2020

Novel Computational Approaches For Multidimensional Brain Image Analysis

Harish Raviprakash
*University of Central Florida*

Find similar works at: [https://stars.library.ucf.edu/etd2020](https://stars.library.ucf.edu/etd2020)

This Doctoral Dissertation (Open Access) is brought to you for free and open access by STARS. It has been accepted for inclusion in Electronic Theses and Dissertations, 2020- by an authorized administrator of STARS. For more information, please contact STARS@ucf.edu.

**STARS Citation**
[https://stars.library.ucf.edu/etd2020/618](https://stars.library.ucf.edu/etd2020/618)
NOVEL COMPUTATIONAL APPROACHES FOR MULTIDIMENSIONAL BRAIN IMAGE ANALYSIS

by

HARISH RAVIPRAKASH
B.E. Telecommunications Engineering, M.S. Ramaiah Institute of Technology, Bangalore, 2010

A dissertation submitted in partial fulfilment of the requirements
for the degree of Doctor of Philosophy
in the Department of Computer Science
in the College of Engineering and Computer Science
at the University of Central Florida
Orlando, Florida

Summer Term
2020

Major Professor: Ulas Bagci
The overall goal of this dissertation is focused on addressing challenging problems in 1D, 2D/3D and 4D neuroimaging by developing novel algorithms that combine signal processing and machine learning techniques. One of these challenging tasks is the accurate localization of the eloquent language cortex in brain resection pre-surgery patients. This is especially important since inaccurate localization can lead to diminished functionalities and thus, a poor quality of life for the patient.

The first part of this dissertation addresses this problem in the case of drug-resistant epileptic patients. We propose a novel machine learning based algorithm to establish an alternate electrical-stimulation-free approach, electro-corticography (ECoG) as a viable technique for localization of the eloquent language cortex. We process the 1D signals in frequency domain to train a classifier and identify language responsive electrodes from the surface of the brain. We then enhance the proposed approach by developing novel multi-modal deep learning algorithms. We test different aspects of the experimental paradigm and identify the best features and models for classification.

Another difficult neuroimaging task is that of identifying biomarkers of a disease. This is even more challenging considering that skill acquisition leads to neurological changes. We propose to help understand these changes in the brain of chess masters via a multi-modal approach that combines 3D and 4D imaging modalities in a novel way. The proposed approaches may help narrow the regions to be tested in pre-surgical localization tasks and in better surgery planning. The proposed work may also pave the way for a holistic view of the human brain by combining several modalities into one.

Finally, we deal with the problem of learning strong signal representations/features by proposing a novel capsule based variational autoencoder, B-Caps. The proposed B-Caps helps in learning a strong feature representation that can be used with multi-dimensional data.
Neuroimaging is an interdisciplinary field involving physicists and engineers designing the imaging systems, computer scientists enabling the processing of brain imaging and statisticians helping quantify any information gleaned from the captured images. Brain images are captured as 1D signals (via electro-encephalography (EEG)), 2D/3D images and 4D volumes (via computer tomography and magnetic resonance imaging). With the increasing availability and quality of imaging technologies, the effort into analyzing these multidimensional signals to understand the human brain has intensified. Towards this, several studies have been performed to capture and detect the brain function related to motor, language and cognition. This has enabled the neuroimaging community to better understand the effect of neurological disorders in the human brain. Neurological disorders is the second leading cause of deaths globally and there are still several gaps in understanding the nature and issues related to these disorders. Neuroscientists and computer scientists have combined their knowledge and expertise to make it easier to detect and diagnose several neurological disorders such as Alzheimer’s disease (AD), Parkinsons and so on. The approaches have used multi-modal images such as magnetic resonance images (MRI), functional MRI (fMRI), positron emission tomography (PET) and EEG in conjunction with machine learning and statistical approaches to identify patterns that distinguish the brain of a healthy subject from that of a subject with a specific neurological disorder.

