B}}$
4: $x_{a,b} \leftarrow 1$
5: $\hat{\mathcal{A}} \leftarrow \hat{\mathcal{A}} \setminus \{a\}, \hat{\mathcal{B}} \leftarrow \hat{\mathcal{B}} \setminus \{b\}$
6: $\forall a' \in \hat{\mathcal{A}}, b' \in \hat{\mathcal{B}} : \phi(a', b') = \phi(a', b') + w_{a,a'} \cdot d_{b,b'}$
7: **end while**
8: **return** $\mathcal{X} = \{x_{a,b} : \forall a \in \mathcal{A}, b \in \mathcal{B}\}$

an amount of separation cost with respect to the recently assigned nodes (line 6). Algorithm iterates until no unassigned node is left in $\hat{\mathcal{A}}$.

We note that, $\phi(a, b)$ is not updated for a node a once it gets assigned, rendering the summation $\sum_{a,b} \phi(a, b)x_{a,b}$ to be equal to the similarity score between the two graphs since it is equal to the value of the objective function in (4) which in turn is equal to the value of the objective function in (3).

3 Results

Here, we demonstrate the utility of our brain network representation and similarity measure on a TBI dataset, where the goal is the binary classification of subjects into healthy controls and TBI patients. We used k-nearest neighbors (kNN) classifier.

3.1 TBI Classification

We used nested leave-one-out approach for cross validation, due to limited number of subjects. For each subject in the dataset, we used the remaining 68 subjects of the dataset as the training set. Using an inner leave-one-out cross validation with training set, we decided the balancing parameter β and the number of neighbors k to be used in the nearest neighbor search. Then, we tested each subject with the learned parameter tuple that achieved best classification accuracy.

For comparison purposes, we performed the experiment using two scenarios. First (baseline), we used only a traditional connectome where we represented edge weights in a vector form without a graph representation. Similarity between subjects is calculated using Euclidean distance between these vectorized edge weights (denoted L_2-dist). Second, we use enriched connectome with Algorithm 1 (denoted Graph-match). Note that, Graph-match allows regions of the first graph to get mapped to any one of the regions in the second graph while L_2-dist inherently assumes an identity matching between the nodes of two graphs. The comparison of two scenarios, i.e., traditional connectome with L_2-dist vs.

Table 1. Classification results from leave-one-out cross validation for the two scenarios: traditional connectome with L_2-dist vs. enriched connectome with Algorithm 1.

Scenario	Accuracy	Sensitivity	Specificity
Traditional connectome & L_2-dist	66.7	51.28	86.67
Enriched connectome & Graph-match	72.46	71.19	73.33

enriched connectome with Graph-match, facilitates subsequent analysis and interpretation on regional matches between brains, possibly providing insights into TBI-induced regional differences.

Classification performance is presented in Table 1 for the two scenarios, showing overall accuracy, specificity, and sensitivity. Comparing overall accuracy of the two scenarios suggests that our graph representation with the similarity measure captures more information to decide about the classification than the baseline. As suggested from the results, incorporating nodal features into the representation along with connectivity information improves the classification accuracy. In addition to this, relaxing node mappings between enriched connectomes in Graph-match makes it possible to capture subtle regional alterations, possibly associated with injury, which is reflected by the increased classification performance of Graph-match. We also note that, our approach achieves similar performance for classifying patients and controls as the sensitivity and specificity have similar values whereas traditional connectome with L_2-dist performs poorly for classifying patients. The comparison of ROC curves presented in Figure 1.a demonstrates the improved performance of our method over the baseline.

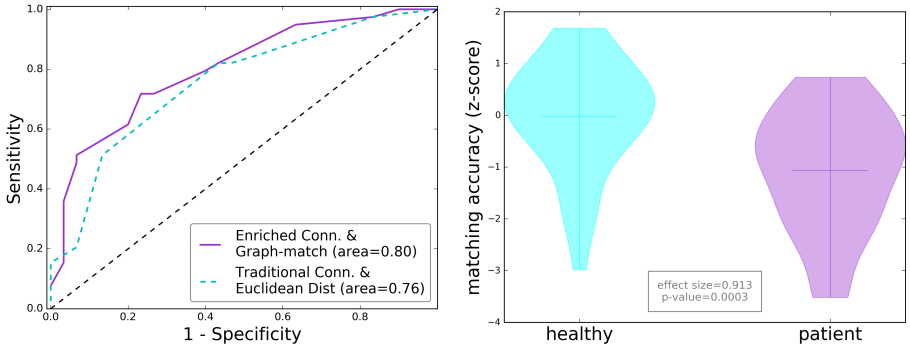
Nested leave-one-out cross validation scheme results in 69 different parameter tuples (β,k) for our method and 69 k values for the baseline approach. In our experiments, we observed that parameter values were mostly consistent for our method across runs. Specifically, we observed that the inner loop of the experiment has chosen $\beta = 0.9$ without any exception and $k = 15$ with only five outliers out of 69 iterations for our method. This can be contrasted to $k = 6$ being chosen for the baseline approach along with 9 outliers, suggesting the robustness of our graph matching algorithm.

3.2 Effect of Feature Types

In order to demonstrate the contribution of graph theory measures and edge weights as node features, in Table 2, classification results for the brain networks with only graph theoretical features and only edge weights as features are con-

Table 2. Classification results of brain graphs with only graph theoretical features and with only edge weights as features, using Graph-match as the similarity measure.

Node features	**Accuracy**	**Sensitivity**	**Specificity**
Graph theoretical measures alone	42.03	56.41	23.34
Edge weights alone	62.32	46.15	83.34



(a) (b)

Fig. 1. (a)Comparison of ROC curves showing the classification performance of the baseline and the proposed method. (b) Z-score distribution of the matching accuracy for controls and patients with respect to the control population.

trasted to both feature types being combined in a single brain graph. We observe that combining both feature types improve the classification accuracy by 10% indicating that enriched connectome maintains more information by combining various features into a single structure relative to a network having either one of them as its only nodal feature. We also observe that using edge weights alone performs better than using graph theoretical measures alone, which can be attributed to larger number of features present in the former, providing a better feature set for classification. However, combination of the two providing an improvement over both of their individual classification accuracies indicate that the two sets of features represent unique aspects of the connectomics.

3.3 Mapping Between Nodes of Graphs

We note that, graph matching provides a mapping between nodes of the two graphs in addition to the similarity score between them. One might expect the regions of a brain graph to match their counterparts in another brain graph (such as, precentral gyrus in one enriched connectome would be expected to match with the precentral gyrus of another enriched connectome) as the brain anatomy is similar across people, with occasional mismatches due to subject-specific differences in connectivity. Leveraging this observation, we define another similarity measure, denoted *matching accuracy*, as the ratio of regions that are accurately matched with their counterparts to total number of regions. Matching enriched connectome of every subject to the healthy controls, we hypothesize that the matching accuracy of healthy controls with respect to themselves should be higher than the matching accuracy of patients with respect to healthy control population, as structural alterations introduced by TBI is expected to cause mismatching regions. As shown in Figure 1.b, we observe a statistically significant group difference between the patients and controls in their matching accuracy

Hierarchical Bayesian Networks for Modeling Inter-Class Dependencies: Application to Semi-Supervised Cell Segmentation

Hamid Fehri[1,2,3], Ali Gooya[1], Yuanjun Lu[1], Simon A. Johnston[2,3], and Alejandro F. Frangi[1]

[1] Center for Computational Imaging Simulation Technologies in Biomedicine (CISTIB), The University of Sheffield, Sheffield, United Kingdom
[2] Bateson Centre, Firth Court, University of Sheffield
[3] Department of Infection, Immunity and Cardiovascular Disease, Medical School, University of Sheffield

Abstract. Quantification of cell-mediated biological interactions requires segmenting different types of cellular and sub-cellular objects in the image and is a prerequisite to disease identification and classification. Limited ground truth and complex irregular structures in these images make automatic segmentation challenging. We propose a graphical model based segmentation that enables training from weakly labeled images. A hierarchical graph is first generated by oversegmenting the image and then merging the superpixels. Learning from a small set of training nodes in the graph, the labels of all nodes are inferred through an efficient message-passing algorithm and the parameters of the model are optimized by Expectation Maximization (EM). We employ polytrees to effectively impose prior knowledge on the inclusion of different classes by capturing both same-level and across-level dependencies. The proposed model is evaluated on synthetic and real microscopy images and compares favorably to GMM classifier, directed tree and SegNet.

1 Introduction

Cell segmentation is a fundamental first step in quantifying biological cell-mediated behaviors that facilitates understanding the physiology of organs and helps disease diagnosis and progression evaluation. For example, the shape and size of certain cell compartments are important indicators for cancer diagnosis and grading [1], and host-pathogen interactions are critical in the outcome of infection [2]. Due to the existence of objects of different types in histology images, analyzing them usually involves solving a multi-class segmentation problem. Since manual segmentation is tedious, subject to inter- and intra-reader variations and has a suboptimal reproducibility, automatic methods are favorable. However, the complex shapes and heterogeneous regions of cells and their low contrast challenge automatic segmentation methods. In these cases, employing inter-class relations based on the nature of the problem can improve segmentation. Examples of these relations are constraining nuclei to be found within cells,

in cell-nucleus segmentation, and allowing the inclusion of pathogens by white blood cells, in host-pathogen cell segmentation.

Inter-class relations have been employed in segmentation using discriminative and generative probabilistic models. Convolutional neural nets (CNNs) are an example of discriminative models that can learn dependencies between different classes through their cascade network architecture [3]. These models perform well, and rely on a set of training samples, with a sufficient quantity and variance, to learn the underlying constraints. On the other hand, generative models, such as Bayesian networks (BNs), impose the interrelations by utilizing a prior term designed based on the problem. Incorporation of prior knowledge for deriving model posteriors reduces the dependency of BNs on the training data, compared to CNNs. Directed acyclic graphs (DAGs) are a type of BNs that provide closed-form posteriors through efficient non-iterative inferences on the graph. Trees and polytrees are two common DAGs that have been used for modeling class dependencies on graphs [4, 5]. These models are similar in having a unique path between every two nodes, with the distinction that nodes in trees (except the highest-level parent node) have exactly one parent node (Fig. 1a), while nodes in polytrees can have multiple parents (Fig. 1b). This feature of polytrees allows them to model both same-level and across-level dependencies that can eliminate blocky artifacts common in tree-based segmentation [6].

Polytree graphical models have been proposed for supervised graph based image segmentation [7], where a hierarchical graph is generated for each image and the segmentation is obtained by labeling the graph nodes. However, that method performs *off-line* training (no optimization), requiring a set of fully segmented images which is time-consuming and expensive to obtain, and is not always available. Additionally, the trained classifier can be suboptimal for segmentation of new images, due to the inherent differences between the train and test images.

In this paper, we derive a novel EM-based polytree segmentation algorithm that infers the posteriors of the graph nodes and refines the model parameters for each image that is sparsely annotated. One synthetic and two real microscopy image datasets were used for evaluation. The proposed method compares favorably to a state-of-the-art convolutional neural network, namely SegNet [3], Gaussian mixture model (GMM) and trees. In addition, by identifying clique label configurations that do not comply with the imposed priors, we show that polytrees are more precise than trees in detecting regions with possible segmentation errors.

2 Method

For image segmentation, a polytree graphical model is firstly generated for the image by grouping similar pixels and regarding them as nodes. An EM algorithm refines the inferred labels through optimization of model parameters. In the E step, an exact non-iterative two-pass procedure infers the posteriors of the nodes, given the parameter updates from the M step. Finally, the segmented image is constructed based on the optimal node labels.

2.1 Data-driven irregular polytree

We first adopt a superpixel representation of the image [8], to group locally similar pixels in the image. These superpixels are recursively merged to obtain a hierarchy of locally coherent areas each corresponding to a node in a merge-polytree (see Fig. 1c) [9]. Depending on the level of the superpixel in the hierarchy, different criteria are employed for merging superpixels. The lower level nodes that correspond to smaller superpixels (i.e. parts of the same objects) can only be merged with their neighboring superpixels. The higher level nodes, on the other hand, that correspond to larger superpixels (i.e. objects or groups of objects) can be merged with similar superpixels, regardless of their location in the image. This scheme makes merging neighboring nodes that correspond to parts of the same object more probable. The generated graph has an irregular structure that adapts to the image and has no loops. We merge two nodes at each merging step that urges each non-leaf node to have two descendant nodes (Fig. 1b). This three-wise clique is denoted by $descendant1 - node - descendant2$.

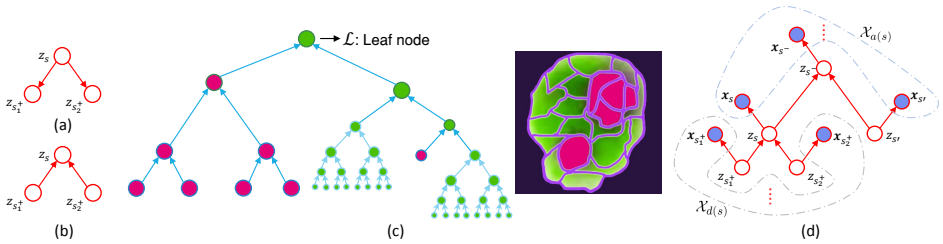


Fig. 1. Graphical models for image segmentation. Panel (a) shows a clique in trees, where nodes $z_{s_1^+}$ and $z_{s_2^+}$ both have one parent, z_s. Panel (b) shows a clique in polytrees, where node z_s has two parents $z_{s_1^+}$ and $z_{s_2^+}$. A sample merge-polytree is depicted for an oversegmented image of a host cell (green) containing two pathogens (magenta), in (c). Panel (d) shows the notation of nodes neighboring node s and the graphical representations of $\mathcal{X}_{a(s)}$ and $\mathcal{X}_{d(s)}$ observation node sets.

2.2 Inference

Herewith, we describe how the posteriors of nodes given the parameters of the likelihood functions are computed in an exact form. This is implemented in the E step of the proposed EM algorithm (see Section 2.3). Sets of labels and observed features are denoted by $\mathbb{Z} = \{z_s\}$ and $\mathbb{X} = \{\boldsymbol{x}_s\}$, respectively, the set of graph nodes and edges is denoted by $\mathcal{G}$, and Λ is the set of all possible class labels. For an internal node s (neither in the lowest level nor the leaf node), s^-, s^+ and s' denote nodes in higher, lower and same layers, respectively (Fig. 1d).