Epilepsy is a neurological disorder which sometimes needs surgery to improve the patient’s quality of life. Epilepsy surgery is a curative option but brain regions associated with language and cognitive functions can be affected by surgery. Therefore, accurate localization of brain regions with language and cognitive functions should be carefully determined prior to surgery. Real Time Functional Mapping (RTFM) has been shown to be a safer alternative to electrical cortical stimulation mapping (ESM), which is the clinical/gold standard. Current methods for analyzing RTFM
signals are based on statistical comparison of signal power at certain frequency bands with limited response assessment accuracies. In the first part of this dissertation, we address the limitation of the current strategy by using statistical signal processing to extract the power spectral density, expanding the analysis of the signal into the full frequency spectrum and then applying machine learning algorithms, specifically random (decision) forest, to automatically classify RFTM signals into positive and negative response based on language comprehension task for language localization. We use power spectral density of a given time-series signal and train the random forest classifier based on the ground truth labels obtained from ESM. To the best of our knowledge, this is the first study (i) exploring the use of machine learning approaches to determine RTFM signal characteristics and auto-classification and (ii) using the whole-frequency band for region localization in RTFM. Promising results, obtained from RTFM and corresponding gold standard (ESM data) of six adult patients in a strictly controlled experimental setup, show that machine learning based auto-classification algorithms increases the strong potential of RTFM to be the clinical gold standard in the near future.

In the second part of the dissertation, we improve upon our results by introducing deep learning algorithms to the ECoG signals. After identifying the strength of frequency domain features in eloquent language cortex localization (the previous part), we incorporate time-domain features into the classification process. We explore several different signal processing approaches to extract different time-domain features and then combine the time-domain and frequency domain features, for the first time in the ECoG problem, to train several deep learning networks. We also explore different training strategies and network structures to thoroughly analyze the ECoG paradigm and achieve results which are the highest reported in recent times.

In the third part of the dissertation, we address challenges in 3D and 4D neuroimaging and explore the functional and anatomical differences between novice and professional chess players. Non-invasive neuroimaging has led to many discoveries about working of the human brain both in
healthy and disease conditions. A common task in brain image analysis includes diagnosis of a certain disorder wherein groups of healthy control subjects and disease subjects are analyzed and compared. However, for two groups of healthy subjects with different professional skills, the analysis of brain function remains largely unexplored. Towards this end, we process the resting state functional magnetic resonance image (MRI) signals to generate functional connectivity (FC) information. We also process T1 weighted MRI to estimate morphometric similarity network (MSN) connectivity information. We further combine the functional and anatomical features into a new connectivity matrix, the functional morphometric similarity connectome. To reduce the dimensionality of extracted information and perform successful classification, statistical features selection is employed and support vector machines are used for classification. Single modality and multi-modality classification is performed. The proposed approach identifies regions of the brain that are functionally and anatomically different between two groups of healthy subjects. These differences can be attributed to the additional development of the brain regions (owing to certain professional skills) which are related to the learning and memory tasks.

In medical imaging, domain knowledge is necessary to improve the signal representation before employing machine learning/deep learning algorithms. However, identifying the ideal signal representation can be a cumbersome process requiring testing several different representations before identifying the optimal one. There is thus a need to automatically learn strong signal representations with minimal supervision. Towards this, in the final part of the dissertation, we propose a novel capsule-based variational autoencoder algorithm that helps generate a better latent representation. The proof-of-concept results show quantifiable improvement in performance compared to a baseline variational autoencoder.
To my parents for their unconditional love and support.

To my wife for taking this journey with me.

To all who dare to pursue their dreams.
ACKNOWLEDGMENTS

I would like to acknowledge the contributions of all the people who helped make this work possible, and without whom this Ph.D. journey would have been incomplete. First and foremost, I wish to express my deepest gratitude to my advisor, Dr. Ulas Bagci, for his exceptional guidance throughout this process. His constant enthusiasm, support and patience helped me develop into a self-motivated researcher. His continuous motivation and guidance supported me through the difficult phases of this journey. I was able to develop several professional and interpersonal skills through various projects and collaborations, and for this I am truly grateful.