Factorization of the joint probability of a clique in directed trees (Fig. 1a); $p(\mathbb{Z}) = p(z_{s_1^+}|z_s)p(z_{s_2^+}|z_s)p(z_s)$ involves across-level constraints $p(z_{s_1^+}|z_s)$ and

$p(z_{s_2^+}|z_s)$. This implies that the labels $z_{s_1^+}$ and $z_{s_2^+}$ are independent, given z_s. However, this conditional independence does not always hold (e.g. when the descendants are nodes corresponding to neighboring superpixels) and is a source of error in tree-based segmentation methods [6]. In polytrees, however, the joint probability of a clique (Fig. 1b); $p(\mathbb{Z}) = p(z_s|z_{s_1^+}, z_{s_2^+})p(z_{s_1^+})p(z_{s_2^+})$ comprises of three-wise constraints $p(z_s|z_{s_1^+}, z_{s_2^+})$, that capture both across-level and same-level dependencies. This feature enables polytrees to eliminate unfeasible label configurations on the graph more precisely, compared to trees.

For instance, allowing a $background-background-cell$ clique on the tree requires setting both $p(z_{s^+} = background|z_s = background)$ and $p(z_{s^+} = cell|z_s = background)$ conditional probabilities to nonzero values. Doing so will also allow the clique $cell - background - cell$, which is not feasible. On the other hand, allowing $background-background-cell$ for cliques on the polytree requires setting $p(z_s = background|z_{s_1^+} = backgound, z_{s_2^+} = cell)$ to nonzero values. Despite trees, allowing these cliques on the polytree does not lead to the emergence of semantically unfeasible cliques.

We now explain the calculation of posteriors on the polytree. Given the observations $\mathbb{X}$, finding the best segmentation is equivalent to associating the most probable label to each node (Bayesian inference): $\forall s \in \mathcal{G}, \hat{z}_s = \arg\max_{z_s \in \Lambda} p(z_s|\mathbb{X})$. Here, $p(z_s|\mathbb{X})$ is computed by marginalizing the probability of the clique $s_1^+ - s - s_2^+$, given $\mathbb{X}$, that is called the *joint posterior* $p(z_s|\mathbb{X}) = \sum_{z_{s_1^+}, z_{s_2^+}} p(z_s, z_{s_1^+}, z_{s_2^+}|\mathbb{X})$. This way of marginalizing the node's posterior reveals the contribution of three-wise constraints on cliques. Using the D-separation rule [10], the joint posterior is expanded as follows (proofs are removed due to the shortage of space):

$$p(z_s|\mathbb{X}) \propto \sum_{z_{s_{1,2}^+}} \frac{p(z_s, z_{s_1^+}, z_{s_2^+}|\mathcal{X}_{a(s)})}{\sum_{z'_s} p(z'_s, z_{s_1^+}, z_{s_2^+}|\mathcal{X}_{a(s)})} p(z_{s_1^+}, z_{s_2^+}|\mathcal{X}_{a(s_1^+, s_2^+)}) \frac{p(z_{s_1^+}|\mathcal{X}_{d(s_1^+)})}{p(z_{s_1^+})} \frac{p(z_{s_2^+}|\mathcal{X}_{d(s_2^+)})}{p(z_{s_2^+})}, \tag{1}$$

where $p(z_s, z_{s_1^+}, z_{s_2^+}|\mathcal{X}_{a(s)}) = p(z_{s_1^+}, z_{s_2^+}|z_s)p(z_s|\mathcal{X}_{a(s)})$ and $p(z_{s_1^+}, z_{s_2^+}|\mathcal{X}_{a(s_1^+, s_2^+)}) \propto \sum_{z_s} p(\boldsymbol{x}_{s_1^+}|z_{s_1^+})p(\boldsymbol{x}_{s_2^+}|z_{s_2^+})p(z_{s_1^+}, z_{s_2^+}|z_s)p(z_s|\mathcal{X}_{a(s)})$. Note that $\mathcal{X}_{a(.)}$ and $\mathcal{X}_{d(.)}$ refer to sets of ascendant and descendant observation nodes, respectively. For each node s (or a set of nodes $\mathcal{S}$), ascendant nodes are nodes connected to s ($\mathcal{S}$) through edges with outward directions (nodes $\boldsymbol{x}_s$, z_{s^-}, $\boldsymbol{x}_{s^-}$, $z_{s'}$ and $\boldsymbol{x}_{s'}$ in Fig. 1d). Similarly, descendant nodes include the nodes connected to node s ($\mathcal{S}$) through edges with inward directions (nodes $z_{s_1^+}$, $\boldsymbol{x}_{s_1^+}$, $z_{s_2^+}$ and $\boldsymbol{x}_{s_2^+}$ in Fig. 1d).

Two types of terms (messages) emerge on the right-hand side of Eq. 1 in the factorization of the joint posterior for node s: 1) *top-down messages* $p(z_s|\mathcal{X}_{a(s)})$: posterior marginals that are merely dependent on ascendant observation nodes $\mathcal{X}_{a(s)}$, and 2) *bottom-up messages* $p(z_s|\mathcal{X}_{d(s)})$: posterior marginals that are merely dependent on descendant observation nodes $\mathcal{X}_{d(s)}$. We derived recursive procedures that calculate the top-down messages from child nodes for each node, creating a top-down inference pass from the leaf to the root nodes. Similarly, bottom-up messages are calculated from parent nodes of each nodes creating a

bottom-up pass from roots to the leaf. Sweeping the two passes, top-down and bottom-up messages are calculated for each node, based on which the posterior of that node (Eq. 1) is computed for a given set of parameters.

2.3 Parameter Estimation

We use an Expectation-Maximization algorithm for optimizing parameters. The joint probability distribution of the model is expressed as:

$$p(\mathbb{X},\mathbb{Z}|\boldsymbol{\Theta}) = \prod_{s\in\mathcal{R}} p(z_s|\boldsymbol{\Theta}) \prod_{s\notin\mathcal{R},s\in\mathcal{G}} p(z_s|z_{s_1^+},z_{s_2^+},\boldsymbol{\Theta}) \prod_{s\in\mathcal{G}} p(\boldsymbol{x_s}|z_s,\boldsymbol{\Theta}), \tag{2}$$

where $\mathcal{R}$ is the set of root nodes in graph $\mathcal{G}$ and $\boldsymbol{\Theta}$ is the vector of parameters subject to optimization. Using this expansion, expectation of the joint log-likelihood function is calculated as follows (E-step).

$$\begin{aligned} Q = &\sum_{s\in\mathcal{G}}\sum_{i\in\Lambda} p(z_s=i|\mathbb{X},\boldsymbol{\Theta}) \ln p(\boldsymbol{x_s}=\boldsymbol{l}|z_s=i,\boldsymbol{\Theta}) + \sum_{s\in\mathcal{R}}\sum_{i\in\Lambda} p(z_s=i|\mathbb{X},\boldsymbol{\Theta}) \ln p(z_s=i|\boldsymbol{\Theta}) \\ &+ \sum_{s\notin\mathcal{R}}\sum_{(i,j,k)\in\mathbb{F}} p(z_s=i,z_{s_1^+}=j,z_{s_2^+}=k|\mathbb{X},\boldsymbol{\Theta}) \ln p(z_s=i|z_{s_1^+}=j,z_{s_2^+}=k,\boldsymbol{\Theta}), \end{aligned} \tag{3}$$

where $\mathbb{F}\subset\Lambda^3$ is the set of feasible configurations and $\boldsymbol{l}$ is the observed feature vector for node s. Using Lagrange multipliers for maximizing these parameters with the constraints, the following update equations are derived for the M step.

$$\begin{aligned} p(z_s=i|\boldsymbol{\Theta}^{(new)}) &= \frac{1}{|\mathcal{R}|}\sum_{s\in\mathcal{R}}\sum_{(j,k)\in\Lambda^2} p(z_s=i,z_{s_1^+}=j,z_{s_2^+}=k|\mathbb{X},\boldsymbol{\Theta}), \\ p(z_s=i|z_{s_1^+}=j,z_{s_2^+}=k,\boldsymbol{\Theta}^{(new)}) &= \frac{\sum_{s\notin\mathcal{R}} p(z_s=i,z_{s_1^+}=j,z_{s_2^+}=k|\mathbb{X},\boldsymbol{\Theta})}{\sum_{s\notin\mathcal{R}}\sum_{i\in\Lambda} p(z_s=i,z_{s_1^+}=j,z_{s_2^+}=k|\mathbb{X},\boldsymbol{\Theta})}. \end{aligned} \tag{4}$$

Assuming Gaussian distributions for class conditional likelihood functions, $p(\boldsymbol{x}_s=\boldsymbol{l}|z_s=i,\boldsymbol{\Theta})=\mathcal{N}(\boldsymbol{x}_s=\boldsymbol{l}|\boldsymbol{\mu}_i,\boldsymbol{\Sigma}_i)$, the mean and covariance of each class i are updated by:

$$\boldsymbol{\mu}_i^{(new)} = \frac{\sum_{s\in\mathcal{G}} p(z_s=i|\mathbb{X},\boldsymbol{\Theta})\boldsymbol{x}_s}{\sum_{s\in\mathcal{G}} p(z_s=i|\mathbb{X},\boldsymbol{\Theta})}, \tag{5}$$

$$\boldsymbol{\Sigma}_i^{(new)} = \frac{\sum_{s\in\mathcal{G}} p(z_s=i|\mathbb{X},\boldsymbol{\Theta})[\boldsymbol{x}_s-\boldsymbol{\mu}_i^{(new)}][\boldsymbol{x}_s-\boldsymbol{\mu}_i^{(new)}]^T}{\sum_{s\in\mathcal{G}} p(z_s=i|\mathbb{X},\boldsymbol{\Theta})}. \tag{6}$$

Posterior probabilities $p(z_s=i|\mathbb{X},\boldsymbol{\Theta})$ that appear in the two update equations are computed using Eq. 1. The algorithm iterates between E and M steps until convergence is reached (usually within 10 EM iterations). To reconstruct the segmented image, labels of nodes are set as classes for which the posterior probability of that node is the maximum.

3 Experiments and results

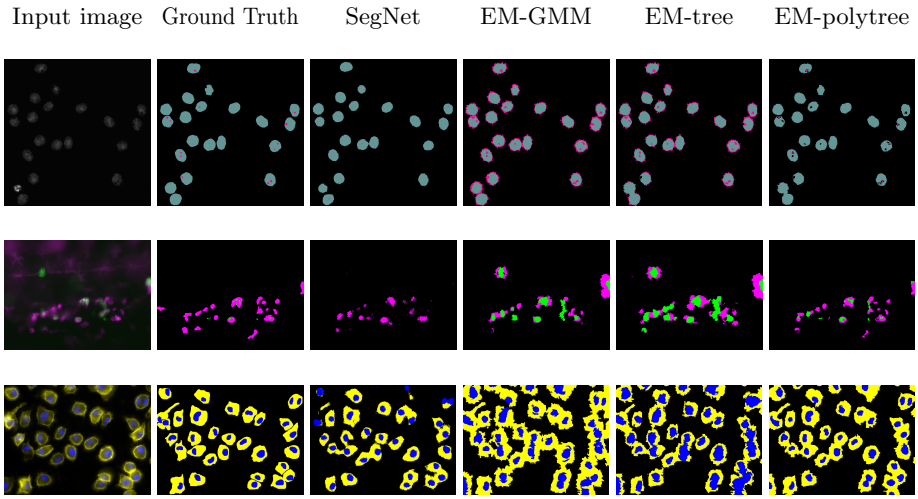


Fig. 2. Evaluation on multi-class image segmentation. The first row shows a sample from the synthetic image set and its segmentations. Light green and magenta colors were respectively used for the nuclei and nucleoli. The second row shows a sample from Zebrafish dataset, where host and pathogen cells are colored magenta and green, respectively. The last row shows a sample from BBBC020 dataset with yellow and blue colors for cells and nuclei, respectively.

3.1 Multi-class image segmentation: synthetic and real microscopy images

The proposed model was evaluated on three datasets, including one synthetic and two real microscopy images. The synthetic dataset was generated by Mitogen framework [11], where 25 images containing cell nuclei and sub-nuclear particles, namely nucleoli, were used for evaluation. Dataset BBBC020 from the set of publicly available datasets on Broad Bioimage Benchmark Collection [12] was used with 20 two-channel microscopy images of bone marrow macrophages and their nuclei. The third dataset (Zebrafish) had 10 random samples taken from Aleksandra Bojarczuk *et al.* [2] with two-channel microscopy images of host and pathogen cells in zebrafish. The three datasets define multi-class segmentation tasks with constraints, each allowing a certain set of label configurations on the graph. For each experiment, 20% of the root nodes were randomly selected and labelled as training for each image. This corresponds to 10% of the whole nodes in the graph, as the graph is binary.

For the synthetic images, we know that the nucleoli can be found only in the nuclei and the nuclei are to be found in the background. Similarly, in BBBC020

dataset, nuclei can only be within cells and cells can be in the background. For the Zebrafish dataset, pathogens can exist both inside and outside the host cells, and host cells exist in the background. This prior knowledge was incorporated by setting the conditional probabilities $p(z_{s^+}|z_s)$ in trees and $p(z_s|z_{s_1^+}, z_{s_2^+})$ in polytrees and was utilized in the inference.

The segmentation performance of EM-polytrees was compared to that of three methods. SegNet [3] was employed as a CNN designed for multi-class segmentation. EM-GMM and EM-trees were compared to EM-polytrees in modeling no and weaker dependencies, respectively. Scale-space differential invariants were used as features for EM-GMM, EM-tree and EM-polytree, and Gaussian distributions were considered for class-conditional likelihood functions. Figure 2 shows the segmentation results on sample images from the synthetic image set, Zebrafish and BBBC020 datasets. The performances of the four methods were compared by calculating Dice similarity coefficients (DSC) between the segmented images and ground truth. Table 1 shows that the EM-polytree outperforms the other three methods on the three datasets, except for the segmentation of nuclei in BBBC020. However, the average multi-class DSCs of EM-polytree were significantly higher than the other methods in all datasets.

Table 1. Median Dice similarity coefficients in multi-class cell segmentation.

Dataset	Synthetic		BBBC020		Zebrafish	
class	nucleus	nucleolus	cell	nucleus	macrophage	pathogen
SegNet	0.84	0.05	0.72	**0.79**	0.29	0.01
EM-GMM	0.91	0.19	0.83	0.72	0.27	0.03
EM-tree	0.72	0.22	0.62	0.70	0.23	0.03
EM-polytree	**0.94**	**0.30**	**0.88**	0.78	**0.60**	**0.20**

3.2 Segmentation error prediction

The proposed method can evaluate the compliance of the segmented image with the imposed priors. A strong non-compliance can indicate a faulty segmentation that should be flagged up for manual annotation. This was implemented by assessing the label configurations on the labeled graph that do not comply with the priors, i.e. $p(z_s|z_{s_1^+}, z_{s_2^+})$ in polytrees and $p(z_{s_1^+}|z_s)$ and $p(z_{s_2^+}|z_s)$ in trees (see Section 3.1). Areas of the segmented image that correspond to such label configurations were nominated as the predicted error and Dice similarity coefficients were calculated between the annotated areas and the actual segmentation error. Figure 3a shows the predicted segmentation error for a sample image from BBBC020 dataset by EM-polytree and EM-tree. The DSC between the predicted and actual segmentation error over the entire dataset in Fig. 3b shows that the EM-polytree outperforms the EM-tree in predicting the error.

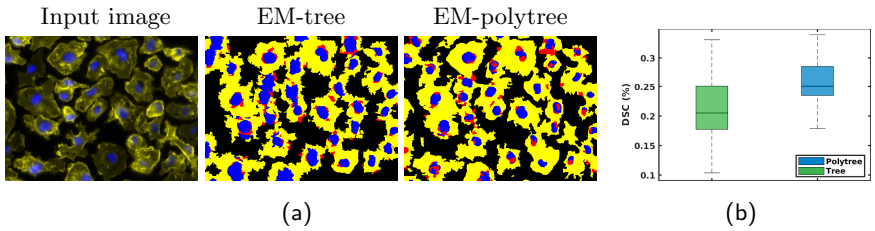


Fig. 3. Performance of EM-tree and EM-polytree in error prediction. Panel (a) shows errors annotated by red on a sample image from BBBC020 dataset. Panel (b) shows the Dice similarity coefficients between the predicted and actual segmentation error for the two methods (P-value of the pairwise t-test was 0.005).

4 Discussion

We present a semi-supervised multi-class segmentation method with a state-of-the-art accuracy on the fluorescence microscopy images. Applications to synthetic and real microscopy images indicate the benefits of the proposed model over the methods investigated. The improved Dice coefficients on segmentations and predicted errors suggest that polytrees can effectively capture inter-class dependencies in segmentation. Trees and GMMs cannot incorporate these dependencies as accurately as polytrees. SegNet, on the other hand, relies on the implicit learning of the constraints, which depends on the size and variation of the training set and is challenging for small datasets.

This work paves the way for the application of polytrees for segmentation when training data is limited. Alternatively, it can be applied to datasets with significant variances from one image to another, where simply training a global classifier (such as SegNet) is not effective. Additionally, the Bayesian generative probabilistic nature of the method enables quantifying non-compliance of the inferred labels with the prior knowledge. This feature can be used to mark up areas that are prone to segmentation error and require manual annotations. The proposed EM algorithm is computationally efficient (about 10 seconds on an Intel Xeon(R) CPU E5620 2.40GHz machine) and thus can be individually applied to each image to boost the segmentation accuracy. To assess the scalability of the algorithm, we inferred on randomly generated graphs with 20 to 200,000 nodes. Results show the time of inference, t, scales with the number of nodes, n, approximately with: $t = 5e^{-5}n^{1.3}$, showing a substantial scalability.

References

1. Alexander, J., Kendall, J., McIndoo, J., Rodgers, L., Aboukhalil, R., Levy, D., Stepansky, A., Sun, G., Chobardjiev, L., Riggs, M., et al.: Utility of single-cell genomics in diagnostic evaluation of prostate cancer. Cancer research **78**(2) (2018) 348–358
2. Bojarczuk, A., Miller, K., Hotham, R., Lewis, A., Ogryzko, N., Kamuyango, A., Frost, H., Gibson, R., Stillman, E., May, R., et al.: Cryptococcus neoformans

intracellular proliferation and capsule size determines early macrophage control of infection. Scientific reports **6** (2016) 21489–21489
3. Badrinarayanan, V., Kendall, A., Cipolla, R.: Segnet: A deep convolutional encoder-decoder architecture for image segmentation. IEEE transactions on pattern analysis and machine intelligence **39**(12) (2017) 2481–2495
4. Wang, Y., Zhang, N.L., Chen, T., Poon, L.K.: Ltc: A latent tree approach to classification. International journal of approximate reasoning **54**(4) (2013) 560–572
5. Zaveri, M., Hammerstrom, D.: Cmol/cmos implementations of bayesian polytree inference: Digital and mixed-signal architectures and performance/price. IEEE Transactions on Nanotechnology **9**(2) (2010) 194–211
6. Wolf, C., Gavin, G.: Inference and parameter estimation on hierarchical belief networks for image segmentation. Neurocomputing **73** (2010) 563–569
7. Fehri, H., Gooya, A., Johnston, S., Frangi, A.: Multi-class image segmentation in fluorescence microscopy using polytrees. In: Information Processing in Medical Imaging. Volume 10265., Springer, Cham (2017) 517–528
8. Van den Bergh, M., Boix, X., Roig, G., Van Gool, L.: Seeds: Superpixels extracted via energy-driven sampling. International Journal of Computer Vision **111**(3) (2015) 298–314
9. Funke, J., Hamprecht, F.A., Zhang, C.: Learning to segment: Training hierarchical segmentation under a topological loss. In: International Conference on Medical Image Computing and Computer-Assisted Intervention, Springer (2015) 268–275
10. Bishop, C.M.: Pattern Recognition and Machine Learning. Springer-Verlag New York (2006)
11. Svoboda, D., Ulman, V.: Mitogen: A framework for generating 3d synthetic time-lapse sequences of cell populations in fluorescence microscopy. IEEE transactions on medical imaging **36**(1) (2017) 310–321
12. Ljosa, V., Sokolnicki, K.L., Carpenter, A.E.: Annotated high-throughput microscopy image sets for validation. Nature Methods **9**(7) (2012) 637

Multi-modal Disease Classification in Incomplete Datasets Using Geometric Matrix Completion

Gerome Vivar[1 2], Andreas Zwergal[2], Nassir Navab[1], and Seyed-Ahmad Ahmadi[2]

[1] Technical University of Munich (TUM), Munich, GER
[2] German Center for Vertigo and Balance Disorders (DSGZ), Ludwig-Maximilians-Universität (LMU), Munich, GER

Abstract. In large population-based studies and in clinical routine, tasks like disease diagnosis and progression prediction are inherently based on a rich set of multi-modal data, including imaging and other sensor data, clinical scores, phenotypes, labels and demographics. However, missing features, rater bias and inaccurate measurements are typical ailments of real-life medical datasets. Recently, it has been shown that deep learning with graph convolution neural networks (GCN) can outperform traditional machine learning in disease classification, but missing features remain an open problem. In this work, we follow up on the idea of modeling multi-modal disease classification as a matrix completion problem, with simultaneous classification and non-linear imputation of features. Compared to methods before, we arrange subjects in a graph-structure and solve classification through geometric matrix completion, which simulates a heat diffusion process that is learned and solved with a recurrent neural network. We demonstrate the potential of this method on the ADNI-based TADPOLE dataset and on the task of predicting the transition from MCI to Alzheimer's disease. With an AUC of 0.950 and classification accuracy of 87%, our approach outperforms standard linear and non-linear classifiers, as well as several state-of-the-art results in related literature, including a recently proposed GCN-based approach.

1 Introduction

In clinical practice and research, the analysis and diagnosis of complex phenotypes or disorders along with differentiation of their aetiologies rarely relies on a single clinical score or data modality, but instead requires input from various modalities and data sources. This is reflected in large datasets from well-known multi-centric population studies like the Alzheimer's Disease Neuroimaging Initiative (ADNI) and its derived TADPOLE grand challenge [3]. TADPOLE data, for example, comprises demographics, neuropsychological scores, functional and morphological features derived from MRI, PET and DTI imaging, genetics, as well as histochemical analysis of cerebro-spinal fluid. The size and richness of such datasets makes human interpretation difficult, but it makes them highly

[3] http://adni.loni.usc.edu ‖ https://tadpole.grand-challenge.org/

suited for computer-aided diagnosis (CAD) approaches, which are often based on machine learning (ML) techniques [10, 11, 16]. Challenging properties for machine learning include e.g. subjective, inaccurate or noisy measurements or a high number of features. Linear [11] and non-linear [16] classifiers for CAD show reasonable success in compensating for such inaccuracies, e.g. when predicting conversion from mild-cognitive-impairment (MCI) to Alzheimer's disease (AD). Recent work has further shown that an arrangement of patients in a graph structure based on demographic similarity [12] can leverage network effects in the cohort and increase robustness and accuracy of the classification. This is especially valid when combined with novel methods from geometric deep learning [1], in particular spectral graph convolutions [7]. Similar to recent successes of deep learning methods in medical image analysis [8], deep learning on graphs shows promise for CAD, by modeling connectivity across subjects or features.

Next to noise, a particular problem of real-life, multi-modal clinical datasets is missing features, e.g. due to restrictions in examination cost, time or patient compliance. Most ML algorithms, including the above-mentioned, require feature-completeness, which is difficult to address in a principled manner [4]. One interesting alternative to address missing features is to model CAD and disease classification as a matrix completion problem instead. Matrix completion was proposed in [5] for simultaneously solving the three tasks of multi-label learning, transductive learning, and feature imputation. Recently, this concept was applied for CAD in multi-modal medical datasets for the first time [15], for prediction of MCI-to-AD conversion on ADNI data. The method introduced a pre-computed feature weighting term and outperformed linear classifiers on their dataset, however it did not yet leverage any graph-modeled network effects of the population as in [12]. To this end, several recent works incorporated a geometric graph structure into the matrix completion problem [6, 9, 13]. All these methods were applied on non-medical datasets, e.g. for recommender systems [9]. Hence, their goal was solely imputation, without classification. Here, we unify previous ideas in a single stream-lined method that can be trained end-to-end.

Contribution. In this work, we follow up on the idea of modeling multi-modal CAD as a matrix completion problem [5] with simultaneous imputation and classification [15]. We leverage cohort network effects by integrating a population graph with a solution based on geometric deep learning and recurrent neural networks [9]. For the first time, we demonstrate geometric matrix completion (GMC) and disease classification from multi-modal medical data, towards MCI-to-AD prediction from TADPOLE features at baseline examination. In this difficult task, GMC significantly outperforms regular linear and non-linear machine learning methods as well as three state-of-the-art results from related works, including a recent approach based on graph-convolutional neural networks.

2 Methods

2.1 Dataset and Preprocessing

As an example application, we utilize the ADNI-based TADPOLE dataset, with the goal of predicting whether an MCI subject will convert to AD given their baseline information. We select all unique subjects with baseline measurements from ADNI1, ADNIGO, and ADNI2 in the TADPOLE dataset which were diagnosed as MCI including those diagnosed as EMCI and LMCI. Following [15], we retrospectively label those subjects whose condition progressed to AD within 48 months as cMCI and those whose condition remained stable as sMCI. The remaining MCI subjects who progressed to AD after month 48 are excluded from this study. We use multi-modal features from MRI, PET, DTI, and CSF at baseline, i.e. excluding longitudinal features. We use all numerical features from this dataset to stack with the labels and include age and gender to build the graph, following the intuition and methodology from [12].

2.2 Matrix Completion

We will start by describing the matrix completion problem. Suppose there exists a matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$ where the values in this matrix are not all known. The goal is to recover the missing values in this matrix. A well-defined description of this problem is to assume that the matrix is of low rank [2],

$$\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \ \text{rank}(\mathbf{X}) \ \text{ s. t. } \ x_{ij} = y_{ij}, \forall ij \in \Omega, \tag{1}$$

where $\mathbf{X}$ is the $m \times n$ matrix with values x_{ij}, Ω is the set of known entries in matrix $\mathbf{Y}$ with y_{ij} values. However, this rank minimization problem (1) is known to be computationally intractable. So instead of solving for rank($\mathbf{X}$), we can replace it with its convex surrogate known as the nuclear norm $||\mathbf{X}||_*$ which is equal to the sum of its singular values [2]. In addition, if the observations in Y have noise, the equality constraint in equation (1) can be replaced with the squared Frobenius norm $||.||_{\text{F}}^2$ [3],

$$\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} \ ||\mathbf{X}||_* \ + \ \frac{\gamma}{2} ||\boldsymbol{\Omega} \circ (\mathbf{Y} - \mathbf{X})||_{\text{F}}^2 \ , \tag{2}$$

where $\boldsymbol{\Omega}$ is the masking matrix of known entries in $\mathbf{Y}$ and $\circ$ is the Hadamard product. Alternatively, a factorized solution to the representation of the matrix $\mathbf{X}$ was also introduced in [13, 14], as the formulation using the full matrix makes it hard to scale up to large matrices such as the famous Netflix challenge. Here, the matrix $\mathbf{X} \in \mathbb{R}^{m \times n}$ is factorized into 2 matrices $\mathbf{W}$ and $\mathbf{H}$ via SVD, where $\mathbf{W}$ is $m \times r$ and $\mathbf{H}$ is $n \times r$, with $r \ll \min(m, n)$. Srebro et al. [14] showed that the nuclear norm minimization problem can then be rewritten as:

$$\min_{W,H} \frac{1}{2} ||\mathbf{W}||_{\text{F}}^2 + \frac{1}{2} ||\mathbf{H}||_{\text{F}}^2 + \frac{\gamma}{2} ||\boldsymbol{\Omega} \circ (\mathbf{W}\mathbf{H}^T - \mathbf{Y})||_{\text{F}}^2 \tag{3}$$

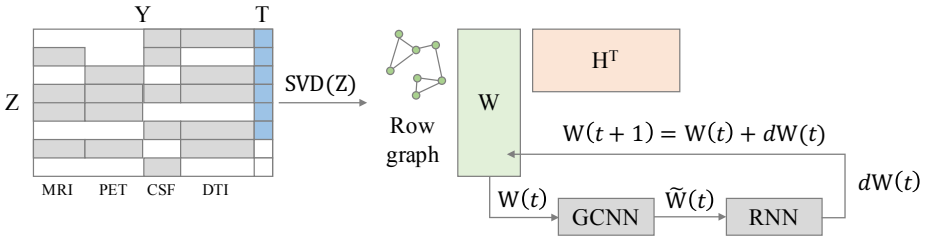


Fig. 1. Illustration of the overall approach: the matrix Z comprising incomplete features and labels is factorized into $Z = WH^T$. A connectivity graph is defined over rows W. During optimization, GCNN filters are learned along with RNN parameters and weight updates for W, towards optimal matrix completion of Z and simultaneous inference of missing features and labels in the dataset.