I would also like to thank Dr. Abhijit Mahalanobis, Dr. Gita Sukthankar, Dr. Marianna Pensky and Dr. Milena Korostenskaja for generously giving their time and constructive feedback that greatly helped improve this work. It has been a privilege to collaborate with Dr. Milena Korostenskaja, whom I have had the privilege to collaborate with

I would also like to acknowledge our collaborators at AdventHealth Orlando, namely, Dr. Milena Korostenskaja and Dr. Eduardo Castillo, for providing me with the opportunity to work on such an interesting project. I am extremely grateful to National Institutes of Health, namely, Dr. Nadia Biassou, for the opportunity to work on exciting projects and for advice and fruitful discussions.

I am also thankful to Dr. Syed Muhammad Anwar for his guidance and for always being available for discussions and analysis of our ideas. I also thank the members of our lab group, namely, Dr. Sarfaraz Hussein, Dr. Aliasghar Mortazi, Dr. Naji Khosravan and Rodney Lalonde for the constant discussion of scientific ideas. The friendship built over late night labwork and discussions over coffee breaks made this experience memorable.

My time as a graduate student would have been incomplete without my friends in Orlando. I
would like to thank Rakshath Kashyap, Phalgun Upadhyaya, Sahana Viswanathan, Raajsindhur Mathikere, Dr. Khurram Soomro, Shomaila Monis, Dr. Mona Mathew, Anukrit Saxena, Namita Varudkar and Nitish Gupta who helped make Orlando feel like home.

Finally and most importantly, I would like to express my gratitude to my parents for their unconditional love and for always supporting me in the pursuit of my dreams. My wife, Dr. Swarnim Ranjan has been a pillar of strength and a constant source of motivation throughout this journey. I thank my sister Divya Prashanth Rao for always being there for me.
# TABLE OF CONTENTS

**LIST OF FIGURES** ................................................................. xv

**LIST OF TABLES** .................................................................. xx

**CHAPTER 1: INTRODUCTION** .................................................. 1

1.1 Chapter 3 & 4: Eloquent Cortex Localization using Electrocorticography (ECoG) . 2

1.2 Chapter 5: Smart Thinkers .................................................. 5

1.3 Chapter 6: Improving Latent Representations via Variational Capsule Encoders . 6

**CHAPTER 2: BACKGROUND AND LITERATURE REVIEW** ............... 9

2.1 Eloquent Cortex Localization using Electrocorticography ............... 9

2.2 Smart Thinkers ........................................................................ 10

2.3 Improving Latent Representations via Variational Capsule Encoders .... 12

**CHAPTER 3: LANGUAGE CORTEX LOCALIZATION USING MACHINE LEARNING** 14

3.1 Overview .............................................................................. 14

3.2 Materials and Methods ......................................................... 15

3.2.1 Data Collection and Experimental Setup ............................... 15
3.2.2 Pre-processing ................................................. 17
3.2.3 Classification Model ....................................... 18

3.3 Experiments and Results ..................................... 21
3.4 Summary .......................................................... 25

CHAPTER 4: IMPROVING LANGUAGE CORTEX LOCALIZATION USING DEEP LEARNING APPROACHES .................. 26

4.1 Overview .......................................................... 26
4.2 Materials and Methods ....................................... 26
  4.2.1 Dataset ....................................................... 27
  4.2.2 Pre-processing ECoG Signal (Step 1) .................... 28
  4.2.3 time-domain RNN (Step 2) ................................. 29
  4.2.4 Frequency Domain Features (Step 3) .................... 32
  4.2.5 Fusion for Hybrid Domain (Step 4) ...................... 34
  4.2.6 Majority Voting (Step 5) .................................. 35
4.3 Experiments and Results ...................................... 36
  4.3.1 Task Paradigm .............................................. 36
  4.3.2 Training Paradigms ....................................... 37
4.3.3 Our Proposed Deep Network Architectures .............................. 37
4.3.4 Model Validation .......................................................... 39
4.3.5 Training on individual features on active task data (Training Paradigm-I) . 40
4.3.6 Training with multiple features on active task data – feature fusion (Training Paradigm-II) ............................................. 42
4.3.7 Training with individual features on individual stories/active task data (Training Paradigm-III) ............................................. 44
4.4 Summary ........................................................................... 45