2.3 Matrix Completion on Graphs

The previous matrix completion problem can be extended to graphs [6, 13]. Given a matrix $\mathbf{Y}$, we can assume that the rows/columns of this matrix are on the vertices of the graph [6]. This additional information can then be included into the matrix completion formulation in equation (2) as a regularization term [6]. To construct the graph, we can use meta-information out of these rows/columns or use the row/column vectors of this matrix to calculate a similarity metric between pairs of vertices. Given that every row in the matrix has this meta-information, Kalofolias et al. [6] showed that we can build an undirected weighted row graph $G_r = (V_r, E_r, A_r)$, with vertices $V_r = \{1, ..., m\}$. Edges $E_r \subseteq V_r \times V_r$ are weighted with non-negative weights represented by an adjacency matrix $A_r \in \mathbb{R}^{m \times m}$. The column graph $G_c = (V_c, E_c, A_c)$ is built the same way as the row graph, where the columns are now the vertices in G_c. Kalofolias et al. [6] showed that the solution to this problem is equivalent to adding the Dirichlet norms, $||\mathbf{X}||^2_{D,r} = \text{tr}(X^T L_r X)$ and $||\mathbf{X}||^2_{D,c} = \text{tr}(X L_c X^T)$, where L_r and L_c are the unnormalized row and column graph Laplacian, to equation (2),

$$\min_{\mathbf{X} \in \mathbb{R}^{m \times n}} ||\mathbf{X}||_* + \frac{\gamma}{2}||\mathbf{\Omega} \circ (\mathbf{Y} - \mathbf{X})||^2_{\text{F}} + \frac{\alpha_r}{2}||\mathbf{X}||^2_{D,r} + \frac{\alpha_c}{2}||\mathbf{X}||^2_{D,c} \tag{4}$$

The factorized formulation [9, 13] of equation (4) is

$$\min_{\mathbf{W},\mathbf{H}} \frac{1}{2}||\mathbf{W}||^2_{\text{D,r}} + \frac{1}{2}||\mathbf{H}||^2_{\text{D,c}} + \frac{\gamma}{2}||\mathbf{\Omega} \circ (\mathbf{Y} - \mathbf{W}\mathbf{H}^T)||^2_{\text{F}} \tag{5}$$

2.4 Geometric Matrix Completion with Separable Recurrent Graph Neural Networks

In [9], Monti et al. propose to solve the matrix completion problem as a learnable diffusion process using Graph Convolutional Neural Networks (GCNN) and Recurrent Neural Networks (RNN). The main idea here is to use GCNN to extract

features from the matrix and then use LSTMs to learn the diffusion process. They argue that combining these two methods allows the network to predict accurate small changes $\mathbf{dX}$ (or $\mathbf{dW}$, $\mathbf{dH}$ of the matrices $\mathbf{W}$, $\mathbf{H}$) to the matrix $\mathbf{X}$. Further details regarding the main ideas in geometric deep learning have been summarized in a review paper [1], where they elaborate how to extend convolutional neural networks to graphs. Following [9], we use Chebyshev polynomial basis on the factorized form of the matrix $\mathbf{X} = \mathbf{W}\mathbf{H}^{\mathrm{T}}$ to represent the filters on the respective graph to each matrix $\mathbf{W}$ and $\mathbf{H}$. In this work, we only apply GCNN to the matrix $\mathbf{W}$ as we only have a row graph and leave the matrix $\mathbf{H}$ as a changeable variable. Figure 1 illustrates the overall approach.

2.5 Geometric Matrix Completion for Heterogeneous Matrix Entries

In this work, we propose to solve multi-modal disease classification as a geometric matrix completion problem. We use a Separable Recurrent GCNN (sRGCNN) [9] to simultaneously predict the disease and impute missing features on a dataset which has partially observed features and labels. Following Goldberg et al. [5], we stack a feature matrix $\mathbf{Y} \in \mathbb{R}^{m \times n}$ and a label matrix $\mathbf{T} \in \mathbb{R}^{m \times c}$ as a matrix $\mathbf{Z} \in \mathbb{R}^{m \times n+c}$, where m is the number of subjects, n is the dimension of the feature matrix, and c is the dimension of the target values. In the TADPOLE dataset, we stack the $m \times n$ feature matrix to the $m \times 1$ label matrix, where the feature matrix contains all the numerical features and the label matrix contains the encoded binary class labels for cMCI and sMCI. We build the graph by using meta-information from the patients such as their age and gender, similar to [12], as these information are known to be risk factors for AD. We compare two row graph construction approaches, first from age and gender information using a similarity metric [12] and second from age information only, using Euclidian distance-based k-nearest neighbors. Every node in a graph corresponds to a row in the matrix $\mathbf{W}$, and the row values to its associated feature vector. Since we only have a row graph, we leave the matrix $\mathbf{H}$ to be updated during backpropagation. To run the geometric matrix completion method we use the loss:

$$\ell(\Theta) = \frac{\gamma_a}{2}||\mathbf{W}||^2_{\mathrm{D,r}} + \frac{\gamma_b}{2}||\mathbf{W}||^2_{\mathrm{F}} + \frac{\gamma_c}{2}||\mathbf{H}||^2_{\mathrm{F}} + \frac{\gamma_d}{2}||\mathbf{\Omega}_a \circ (\mathbf{Z} - \mathbf{W}\mathbf{H}^T)||^2_{\mathrm{F}} + \gamma_e(\ell_{\mathbf{\Omega}_b}(\mathbf{Z}, \mathbf{X})), \tag{6}$$

where Θ are the learnable parameters, where $\mathbf{Z}$ denotes the target matrix, $\mathbf{X}$ is the approximated matrix, $||.||^2_{D,r}$ denotes the Dirichlet norm on a normalized row graph Laplacian, $\mathbf{\Omega}_a$ denotes the masking on numerical features, $\mathbf{\Omega}_b$ is the masking on the classification labels, and ℓ is the binary cross-entropy.

3 Results

We evaluate our approach on multi-modal TADPOLE data (MRI, PET, CSF, DTI) to predict MCI-to-AD conversion and compare it to several other multi-

modal methods as baseline. We use a stratified 10-fold cross-validation strategy for all methods. Hyperparameters were optimized using Hyperopt [4], through nested cross-validation, targeting classification loss (binary cross-entropy) on a hold-out validation set (10% in each fold of training data). Following [9], we use the same sRGCNN architecture with parameters: rank=156, chebyshev polynomial order=18, learning rate=0.00089, hidden-units=36, γ_a=563.39, γ_b=248.91, γ_c=688.85, γ_d=97.63, and γ_e=890.14.

It is noteworthy that at baseline, the data matrix Y with above-mentioned features is already feature-incomplete, i.e. only 53% filled. We additionally reduce the amount of available data randomly to 40%, 30% etc. to 5%. Figure 2 shows a comprehensive summary of our classification results in terms of area-under-the-curve (AUC). Methods we compare include mean imputation with random forest (RF), linear SVM (SVC) and multi-layer-perceptron (MLP), as well as three reference methods from literature [10, 12, 15], which operated on slightly different selections of ADNI subjects and on all available multi-modal features. While implementations of [10, 15] are not publicly available, we tried to re-evaluate the method [12] using their published code. Unfortunately, despite our best efforts and hyperparameter optimization on our selection of TADPOLE data, we were not able to reproduce any AUC value close to their published value. To avoid any mistake on our side, we provide the reported AUC results rather than the worse results from our own experiments.

At baseline, our best-performing method with a graph setup based on age and gender ("GMC age-gender") [12] achieves classification with an AUC value of 0.950, compared to 0.902 [10], $\sim$ 0.87 [12] and 0.851 [15]. In terms of classification

[4] http://hyperopt.github.io/hyperopt/

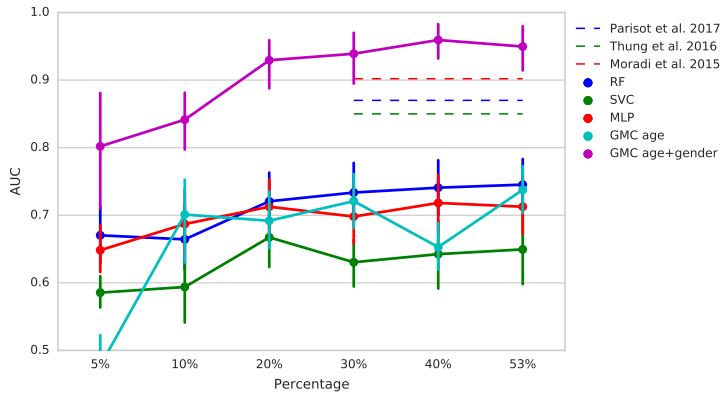


Fig. 2. Classification results: Area under the curve (AUC) of our method, for different amounts of feature-completeness and in comparison to linear/non-linear standard methods, and three state-of-the-art results in literature (Parisot et al. [12], Thung et al. [15], Moradi et al. [10]).

accuracy, we achieved a value of 87%, compared to 82% [10] and 77% [12] (accuracy not reported in [15]). Furthermore, our method significantly outperforms standard classifiers RF, MLP and SVC at all levels of matrix completeness. The second graph configuration for our method ("GMC age" only) performs significantly worse and less stable than ("GMC age-gender"), confirming the usefulness of the row graph construction based on the subject-to-subject similarity measure proposed in [12]. Due to lower complexity of the GMC approach [9], training a single fold on recent hardware (Tensorflow on Nvidia GTX 1080 Ti) is on average 2x faster (11.8s) than GCN (25.9s) [12].

4 Discussion and Conclusion

In this paper, we proposed to view disease classification in multi-modal but incomplete clinical datasets as a geometric matrix completion problem. As an exemplary dataset and classification problem, we chose MCI-to-AD prediction. Our initial results using this method show that GMC outperforms three competitive results from recent literature in terms of AUC and accuracy. At all levels of additional random dropout of features, GMC also outperforms standard imputation and classifiers (linear and non-linear). There are several limitations which are worthy to be addressed. Results in Figure 2 demonstrate that GMC is still sensitive to increasing amounts of feature incompleteness, in particular at feature presence below 15%. This may be due to our primary objective of disease classification during hyper-parameter optimization. For the same reason, we did not evaluate the actual imputation performed by GMC. However, an evaluation in terms of RMSE and a comparison to principled imputation methods [4] would be highly interesting, if this loss is somehow incorporated during hyperparameter optimization. Furthermore, we only evaluated GMC on ADNI data as represented in the TADPOLE challenge, due to the availability of multiple reference AUC/accuracy values in literature. As mentioned, however, disease classification in high-dimensional but incomplete datasets with multiple modalities is an abundant problem in computer-aided medical diagnosis. In this light, we believe that the promising results obtained through GMC in this study are of high interest to the community.

Acknowledgments. The study was supported by the German Federal Ministry of Education and Health (BMBF) in connection with the foundation of the German Center for Vertigo and Balance Disorders (DSGZ) (grant number 01 EO 0901).

Bibliography

[1] Michael M. Bronstein, Joan Bruna, Yann Lecun, Arthur Szlam, and Pierre Vandergheynst. Geometric Deep Learning: Going beyond Euclidean data. *IEEE Signal Processing Magazine*, 34(4):18–42, 2017.

[2] E.J. Candes and B. Recht. Exact low-rank matrix completion via convex optimization. *46th Annual Allerton Conference on Communication, Control, and Computing*, pages 1–49, 2008.

[3] Emmanuel J. Candes and Yaniv Plan. Matrix completion with noise. *Proceedings of the IEEE*, 98(6):925–936, 2010.

[4] Y. Dong and C. Y. Peng. Principled missing data methods for researchers. *Springerplus*, 2(1):222, Dec 2013.

[5] Andrew Goldberg, Ben Recht, Junming Xu, Robert Nowak, and Xiaojin Zhu. Transduction with matrix completion: Three birds with one stone. In *Advances in Neural Information Processing Systems (NIPS)*, pages 757–765. 2010.

[6] Vassilis Kalofolias, Xavier Bresson, Michael Bronstein, and Pierre Vandergheynst. Matrix completion on graphs. *arXiv:1408.1717*, 2014.

[7] Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. *CoRR*, arXiv:1609.02907, 2016.

[8] Geert Litjens, Thijs Kooi, Babak Ehteshami Bejnordi, Arnaud Arindra Adiyoso Setio, Francesco Ciompi, Mohsen Ghafoorian, Jeroen A.W.M. van der Laak, Bram van Ginneken, and Clara I. Snchez. A survey on deep learning in medical image analysis. *Medical Image Analysis*, 42:60 – 88, 2017.

[9] Federico Monti, Michael M. Bronstein, and Xavier Bresson. Geometric matrix completion with recurrent multi-graph neural networks. *CoRR*, arXiv:1704.06803, 2017.

[10] Elaheh Moradi, Antonietta Pepe, Christian Gaser, Heikki Huttunen, and Jussi Tohka. Machine learning framework for early MRI-based Alzheimer's conversion prediction in MCI subjects. *NeuroImage*, 104:398–412, 2015.

[11] Kenichi Oishi, Kazi Akhter, Michelle Mielke, Can Ceritoglu, Jiangyang Zhang, Hangyi Jiang, Xin Li, Laurent Younes, Michael Miller, Peter van Zijl, Marilyn Albert, Constantine Lyketsos, and Susumu Mori. Multi-Modal MRI Analysis with Disease-Specific Spatial Filtering: Initial Testing to Predict Mild Cognitive Impairment Patients Who Convert to Alzheimers Disease. *Frontiers in Neurology*, 2:54, 2011.

[12] Sarah Parisot, Sofia Ira Ktena, Enzo Ferrante, Matthew Lee, Ricardo Guerrerro Moreno, Ben Glocker, and Daniel Rueckert. Spectral graph convolutions for population-based disease prediction. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 177–185, 2017.

[13] Nikhil Rao, Hsiang-Fu Yu, Pradeep Ravikumar, and Inderjit S Dhillon. Collaborative Filtering with Graph Information: Consistency and Scalable Methods. *Neural Information Processing Systems (NIPS)*, pages 1–9, 2015.

[14] Nathan Srebro, Jason D M Rennie, and Tommi S Jaakkola. Maximum-Margin Matrix Factorization. *Advances in Neural Information Processing Systems (NIPS)*, 17:1329–1336, 2005.

[15] Kim-Han Thung, Ehsan Adeli, Pew-Thian Yap, and Dinggang Shen. Stability-weighted matrix completion of incomplete multi-modal data for disease diagnosis. In *International Conference on Medical Image Computing and Computer-Assisted Intervention*, pages 88–96, 2016.

[16] Daoqiang Zhang, Yaping Wang, Luping Zhou, Hong Yuan, and Dinggang Shen. Multimodal classification of alzheimer's disease and mild cognitive impairment. *NeuroImage*, 55(3):856 – 867, 2011.

BrainParcel: A Brain Parcellation Algorithm for Cognitive State Classification

Hazal Mogultay[1] and Fatos Tunay Yarman Vural[1]

Department of Computer Engineering, Middle East Technical University, Ankara, Turkey
{hazal,vural}@ceng.metu.edu.tr

Abstract. In this study, we propose a novel brain parcellation algorithm, called BrainParcel. BrainParcel defines a set of supervoxels by partitioning a voxel level brain graph into a number of subgraphs, which are assumed to represent "homogeneous" brain regions with respect to a predefined criteria. Aforementioned brain graph is constructed by a set of local meshes, called mesh networks. Then, the supervoxels are obtained using a graph partitioning algorithm. The supervoxels form partitions of brain as an alternative to anatomical regions (AAL). Compared to AAL, supervoxels gather functionally and spatially close voxels. This study shows that BrainParcel can achieve higher accuracies in cognitive state classification compared to AAL. It has a better representation power compared to similar brain segmentation methods, reported the literature.

Keywords: fMRI · Brain Partitioning · Mesh Model.

1 Introduction

Functional Magnetic Resonance Imaging (fMRI) is one of the most common imaging techniques for detecting the activation levels of human brain, during a cognitive process. fMRI measures the change of oxygen level in the brain with respect to neural activities. In principle, oxygen dependencies of neuron groups fluctuate in accordance with the activation and MRI machines can detect those changes through the scan. An intensity value is recorded at each 1-2 seconds for a neuron group called voxel. Each voxel is a cubic volume element around 1-$2mm^3$ size. Classification of the cognitive stimulus from the voxel intensity values are called brain decoding and the pioneering studies in this area are called Multi Voxel Pattern Analysis (MVPA) [11, 9]. MVPA involves recognizing the cognitive states represented by voxel intensity values of fMRI data, using machine learning techniques. A set of features is extracted from voxel intensity values recorded during each cognitive task. However, due to the large feature space formed by voxels (about 100,000-200,000 voxels per brain volume), dimension reduction techniques are required, such as, clustering the voxels groups into homogeneous regions.

Anatomical regions, defined by experimental neuroscience can be used as brain parcels. In most common approach, called AAL, there are 116 major regions and each region is assumed to contain voxel groups which work together. In order to reduce the dimension of the feature space, representative signals can be selected for each region or average time series can be computed within each region [1, 16]. However, anatomical regions lose the subject-specific and task dependent information of brain activities. Besides, sizes of the regions vary extremely and activation levels of voxels may not be homogeneous within an anatomic region.

In order to partition the voxels into a set of homogeneous regions, well-defined clustering methods such as k-means [6, 7, 10], hierarchical clustering [1, 4], and spectral clustering [17, 20] can be used. The pros and cons of these clustering algorithms are widely studied in fMRI literature on a variety of datasets [18, 8]. Some studies bring spatially close voxels together considering only the location information in analogy with the AAL [6]. Although this method improves the strict norms of AALs, it lacks the functional similarity of voxel time series, which belongs to the same regions. Recent literature reveals that functionally close voxels tend to contribute to the same cognitive task, thus, form homogeneous regions. Therefore, one needs to bring both functionally similar and spatially contiguous voxels together to define homogeneous brain regions [21]. Similarly, Wang et. al. suggest to combine n-cut segmentation algorithm with simple linear iterative clustering (SLIC) [21]. Blumensath et. al. use region growing for brain segmentation with functional metrics and spatial constraints between samples [3]. Bellec et. al. also use region growing with functional metrics within the 26 spatial neighborhood in 3-Dimensional space[2]. Background on neuronal activity, also, supports this idea, such that physically close neurons are in chemical interaction with each other and this interaction can be interpreted as functional similarity. With these objectives in mind, many different clustering algorithms are applied to create data dependent homogeneous brain parcels. Depending on the predefined distance measure, the clustering algorithms can group spatially or functionally similar voxels under the same cluster. Craddock et. al. adopt this idea and propose a brain parcellation method, in which they represent the voxels in a graph structure and used n-cut on a spatially constrained brain graph with functional edges [5]. In order to achieve spatial contiguity they connected each voxel to its 26 closest neighbors in 3D space. On the other hand, to accomplish functional homogeneity, they set edge weights of the graph to the correlation between the time series of two voxels as follows;

$$e_{i,j} = \begin{cases} corr(\mathbf{v_i}, \mathbf{v_j}) & , dist(\mathbf{v_i}, \mathbf{v_j}) \leq d_t \\ 0 & , otherwise, \end{cases} \tag{1}$$

where d_t is selected to be$\sqrt{3}$ and $corr(\mathbf{v_i}, \mathbf{v_j})$ is the Pearson Correlation between the intensity values of voxels $\mathbf{v_i}$ and $\mathbf{v_j}$. They, also, remove the edges with correlation values less than 0.5 to reduce the weak connections. Then, they define a brain graph $G = (V, E)$, where the set of voxels $V = [\mathbf{v_1}, \mathbf{v_2} \dots \mathbf{v_N}]$ are the nodes of the graph, and $E = [e_{1,1}, e_{1,2}, \dots e_{N,N}]$ are the edge weights computed

according to Equation 1. They partition the graph G into subgraphs by removing the edges iteratively using N-cut segmentation method using the following formula,

$$N\_cut = \frac{\sum_{\mathbf{v_i}\in A,\mathbf{v_j}\in B} e_{i,j}}{\sum_{\mathbf{v_i}\in A,\mathbf{v_n}\in V} e_{i,n}} + \frac{\sum_{\mathbf{v_i}\in A,\mathbf{v_j}\in B} e_{i,j}}{\sum_{\mathbf{v_j}\in B,\mathbf{v_n}\in V} e_{j,n}}. \tag{2}$$

As it is mentioned above, conventional MVPA methods create features sets from the selected voxel intensity values or use some averaging techniques to represent each brain region. This approach is quite restrictive to represent cognitive states. Recent studies suggest to model the relationships among voxels rather than using voxel intensity values. Ozay et. al, demonstrate this idea by suggesting the Mesh Model which is a graph structure that identifies the connectivity among voxels [15].

Mesh Model (MM) represents intensity values of voxels as a weighted linear combination of its neighboring voxels, defined on a neighborhood system. The estimated weights represent the arc weights between the voxels and the voxels represents a node in the overall brain graph. A star mesh is formed around each voxel and its p neighbors, independently. In each mesh, the voxel at the center is called seed-voxel and the surrounding voxels are called neighbors. p nearest neighbors of voxel $\mathbf{v_i}$ for cognitive stimulus k are shown as $\eta^p_{\mathbf{v_i}(k)}$ and they can be selected spatially (Spatial Mesh Model - SMM) [12, 14] or functionally (Functional Mesh Model - FMM) [12, 13] such that, spatial neighbors has the smallest Euclidean distance with the seed-voxel whereas functional neighbors has maximum functional similarity. Meshes are formed using the full length time series for voxels, recorded during an fMRI experiment session. Assuming s measurements are taken for each cognitive stimulus, time series of a voxel $\mathbf{v_i}$ for stimulus k is an s dimensional vector shown as $\mathbf{v_i}(k) = [\mathbf{v_i}(k)^1, \mathbf{v_i}(k)^2, \dots \mathbf{v_i}(k)^s]$. Spatial Mesh Model (SMM) selects the neighbors according to the physical distances among voxels in 3-dimensional space by Euclidean distance [12, 14]. On the other hand, Functional Mesh Model (FMM), proposed by Onal et. al., selects functional neighbors with the highest p-correlation values obtained by Pearson Correlation [12, 13]. Afterwards, time series of the seed voxel is represented as a weighted combination of its neighbors by the following equation for each cognitive stimuli:

$$\mathbf{v_i}(k) = \sum_{\mathbf{v_j}(k)\in\eta^p_{\mathbf{v_i}(k)}} a_{i,j,k}\mathbf{v_j}(k) + \epsilon_{i,j}, \tag{3}$$

where $\eta^p_{v_i(k)}$ is the p nearest neighbors of voxel v_i for sample k and $a_{i,j,k}$ are the arc weights of the mesh network between the voxels and they are called Mesh Arc Descriptors (MADs). MADs are estimated using regularized Ridge regression method by the minimization of error term $\epsilon_{i,j}$. Concatenating each MAD for each voxel and cognitive task creates a new feature space and classification is performed on this feature space.

In this study we combine classical brain parcellation approach proposed by Craddock et. al. and Mesh Model and propose a novel brain parcellation al-

gorithm, called BrainParcel. Unlike current methods, we partition the graph formed by star meshes and partition this graph into brain regions. We show that brain partitions obtained by BrainParcel have better representation power than the partitions obtained by state of the art clsutering methods and AAL in cognitive state classification problem.

Fig. 1: Overall architecture of the BrainPacel algorithm.

2 BrainParcel

Brain parcel is a brain partitioning algorithm that uses graph theoretic approaches. First, we form a brain graph by ensembling the meshes estimated around each voxel . Then, we partition this graph using n-cut segmentation algorithm. Each region is represented by the average time series of all voxels in that region. Then, these representative time series are fed to a machine learning algorithm to classify the underlying cognitive states. Figure 1 indicates the stages of suggested BrainParcel method for brain decoding problem. Each stage is explained in the following subsections.

2.1 Neighborhood System

In order to estimate a star mesh around each voxel independently, we need to define a neighborhood system. The concept of neighborhood takes an important place in this study. We inspire from the biological structure of human brain, where spatially close neurons act together by means of some electro-chemical interactions. Additionally, experimental evidence indicates that physically far apart neurons may contribute to the same cognitive process through the brain connectome. We try to utilize these observations in our brain parcellation model by defining a neighborhood system around each voxel and employ multiple connections between the neighboring voxels.

Neighborhood of the i^{th} voxel $\mathbf{v_i}$, is defined as the set of voxels that are closest to $\mathbf{v_i}$ according to a predefined rule. Assuming p_c many neighbors around a voxel, neighborhood of $\mathbf{v_i}$ is represented by $\eta_{v_i}^{p_c}$.

Letting N be the number of voxels, we define an $N-by-N$ adjacency matrix, ND, to represent the neighborhood of voxels. Each entry of ND is calculated as follows;

$$ND(i,j) = \begin{cases} 1 & , v_j \in \eta_{v_i}^{p_c} \\ 0 & , otherwise. \end{cases} \tag{4}$$

In this study, we define two types of neighborhood, given below:

Spatial neighborhood $\eta_{v_i}^{p_c}$ is defined as the set of voxels, which has the p_c smallest Euclidean distance in 3-Dimensional space to voxel $\mathbf{v_i}$. This neighborhood system ensures resulting brain parcels to be spatially contiguous.

Functional neighborhood $\eta_{v_i}^{p_c}$ is defined as the set of voxels, which has the highest p_c-Pearson Correlation to voxel $\mathbf{v_i}$. This neighborhood system connects functionally similar voxels, even if they are physically apart from each other.

Note that, selection of the number of neighbors, p_c, and the type of the neighborhood system highly effects the rest of the steps of BrainParcel. Specifically, functional neighborhood relaxes the spatial similarity, selecting the neighboring voxels which are physically far apart. Therefore, the resulting brain parcels are not guaranteed to be spatially contiguous. It is very crucial to define a sort of balance in these two types of neighborhood, so that the resulting brain parcels consist of functionally similar and spatially contiguous voxels.

2.2 Extracting Mesh Arc Descriptors (MADs) Among Voxels

Each voxel is connected to its neighboring voxels according to one of the above neighborhood systems to form a star mesh around that voxel. The structure of star mesh depends on the type of the neighborhood system defined above. The arc weights of each local mesh are estimated by adopting the mesh model of Onal et. al. [12–14]. As opposed to the current studies, we form the meshes, based on the complete time series recorded at each voxel rather than forming a different mesh for each cognitive task. This approach enables us to form a shared brain partition across all of the cognitive tasks

fMRI technique collects a time series for each voxel, when the subject is exposed to a cognitive stimulus. In the case of a block experiment design, which we have used, subjects are exposed to a stimulus for a specific time interval and the voxel time series over the entire brain volume are collected. Then, after a rest period, another stimulus is given to the subject. The time series recorded during a stimulus at i^{th} voxel is represented by the vector $\mathbf{v_i}$. Based on the idea of mesh model, we represent each $\mathbf{v_i}$ as weighted sum of other voxels in the $\eta_{v_i}^{p_c}$ neighborhood of $\mathbf{v_i}$ according to Equation 3. Notice that Mesh Arc Descriptors (MADs) for classification are calculated per cognitive stimulus. However, we compute MADs from the entire time series of the voxels. Therefore, k index, which indicates a specific cognitive task, is removed from Equation 3, since we compute MADs for the entire time duration of fMRI recordings. This representation is carried with a linear equation by the following formula;

$$\mathbf{v_i} = \sum_{v_j \in \eta_{v_i}^{p}} a_{i,j}\mathbf{v_j} + \epsilon_{i,j}. \tag{5}$$

Weights of the representation, called Mesh Arc Descriptors (MADs) are shown as $a_{i,j}$ and are estimated by Regularized Ridge Regression which minimizes the mean squared error $\epsilon_{i,j}^2$ [12, 14, 13].

2.3 Constructing a Voxel-Level Brain Graph

In order to construct a brain graph from the estimated MADs, we ensemble all the local meshes under the same graph, $G_m = (V, E_m)$. The set of nodes of this graph correspond the set of voxels $V = [\mathbf{v_1}, \mathbf{v_2}, \dots \mathbf{v_N}]$. The set of edges corresponds to set of all MADs, $a_{i,j} \epsilon E_m$. Note that, since $a_{i,j} \neq a_{j,i}$, the graph G_m is directed. On the other hand, the graph partitioning methods, such as n-cut requires undirected graphs, in which each edge weight, $e_{i,j}$ is represented by a scalar number. In order to obtain an undirected graph from the directed graph G_m, a set of heuristic rules are used. Suppose that the mesh is formed for the voxel $\mathbf{v_i}$, and $\mathbf{v_j}$ is in the neighborhood of $\mathbf{v_i}$ with mesh arc-weight $a_{i,j}$. Edge value $e_{i,j}$ is determined, based on the following rules:

- **Case 1:** IF $\mathbf{v_i} \notin \eta_{v_j}^{p_c}$ AND $\mathbf{v_j} \in \eta_{v_i}^{p_c}$ THEN $e_{i,j} = a_{i,j}$
- **Case 2:** IF $\mathbf{v_i} \in \eta_{v_j}^{p_c}$ AND $\mathbf{v_j} \in \eta_{v_i}^{p_c}$ THEN this case requires further analysis. Assuming highly correlated voxels should have a stronger edge between them, we employ the following thresholding method;
  IF $corr(v_i, v_j) \geq 0$, THEN $e_{i,j} = max(a_{i,j}, a_{j,i})$
  IF $corr(v_i, v_j) < 0$, THEN $e_{i,j} = min(a_{i,j}, a_{j,i})$.

Above rules prune the directed graph G_m to an undirected graph G to be partitioned for obtaining homogeneous brain regions, called supervoxels.

2.4 Graph Partitioning for Obtaining Supervoxels

After constituting the brain graph G, n-cut segmentation method is used for clustering this graph. N-cut is a graph partitioning algorithm which carries a graph cut method on a given undirected graph. Given G, n-cut cuts the edges one by one in an iterative manner. With each cut, the graph is split into two smaller connected components. Letting N be the number of voxels, n-cut method requires the representative graph G, which is actually an $N - by - N$ adjacency matrix explained in the previous sections. The number of intended brain parcels is set to $\mathbb{C}$. With graph cut operations, graph is split into C connected components where $C \leq \mathbb{C}$. Each sample is a member of one of this clusters and assigned with a cluster index. In other words, n-cut method returns an $1 - by - N$ dimensional vector $\mathbb{L}_{\mathbb{C}} = [l_1^c, l_2^c, \dots, l_N^c]$ where each l_i^c is a number between 1 and C. The n-cut method, as applied to undirected graph G is called BrainParcel. The output of this algorithm yields a set of supervoxels, which are homogeneous with respect to the subgraphs of mesh network.

Recall that, anatomical regions (AAL) produce an experimentally neuroscientific parcellation of the brain. In order to compare the brain decoding performances, we conducted our experiment, where we form mesh network for both among anatomical regions and the network formed among supervoxels obtained at the output of BrainParcel. There are 116 basic brain regions in AAL and each voxel resides in one and only one region. Let us represent the anatomically defined region indices of voxels with $\mathbb{L}_{\mathbb{A}} = [l_1^a, l_2^a, \dots, l_N^a]$, in order to avoid confusions. Notice that, with $\mathbb{L}_{\mathbb{A}}$ we skip all of the brain parcellation steps. Also,

let us call $\mathbb{L} = [l_1, l_2, \ldots, l_N]$ to all kinds of brain segmentations; in our case it means $\mathbb{L} \supset (\mathbb{L}_{\mathbb{C}} \cap \mathbb{L}_{\mathbb{A}})$.

2.5 Representation of Supervoxels

We need to calculate a representative signal for each supervoxel. For this purpose, we take an average among the time series of voxels within each supervoxel. With C supervoxels, we calculate set of vectors $U = [\mathbf{u_1}, \mathbf{u_2}, \ldots, \mathbf{u_C}]$, where each u_i is the representative vector of supervoxel i and they are calculated as follows;

$$\mathbf{u_i} = \frac{\sum_{l_j==i} \mathbf{v_j}}{\sum_{l_j==i} 1}. \tag{6}$$

In the dataset on which we have performed our experiments, six measurements were taken for each cognitive stimulus. Assuming K stimuli were shown to each subject, time series of each voxel has a length $\mathbb{K} = 6K$. Therefore, at the output of the clustering algorithm we construct a data matrix U of size $C - by - \mathbb{K}$, where each row represents a feature, and each column corresponds to a cognitive stimulus.

2.6 Constructing Supervoxel-Level Brain Graph

The original area of utilization of the mesh model was to model the relationships among voxels and use this relationship for decoding the cognitive processes. Both spatial and functional neighborhoods were considered, and their representation powers were demonstrated by relatively high recognition performances compared to the available state of the art network models. Specifically, Functional Mesh Model (FMM) outperform most of the MVPA and Spatial Mesh Model (SMM) results. Therefore, we use FMM for classifying the cognitive states.

Data matrix U, defined in the previous section, is feed into the FMM algorithm. Each supervoxel $\mathbf{u_i}$ is represented by linear combination of its functional neighbors, the arc weights are estimated at each mesh using Ridge Regression for each cognitive stimulus. Recall that fMRI collects multiple measurements during the time course of each cognitive stimulus. In our dataset 6 measurements are collected for each stimulus, and $\mathbf{u_i}$ is a vector of length $\mathbb{K} = 6K$ for K stimuli. Let us represent the vector of the stimulus k by $\mathbf{u_i(k)}$. First, functionally closest p_m neighbors of $\mathbf{u_i(k)}$; $\eta^{p_m}_{u_i(k)}$, are selected from the supervoxels $\mathbf{u_j}$ which has the highest correlation with supervoxel $\mathbf{u_i}$ according to Pearson Correlation. Then, the mean square error $E(\epsilon^2_{i,j})$ is minimized to estimate $a_{i,j,k}$ of the Equation 3. Estimated MADs are concatenated so that they represent a more powerful feature space compared to the raw fMRI signal intensity values that is used in MVPA studies. We concatenate all the MADs and represent the stimulus in a feature space formed by MADs.

2.7 Classification

MADs estimated at supervoxel-level, are concatenated under a feature vector for classifying the cognitive states. 6 fold cross validation schema is applied on the dataset, where at each fold, one run is reserved from the data as a test set. Logistic regression is used for classification.

3 Experiments

3.1 Dataset

In this study, we use a dataset called "Object Experiment". This dataset is recorded by the team of ImageLab of METU members at Bilkent University UMRAM Center. It consists of 4 subjects in the age of twenties. Each subject is shown various bird and flower pictures. In between those stimuli, simple mathematics questions are shown as transition. There are 6 runs in the experiment and in each run, 36 pictures are shown to each subject. Thus, there are total of 216 samples. Number of samples are balanced for the two classes (bird and flower). Preprocessing of the dataset is carried with the SPM toolbox and the number of voxels is decreased to 20,000 for each subject. Also, there are 116 labeled anatomical regions, defined under MNI coordinate system [19]. We provide experimental results, where each given accuracy is the output of a six fold cross validation. Recall that, each subject is given 6 runs of stimuli. At each fold, we split a run for testing and use the other 5 runs for training. The reported accuracies are the average of these 6 folds for each subject.

3.2 Comparative Results

In this section, we provide a comparison between BrainParcel and the parcellation algorithm suggested by Craddock et. al. Table 1 shows the classification performances for various number of parcels. The results are reported after optimizing the mesh sizes empirically. Recall that, functionally constrained systems that construct the graph with $Functional\_ND$ neighborhood system does not provide any spatial integrity within the brain parcels, since the brain graph is not formed on these grounds. On the other hand, spatially constrained systems provide both spatial continuity and functional homogeneity since the brain graph is formed by spatial restrictions and edges are weighted in terms of functional connectivity.

In Table 1, the first and third column give the best results for the brain parcellation method suggested by Craddock et. al. (called classical, in the Table), and the other two gives the results obtained by BrainParcel that we have proposed. Each row of this table gives the results for a different number of supervoxels (SV). Notice that spatially constrained BrainParcel gives the best classification performances in the overall schema.

These results point to the idea that, in order to achieve better representational power for cognitive state classification, one needs spatially contiguous

and functionally homogeneous brain parcels, which is accomplished by spatial BrainParcel. Moreover, recall that we have offered BrainParcel as an alternative to AAL, which has 116 basic anatomic regions and gives 53% performance on average. A compatible parcellation scheme consists of 100 super voxels, where, Spatial BrainParcel results in higher classification accuracies compared to the other methods.

Table 1: Overall 2-class classification accuracy acquired from the MADs constructed among super voxels and method suggested by Craddock et. al. (called, classical). These results suggest that Spatial BrainParcel gives higher performances, since it provides spatial continuity and functional homogeneity within each brain parcel.

	Spatial Constraints		Functional Constraints	
# of SV	Classical	BrainParcel	Classical	BrainParcel
100	67.79	**74.00**	70.63	73.29
250	72.96	**78.71**	76.79	77.54
500	75.46	**79.42**	77.33	77.54
750	77.46	78.04	**79.38**	77.54
1000	78.08	78.83	79.96	**80.08**

4 Conclusion

In this study, we offer a brain parcellation methodology, which combines the spatial and functional connectivity of brain on a novel graph representation. This approach offers a better alternative to the current brain parcellation methods in the literature [8, 18], for brain decoding problems. BrainParcel uses spectral clustering methods, which represents the voxel space as a graph formed by mesh model. Common studies compute the edge weights of the brain graph as the pairwise correlation between voxels, whereas we computed the edge weights by estimating them using the mesh model among a group of voxels. Then, brain graph is partitioned with n-cut segmentation method to generate supervoxels.

As suggested, using task dependent brain parcellation methods enable better brain decoding performances compared to anatomical regions. Moreover, it is demonstrated that functional connectivity, united with the spatial contiguity is the best approach to represent homogeneous brain regions.

Also, results show that using the MADs of the mesh model for classification, improves the brain decoding performances in all of the experiment setups.

Our study reveals that mesh model not only improves the classification performance, but also creates a brain graph, where the nodes represent homogeneous super voxels with a better representation power for brain decoding. Although

the performance increase looks relatively small, when the large size of the data set is considered, the performance boost becomes quite meaningful.

In the future, experimental set up can be refined for parameter selection.

5 Acknowledgement

This project is supported by TUBITAK under grant number 116E091. We thank UMRAM Center of Bilkent University for opening their facilities to collect fMRI dataset. We also thank to Dr. Itir Onal Ertugrul and Dr. Orhan Firat for their contribution and effort of data collection .

References

1. Alkan, S., Yarman-Vural, F.T.: Ensembling brain regions for brain decoding. In: 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). pp. 2948–2951. IEEE (2015)
2. Bellec, P., Perlbarg, V., Jbabdi, S., Pélégrini-Issac, M., Anton, J.L., Doyon, J., Benali, H.: Identification of large-scale networks in the brain using fmri. Neuroimage **29**(4), 1231–1243 (2006)
3. Blumensath, T., Jbabdi, S., Glasser, M.F., Van Essen, D.C., Ugurbil, K., Behrens, T.E., Smith, S.M.: Spatially constrained hierarchical parcellation of the brain with resting-state fmri. Neuroimage **76**, 313–324 (2013)
4. Cordes, D., Haughton, V., Carew, J.D., Arfanakis, K., Maravilla, K.: Hierarchical clustering to measure connectivity in fmri resting-state data. Magnetic resonance imaging **20**(4), 305–317 (2002)
5. Craddock, R.C., James, G.A., Holtzheimer, P.E., Hu, X.P., Mayberg, H.S.: A whole brain fmri atlas generated via spatially constrained spectral clustering. Human brain mapping **33**(8), 1914–1928 (2012)
6. Flandin, G., Kherif, F., Pennec, X., Malandain, G., Ayache, N., Poline, J.B.: Improved detection sensitivity in functional mri data using a brain parcelling technique. Medical Image Computing and Computer-Assisted Intervention, MICCAI pp. 467–474 (2002)
7. Flandin, G., Kherif, F., Pennec, X., Riviere, D., Ayache, N., Poline, J.B.: Parcellation of brain images with anatomical and functional constraints for fmri data analysis pp. 907–910 (2002)
8. Liao, T.W.: Clustering of time series dataa survey. Pattern recognition **38**(11), 1857–1874 (2005)
9. Mitchell, T.M., Shinkareva, S.V., Carlson, A., Chang, K.M., Malave, V.L., Mason, R.A., Just, M.A.: Predicting human brain activity associated with the meanings of nouns. science **320**(5880), 1191–1195 (2008)
10. Moğultay, H., Alkan, S., Yarman-Vural, F.T.: Classification of fmri data by using clustering. In: 23th Signal Processing and Communications Applications Conference, SIU. pp. 2381–2383. IEEE (2015)
11. Norman, K.A., Polyn, S.M., Detre, G.J., Haxby, J.V.: Beyond mind-reading: multi-voxel pattern analysis of fmri data. Trends in cognitive sciences **10**(9), 424–430 (2006)

12. Onal, I., Ozay, M., Mizrak, E., Oztekin, I., Yarman-Vural, F.T.: A new representation of fmri signal by a set of local meshes for brain decoding. IEEE Transactions on Signal and Information Processing over Networks (2017). https://doi.org/10.1109/TSIPN.2017.2679491
13. Onal, I., Ozay, M., Yarman-Vural, F.T.: Functional mesh model with temporal measurements for brain decoding. In: 37th Annual International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC). pp. 2624–2628. IEEE (2015)
14. Onal, I., Ozay, M., Yarman-Vural, F.T.: Modeling voxel connectivity for brain decoding. In: International Workshop on Pattern Recognition in NeuroImaging (PRNI). pp. 5–8. IEEE (2015)
15. Ozay, M., Öztekin, I., Öztekin, U., Yarman-Vural, F.T.: Mesh learning for classifying cognitive processes. arXiv preprint arXiv:1205.2382 (2012)
16. Richiardi, J., Eryilmaz, H., Schwartz, S., Vuilleumier, P., Van De Ville, D.: Decoding brain states from fmri connectivity graphs. Neuroimage **56**(2), 616–626 (2011)
17. Shen, X., Papademetris, X., Constable, R.T.: Graph-theory based parcellation of functional subunits in the brain from resting-state fmri data. Neuroimage **50**(3), 1027–1035 (2010)
18. Thirion, B., Varoquaux, G., Dohmatob, E., Poline, J.B.: Which fmri clustering gives good brain parcellations? Frontiers in neuroscience **8** (2014)
19. Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., Joliot, M.: Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. Neuroimage **15**(1), 273–289 (2002)
20. Van Den Heuvel, M., Mandl, R., Pol, H.H.: Normalized cut group clustering of resting-state fmri data. PloS one **3**(4), e2001 (2008)
21. Wang, J., Wang, H.: A supervoxel-based method for groupwise whole brain parcellation with resting-state fmri data. Frontiers in human neuroscience **10** (2016)

Modeling Brain Networks with Artificial Neural Networks

Baran Baris Kivilcim[1], Itir Onal Ertugrul[2], and Fatos T. Yarman Vural[1]

[1] Department of Computer Engineering, Middle East Technical University, Ankara, Turkey
{baran.kivilcim,vural}@ceng.metu.edu.tr
[2] Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, USA
iertugru@andrew.cmu.edu

Abstract. In this study, we propose a neural network approach to capture the functional connectivities among anatomic brain regions. The suggested approach estimates a set of brain networks, each of which represents the connectivity patterns of a cognitive process. We employ two different architectures of neural networks to extract directed and undirected brain networks from functional Magnetic Resonance Imaging (fMRI) data. Then, we use the edge weights of the estimated brain networks to train a classifier, namely, Support Vector Machines(SVM) to label the underlying cognitive process. We compare our brain network models with popular models, which generate similar functional brain networks. We observe that both undirected and directed brain networks surpass the performances of the network models used in the fMRI literature. We also observe that directed brain networks offer more discriminative features compared to the undirected ones for recognizing the cognitive processes. The representation power of the suggested brain networks are tested in a task-fMRI dataset of Human Connectome Project and a Complex Problem Solving dataset.

Keywords: Brain Graph · Brain Decoding · Neural Networks

1 Introduction

Brain imaging techniques, such as, functional Magnetic Resonance Imaging (fMRI) have facilitated the researches to understand the functions of human brain using machine learning algorithms [20, 15, 25, 14]. In traditional approaches, such as Multi-Voxel Pattern Analysis (MVPA), the aim was to discriminate cognitive tasks from the fMRI data itself without forming brain graphs and considering relationship between nodes of graphs. Moreover, Independent Component Analysis (ICA) and Principal Component Analysis (PCA) have been applied to obtain better representations. In addition to feature extraction methods, General Linear Model (GLM) and Analysis of Variance (ANOVA) have been used to select important voxels [20]. None of these approaches take into account the massively connected network structure of the brain [26, 22, 4, 3, 12]. Recently, use of deep

learning algorithms have also emerged in several studies [8, 9, 7] to classify cognitive states. Most of these studies mainly focus on using deep learning methods to extract better representations from fMRI data for brain decoding.

Several studies form brain graphs using voxels or anatomical regions as nodes and estimate the edge weights of brain graphs with different approaches. Among them, Richiardi et al. [21] have created undirected functional connectivity graphs in different frequency subbands. They have employed Pearson correlation coefficient between responses obtained from all region pairs as edge weights and use these edge weights to perform classification in an audio-visual experiment. Brain graphs, constructed using pairwise correlations and mutual information as edge weights, have been used to investigate the differences in networks of healthy controls and patients with Schizophrenia [11] or Alzheimer's disease [13, 10]. Yet, these studies consider only pairwise relationships while estimating the edge weights and ignore the locality property of the brain.

Contrary to pairwise relationships, a number of studies have estimated the relationships among nodes within a local neighborhood. Ozay et. al. [19] and Firat et al. [6] have formed local meshes around nodes and constructed directed graphs as ensembles of local meshes. They have applied Levinson-Durbin recursion [24] to estimate the edge weights representing the linear relationship among voxels and have used these weights to classify the category of words in a working memory experiment. Similarly, Alchihabi et al. [2] have applied Levison-Durbin recursion to estimate the edge weights of local meshes of dynamic brain network for every brain volume in Complex Problem Solving task and have explored activation differences between sub-phases of problem solving. While these studies conserve the locality in the brain, construction of a graph for every time instant discards temporal relationship among nodes of the graph. Onal et al. [18, 17] have formed directed brain graphs as ensemble of local meshes. They have estimated the relationships among nodes within a time period considering the temporal information using ridge regression. Since the spatially neighboring voxels are usually correlated, linear independence assumption of features required for closed form solution to the estimation of linear relationship among voxels is violated. This may result in large errors and inadequate representation. Since the aforementioned studies form local meshes around each node separately, associativity is ignored in the resulting brain graphs.

In this study, we propose two brain network models, namely, directed and undirected Artificial Brain Networks to model the relationships among anatomical regions within a time interval using fMRI signals. In both network models, we train an artificial neural network to estimate the time series recorded at node which represent an anatomic region by using the rest of the time series recorded in the remaining nodes. In our first neural network architecture, called directed Artificial Brain Networks (dABN), global relationships among nodes are estimated without any constraint whereas in our second architecture of undirected Artificial Brain Networks (uABN), we apply a weight sharing mechanism to ensure undirected functional connections.

We test the validity of our dABN and uABN in two fMRI datasets and compare the classification performances to the other network models available in the literature. First, we employ the Human Connectome Project (HCP), task-fMRI (tfMRI) dataset, in which the participants were required to complete 7 different mental tasks. The second fMRI dataset contains fMRI scans of subjects solving Tower of London puzzle and has been used to study regional activations of Complex Problem Solving [16, 2]. The task recognition performances of the suggested Artificial Brain Networks are significantly greater than the ones obtained with state of the art functional connectivity methods.

2 Extraction of Artificial Brain Networks

In this section, we explain how we estimate the edge weights of directed and undirected brain networks using artificial neural networks. Throughout this study, we represent a brain network by $G = (V, E)$, where $V = \{v_1, v_2, v_3, \ldots, v_M\}$, denotes the vertices of the network, which represent $M = 90$ anatomical brain regions, $R = \{r_1, r_2, r_3, \ldots, r_M\}$. The attribute of each node is the average time series of BOLD activations. The average BOLD activation of an anatomical region r_i at time t is denoted with $b_{i,t}$. We use all anatomical regions defined by Anatomical Atlas Labeling (AAL) [23], except for the ones residing in Cerebellum and Vermis. We represent the edges of the brain network by $E = \{e_{i,j} | \forall v_i, v_j \in V, i \neq j\}$. The weights of edges depend on the estimation method. We denote the adjacency matrix which consists of the edge weights, as A, where $a_{i,j}$ represents the weight of edge from v_i to v_j, when the network is directed. When the network is undirected the weight of the edge formed between v_i and v_j is $a_{i,j} = a_{j,i}$. Sample representations of directed and undirected brain networks are shown in Fig. 1 and Fig. 2, respectively.

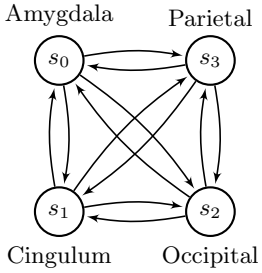


Fig. 1. A Directed Brain Network.

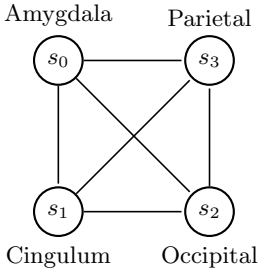


Fig. 2. An Undirected Brain Network.

We temporally partition the fMRI signal into chunks with length L recorded during each cognitive process. The fMRI time series at each chunk is used to estimate a network to represent the spatio-temporal relationship among anatomic regions. Then, the cognitive process k of subject s is described as a consecutive

list (T_k^s) of brain networks, formed for each chunk within time interval $[t, t+L]$, where $T_k^s = \{G_1, G_2, \ldots, G_{C_k}\}$. Note that, C_k is the number of chunks obtained for cognitive process k and equals to $\lfloor N_k/L \rfloor$, where N_k denotes the number of measurements recorded for cognitive process k. Since we obtain a different network for each duration of length L for a cognitive process of length N_k, this approach estimates a dynamic network for the cognitive process, assuming that N_k is sufficiently large.

For a given time interval $[t, t+L]$, weights of incoming edges to vertex v_i is defined by an M dimensional vector, $\bar{\mathbf{a}}_\mathbf{i} = [a_{i,1}, a_{i,2}...a_{i,M}]$. Note that the ith entry $a_{i,i} = 0$, which implies that a node does not have an edge value into itself. These edge weights define the linear dependency of activation, $b_{i,t}$, of region r_i at time t to the activations of the remaining regions, $b_{j,t}$ for a time interval $t' \in \{t, t+L\}$

$$b_{i,t'} = \sum_{j \neq i, j=1}^{M} a_{i,j} b_{j,t'} + \epsilon_{t'} = \hat{b}_{i,t'} + \epsilon_{t'} \qquad \forall t' \in \{t, t+L\} \tag{1}$$

where $\hat{b}_{i,t'}$ is the estimated value of $b_{i,t'}$ at time t' with error rate $\epsilon_{t'}$, which is the difference between the real and estimated activation. Note that each node is connected to the rest of $M-1$ nodes each of which corresponding to anatomic regions.

2.1 Directed Artificial Brain Networks (dABN)

In fully connected directed networks, we define two distinct edges between all pairs of vertices, $E = \{e_{i,j}, e_{j,i} | v_i, v_j \in V, i \neq j\}$ where $e_{i,j}$ denotes an edge from v_i to v_j. The weights of the edge pairs are not to be symmetrical, $a_{i,j} \neq a_{j,i}$.

The neural network we design to estimate edge weights consists of an input layer and an output layer. For every edge in the brain network, we have an equivalent weight in the neural network, such that weight between $input_i$ and $output_j$, $w_{i,j}$ is assumed to be an estimate for the weight , $a_{i,j}$ of the edge from v_i to v_j, in the artificial brain network.

We employ a regularization term λ to increase generalization capability of the model and minimize the expected value of sum of squared error through time. Loss of an output node $output_i$ is defined as,

$$Loss(output_i) = E((b_{i,t'} - \sum_{j \neq i, j=1}^{M} w_{i,j} b_{j,t'})^2) + \lambda \mathbf{w}_i^T \mathbf{w}_i, \tag{2}$$

where $w_{i,j}$ denotes the neural network weight between $input_i$ and $output_j$ and E(.) is the expectation operator taken over time interval [t,t+L]. For each training step of the neural network, e, gradient descent is applied for the optimization of the weights as in Equation (3) with empirically chosen learning rate, α. The whole system is trained for an empirically selected number of epochs.

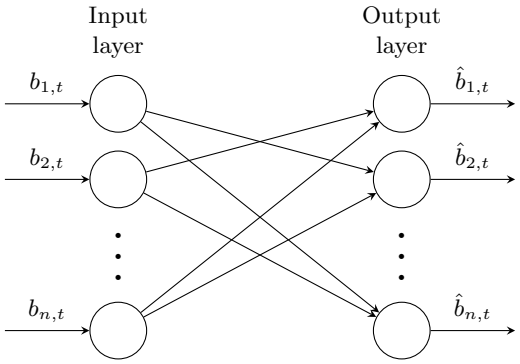


Fig. 3. Directed Artificial Brain Network Architecture.

$$w_{i,j}^{(e)} \leftarrow w_{ij}^{(e-1)} - \alpha \frac{\partial Loss(output_i)}{\partial w_{i,j}}. \tag{3}$$

After training, the weights of neural network are assigned to edge weights of the corresponding brain graph, $a_{i,j} \leftarrow w_{i,j}, \forall_{i,j}$.

2.2 Undirected Artificial Brain Network (uABN)

In undirected brain networks, similar to directed brain network, we define double connections between every pair of vertices $E = \{e_{i,j}, e_{j,i} | v_i, v_j \in V, i \neq j\}$. However, in order to make the network undirected, we must satisfy the constraint that twin (opposite) edges have the equal weights, $a_{i,j} = a_{j,i}$. In order to assure th's property in the neural network explained in the previous section, we use a weight sharing mechanism and keep the weights of the twin (opposite) edges in the neural network equal through the learning process, such that $w_{i,j} = w_{j,i}$. The proposed architecture is shown at Figure 4.

We use Equation (2) for undirected Artificial Brain Networks. The weight matrix of uABN is initialized symmetrically, $w_{i,j} = w_{j,i}$ and in order to satisfy the symmetry constraint through training epochs, we define the following update rule for the weights, $w_{i,j}$ and $w_{j,i}$ at epoch e.

$$w_{i,j}^{(e)} = w_{j,i}^{(e)} \leftarrow w_{i,j}^{(e-1)} - \frac{1}{2}\alpha \left[\frac{\partial Loss(output_i)}{\partial w_i, j} + \frac{\partial Loss(output_j)}{\partial w_{i,j}} \right]. \tag{4}$$

Again, after an empirically determined number of epochs, the weights of edges in the undirected graph is assigned to the neural network weights , $a_{i,j} \leftarrow w_{i,j}$.

2.3 Baseline Methods

In this subsection, we briefly describe the popular methods that have been used to build functional connectivity graphs, in order to provide some comparison for the suggested Artificial Brain Network.

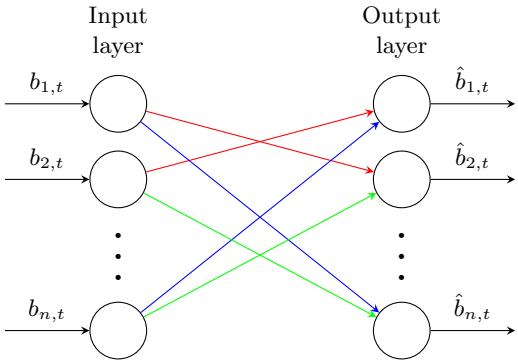


Fig. 4. Neural Network Structure to Create Undirected Artificial Brain Networks (connections with the same colors are shared).

Pearson Correlation In their work, Richiardi et al. [21] defined the functional connectivity between two anatomic regions as pair-wise Pearson correlation coefficients computed between the average activations of these regions in a time interval. The edge weights are calculated by,

$$\rho_{b_{i,t,L},b_{j,t,L}} = \frac{cov(\mathbf{b_{i,t,L}}, \mathbf{b_{j,t,L}})}{\sigma_{\mathbf{b_{i,t,L}}}\sigma_{\mathbf{b_{j,t,L}}}}, \tag{5}$$

where $\mathbf{b_{i,t,L}} = [b_{i,t}, b_{i,(t+1)}, ..., b_{i,(t+L)}]$ represents the average time series of BOLD activations of region i between time t and $t+L$, $cov()$ defines the covariance, and σ_s represents the standard deviation of time series s. This approach assumes that the pair of similar time series represent the same cognitive process measured by fMRI signals.

Closed Form Ridge Regression In order to generate brain networks with the method proposed in [18], we estimate the activation of a region from the activations of its neighboring regions in a time interval $[t, t+L]$. We minimize the loss function in Equation 2 using closed form solution for ridge regression. The loss function is minimized with respect to the edge weights outgoing from a vertex v_i, $\overline{a}_i = [a_{i,1}, a_{i,2}...a_{i,M}]$ and the following closed form solution of ridge regression is obtained:

$$\mathbf{\overline{a}_i} = (\mathbf{B}^T\mathbf{B} + \lambda\mathbf{I})^{-1}\mathbf{B}^T\mathbf{b_{i,t,L}}, \tag{6}$$

where $\mathbf{B}$ is an $L \times (M-1)$ matrix, whose columns consist of the average BOLD activations of anatomic regions except for the region r_i in the time interval $[t, t+L]$ such that column j of matrix $\mathbf{B}$ is $\mathbf{b_{j,t,L}}$. $\lambda \in R$ represents the regularization parameter.

3 Experiments & Results

In order to examine the representation power of the suggested Artificial Brain Networks, we compare them with the baseline methods, presented in the previous subsection, on two different fMRI dataset. The comparison is done by measuring the cognitive task classification performances of all the models.

3.1 Human Connectome Project (HCP) Experiment

In Human Connectome Project dataset, 808 subjects attended 7 sessions of fMRI scanning in each of which the subjects were required to complete a different cognitive task with various durations, namely, Emotion Processing, Gambling, Language, Motor, Relational Processing, Social Cognition, and Working Memory. We aim to discriminate these 7 tasks using the edge weights of the formed brain graphs.

In the experiments, the learning rate α was empirically chosen as $\alpha = 10^{-5}$ for both dABN and uABN and window size is chosen as $L = 40$. We tested the directed and undirected Artificial Brain Networks and Ridge Regression method using various λ values. Since computation of Pearson correlation does not require any hyper parameter estimation, a single result is obtained for the Pearson correlation method.

After estimating the Artificial Brain Networks and forming the feature vectors from edge weights of the brain networks, we performed within-subject and across-subject experiments using Support Vector Machines with linear kernel. During the within-subjects experiments, we performed 3-fold cross validation using only the samples of a single subject. Table 1 shows the average of within-subject experiment results over 807 subjects, when the classification is performed using a single subject brain network of 7 tasks. During the across-subject experiments, we performed 3-fold cross validation using the samples obtained from 807 subjects. For each fold we employed the samples from 538 subjects to train and 269 subject to test the classifier. Table 2 shows the across-subject experiment results.

Table 1. Within-Subject Performances of Brain Networks on HCP Dataset.

	Pearson Corr.		Ridge Reg.		dABN		uABN	
λ	Mean	Std	Mean	Std	Mean	Std	Mean	Std
0	0.7194	0.16	-	-	**0.7435**	0.13	0.5918	0.13
32	0.7194	0.16	0.7957	0.11	**0.9133**	0.08	**0.913**	0.08
64	0.7194	0.16	0.8304	0.11	**0.9406**	0.07	**0.9402**	0.07
128	0.7194	0.16	0.8377	0.11	**0.9463**	0.06	**0.9462**	0.07
256	0.7194	0.16	0.8119	0.12	**0.9313**	0.08	**0.9307**	0.08
512	0.7194	0.16	0.7462	0.13	**0.8852**	0.1	**0.8849**	0.1

Table 2. Across-Subject Performances of Brain Networks on HCP Dataset.

	Pearson Corr.		Ridge Reg.		dABN		uABN	
λ	Mean	Std	Mean	Std	Mean	Std	Mean	Std
0	**0.7524**	0.01	-	-	0.6654	0.01	0.5681	0.01
32	0.7524	0.01	0.8027	0.01	**0.8153**	0.00	0.8123	0.00
64	0.7524	0.01	0.8223	0.00	**0.8312**	0.01	0.8297	0.01
128	0.7524	0.01	0.8370	0.01	**0.8401**	0.01	0.8393	0.01
256	0.7524	0.01	**0.8461**	0.01	0.8410	0.01	0.8406	0.00
512	0.7524	0.01	**0.8466**	0.01	0.8357	0.01	0.8357	0.01

Table 1 shows that in within subject experiments our methods, dABN and uABN, have the best performances in classifying the cognitive task under different λ values, furthermore performances of directed networks are slightly better than undirected ones. It can be observed that as λ increases, generalization of our models also increase up to $\lambda = 128$.

Table 2 shows that our methods outperforms the others within a range of lambdas, $\lambda = \{32, 64, 128\}$. Pearson Correlation results in the best accuracy when no regularization is applied to Artificial Brain Networks. Closed Form Ridge Regression solution offers more discriminative power in higher λ values.

3.2 Tower of London(TOL) Experiment

We also test the validity of the suggested Artificial Brain Network on a relatively more difficult fMRI dataset, recorded when the subjects solve Tower of London (TOL) problem. TOL is a puzzle game which has been used to study complex problem solving tasks in human brain. TOL dataset used in our experiments contains fMRI measurements of 18 subjects attending 4 session of problem solving experiment. In the fMRI experiments, subjects were asked to solve 18 different puzzles on computerized version of TOL problem [16]. There are two labeled subtask of problem solving with varying time periods namely, planning and execution phases.

As the nature of the data is not compatible with a sliding window approach and the dimensionality is too high for a computational model, in the study of Alchihabi et al. [1], a series of preprocessing steps were suggested for the TOL dataset. In this study, we employ the first two steps of their pipeline. In the first step called *voxel selection and regrouping*, a feature selection method is applied on time series of voxels to select the "important" ones. Then, the activations of the selected voxels in the same region are averaged to obtain the activity of corresponding region. As a result, a more informative and lower dimensional representation is achieved. In the second step, bi-cubic spline interpolation is applied to every consecutive brain volumes and a number of new brain volumes are inserted between two brain volumes to increase temporal resolution. For the details of interpolation, refer to [1]. In this study, the optimal number of volumes

inserted between two consecutive brain volumes are found empirically and it is set to 4. Therefore, the time resolution of the data is increased four times.

We applied the above-mentioned preprocessing steps to all of the 72 sessions in the dataset. After the voxel selection phase, number of regions containing selected voxels is much less than 116 regions. Note that, we discard regions located in Cerebellum and Vermis. Window size for this dataset was set to $L = 5$, since there are at least 5 measurements for every sub-phase after the interpolation. The neural network parameters used in our experiments are $\alpha = 10^{-6}$ and $\#epochs = 10$. Table 3 shows the mean and standard deviation of classification accuracies obtained with our method and the base-line methods. Similar to HCP experiments, we slided non-overlapping windows on the measurements and we performed 3-fold cross validation during TOL experiments.

Table 3. Across-Subject Performances of Mesh Networks on TOL Dataset.

	Pearson Corr.		Ridge Reg.		dABN		uABN	
λ	Mean	Std	Mean	Std	Mean	Std	Mean	Std
0	0.6119	0.09	-	-	**0.8914**	0.11	0.8499	0.12
32	0.6119	0.09	0.6688	0.10	**0.8913**	0.11	0.8499	0.12
64	0.6119	0.09	0.6651	0.10	**0.8914**	0.11	0.8499	0.12
128	0.6119	0.09	0.6679	0.10	**0.8906**	0.11	0.8499	0.12
256	0.6119	0.09	0.6685	0.10	**0.8905**	0.11	0.8500	0.12
512	0.6119	0.09	0.6705	0.10	**0.8912**	0.11	0.8498	0.12

Table 3 shows that using Artificial Brain Networks gives better performances than using Pearson Correlation and Closed Form Ridge Regression methods in classifying sub-phases of complex problem solving under various regularization parameters. We observe that decoding performances of directed brain networks outperforms those of undirected brain networks.

4 Discussion and Future Work

In this study, we introduce a network representation of fMRI signals, recorded when the subjects perform a cognitive task. We show that the suggested Artificial Brain Network estimated from the average activations of anatomic regions using an artificial neural network leads to a powerful representation to discriminate cognitive processes. Compared to the brain networks obtained by ridge regression, the suggested Artificial Brain Network achieves more discriminative features. The success of the suggested brain network can be attributed to the iterative nature of the neural network algorithms to optimize the loss function, which avoids the singularity problems of Ridge Regression.

In most of the studies, it is customary to represent functional brain connectivities as an undirected graphs. However, in this study, we observe that the

directed network representations capture more discriminative features compared to the undirected ones in brain decoding problems.

In this study, we consider complete brain graphs where all regions are assumed to have connections to each other. A sparser brain representation can be computationally more efficient and neuro-scientifically more accurate. As a future work, we aim to estimate more efficient brain network representations by employing some sparsity parameters in the artificial neural networks.

It is well-known that brain processes the information in various frequency bands. [21] and [5] applied discrete wavelet transform before creating connectivity graphs. A similar approach can be taken for a more complete temporal information in brain decoding problems.

5 Acknowledgment

The work is supported by TUBITAK (Scientific and Technological Research Council of Turkey) under the grant No: 116E091. We also thank Sharlene Newman, from Indiana University, for providing us the TOL dataset.

References

1. Alchihabi, A., Kivilicim, B.B., Ekmekci, O., Newman, S.D., Vural, F.T.Y.: Decoding cognitive subtasks of complex problem solving using fmri signals. In: 2018 26th Signal Processing and Communications Applications Conference (SIU). IEEE (2018)
2. Alchihabi, A., Kivilicim, B.B., Newman, S.D., Vural, F.T.Y.: A dynamic network representation of fmri for modeling and analyzing the problem solving task. In: Biomedical Imaging (ISBI 2018), 2018 IEEE 15th International Symposium on. pp. 114–117. IEEE (2018)
3. Calhoun, V.D., Adali, T., Hansen, L.K., Larsen, J., Pekar, J.J.: Ica of functional mri data: an overview. In: in Proceedings of the International Workshop on Independent Component Analysis and Blind Signal Separation. Citeseer (2003)
4. Calhoun, V.D., Liu, J., Adalı, T.: A review of group ica for fmri data and ica for joint inference of imaging, genetic, and erp data. Neuroimage **45**(1), S163–S172 (2009)
5. Ertugrul, I.O., Ozay, M., Vural, F.T.Y.: Hierarchical multi-resolution mesh networks for brain decoding. Brain imaging and behavior pp. 1–17 (2016)
6. Fırat, O., Özay, M., Önal, I., Öztekiny, İ., Vural, F.T.Y.: Functional mesh learning for pattern analysis of cognitive processes. In: Cognitive Informatics & Cognitive Computing (ICCI* CC), 2013 12th IEEE International Conference on. pp. 161–167. IEEE (2013)
7. Firat, O., Oztekin, L., Vural, F.T.Y.: Deep learning for brain decoding. In: Image Processing (ICIP), 2014 IEEE International Conference on. pp. 2784–2788. IEEE (2014)
8. Kawahara, J., Brown, C.J., Miller, S.P., Booth, B.G., Chau, V., Grunau, R.E., Zwicker, J.G., Hamarneh, G.: Brainnetcnn: convolutional neural networks for brain networks; towards predicting neurodevelopment. NeuroImage **146**, 1038–1049 (2017)

9. Koyamada, S., Shikauchi, Y., Nakae, K., Koyama, M., Ishii, S.: Deep learning of fmri big data: a novel approach to subject-transfer decoding. arXiv preprint arXiv:1502.00093 (2015)
10. Kurmukov, A., Ananyeva, M., Dodonova, Y., Gutman, B., Faskowitz, J., Jahanshad, N., Thompson, P., Zhukov, L.: Classifying phenotypes based on the community structure of human brain networks. In: Graphs in Biomedical Image Analysis, Computational Anatomy and Imaging Genetics, pp. 3–11. Springer (2017)
11. Lynall, M.E., Bassett, D.S., Kerwin, R., McKenna, P.J., Kitzbichler, M., Muller, U., Bullmore, E.: Functional connectivity and brain networks in schizophrenia. Journal of Neuroscience **30**(28), 9477–9487 (2010)
12. McKeown, M.J., Sejnowski, T.J., et al.: Independent component analysis of fmri data: examining the assumptions. Human brain mapping **6**(5-6), 368–372 (1998)
13. Menon, V.: Large-scale brain networks and psychopathology: a unifying triple network model. Trends in cognitive sciences **15**(10), 483–506 (2011)
14. Michel, V., Gramfort, A., Varoquaux, G., Eger, E., Keribin, C., Thirion, B.: A supervised clustering approach for fmri-based inference of brain states. Pattern Recognition **45**(6), 2041–2049 (2012)
15. Mitchell, T.M., Hutchinson, R., Niculescu, R.S., Pereira, F., Wang, X., Just, M., Newman, S.: Learning to decode cognitive states from brain images. Machine learning **57**(1-2), 145–175 (2004)
16. Newman, S.D., Greco, J.A., Lee, D.: An fmri study of the tower of london: a look at problem structure differences. Brain research **1286**, 123–132 (2009)
17. Onal, I., Ozay, M., Mizrak, E., Oztekin, I., Vural, F.T.Y.: A new representation of fmri signal by a set of local meshes for brain decoding. IEEE Transactions on Signal and Information Processing over Networks **3**(4), 683–694 (2017)
18. Onal, I., Ozay, M., Vural, F.T.Y.: Modeling voxel connectivity for brain decoding. In: Pattern Recognition in NeuroImaging (PRNI), 2015 International Workshop on. pp. 5–8. IEEE (2015)
19. Ozay, M., Öztekin, I., Öztekin, U., Vural, F.T.Y.: Mesh learning for classifying cognitive processes. arXiv preprint arXiv:1205.2382 (2012)
20. Pereira, F., Mitchell, T., Botvinick, M.: Machine learning classifiers and fmri: a tutorial overview. Neuroimage **45**(1), S199–S209 (2009)
21. Richiardi, J., Eryilmaz, H., Schwartz, S., Vuilleumier, P., Van De Ville, D.: Decoding brain states from fmri connectivity graphs. Neuroimage **56**(2), 616–626 (2011)
22. Smith, S.M., Hyvärinen, A., Varoquaux, G., Miller, K.L., Beckmann, C.F.: Group-pca for very large fmri datasets. Neuroimage **101**, 738–749 (2014)
23. Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., Joliot, M.: Automated anatomical labeling of activations in spm using a macroscopic anatomical parcellation of the mni mri single-subject brain. Neuroimage **15**(1), 273–289 (2002)
24. Vaidyanathan, P.: The theory of linear prediction. Synthesis lectures on signal processing **2**(1), 1–184 (2007)
25. Wang, X., Hutchinson, R., Mitchell, T.M.: Training fmri classifiers to detect cognitive states across multiple human subjects. In: Advances in neural information processing systems. pp. 709–716 (2004)
26. Zhou, Z., Ding, M., Chen, Y., Wright, P., Lu, Z., Liu, Y.: Detecting directional influence in fmri connectivity analysis using pca based granger causality. Brain research **1289**, 22–29 (2009)

Index of Authors