CHAPTER 5: SMART THINKERS ............................................. 47

5.1 Overview ........................................................................... 47
5.2 Materials and Methods ...................................................... 48
5.2.1 Data ............................................................................. 48
5.2.2 Morphometric Similarity Network ..................................... 51
5.2.3 Functional Connectivity ................................................ 52
5.2.4 Functional Morphometric Similarity (FMS) Connectome .... 52
5.3 Experiments and Results ...................................................... 53
5.3.1 Dataset ......................................................................... 53
5.3.2 Statistical Significance of Morphometric Features ............. 55
LIST OF FIGURES

Figure 1.1: The scope of the dissertation is shown. The clinical motivation is to help understand the functional and anatomical differences in different groups and improve the surgical outcome via accurate presurgical eloquent cortex localization. Machine learning, deep learning and uncertainty driven algorithms drive the study. ................................. 2

Figure 1.2: Language specific areas in the brain ................................. 3

Figure 3.1: The language localization framework with RTFM approach include the following steps: ECoG signal recording, data transfer, storage, research and clinical paths, and tasks. Note that, RTFM signals are obtained from subdurally implanted grid electrodes. ................................. 15

Figure 3.2: Subdural grid localization and position of ECoG electrodes on the brain surface are illustrated (left). For a sample of 1 min duration, both control and active tasks are illustrated (right). ................................. 17

Figure 3.3: PRC vs NRC in different frequency bands. a,b show an example of the difference between PRC and NRC in high-γ band. c,d show the same samples in a higher frequency range. ................................. 19
Figure 3.4: Auto-classification workflow: First the signals are split into its contributing blocks. After, power spectral density (PSD) of the signal is estimated and the blocks are stacked from all channels. Finally, a random forest (RF) classifier is used for discriminating positive response channels (PRC) and negative response channels (NRC).

Figure 3.5: Classification scores on Language Comprehension Task. for E1, E2 & E3 which are the experiments on the full frequency spectrum, \( \mu, \beta, \text{high-}\gamma \) bands and high-\( \gamma \) band only respectively.

Figure 3.6: Comparison of proposed approach, Random Forest (RF) and current state-of-the-art, ECoG-EM on language comprehension task.

Figure 4.1: Overview of the proposed ECoG signal classification approach for each channel on the cortical surface in the subdural space. PRC - Positive Response Channel, NRC - Negative Response Channel. Each individual step is considered as a “module” in the overall system design.

Figure 4.2: Deep network structure for extraction of time-domain features from the input signal. Note that these features are combined with frequency domain features (Auto-Regressive) later for final prediction of the channels.

Figure 4.3: Deep network structure of the fusion module. AR (i.e., frequency) features (orange) and time-domain features (green) are concatenated and classified.
Figure 4.4: A overview of the steps involved in the proposed methodology showing ECoG data recording in response to story listening task, time and frequency domain feature extraction, different training paradigms and majority voting to classify positive and negative response channels. AT: Active Time, AR: Auto Regressive; CT: Control Task, EF: Early Fusion, LF: Late Fusion, PRC: Positive Response Channel, NRC: Negative Response Channel. .................................................. 38

Figure 4.5: Deep network structure of the various AT and AT-AR models. AT- Active Time, AR- Autoregressive. ................................................................. 41

Figure 5.1: Generating the Morphometric Similarity Network (MSN) from the anatomical images. Morphometric measures are extracted from the cortical surface of the T1 image and pearson’s correlation is used to build the MSN.  ........................................ 51

Figure 5.2: Generating the Functional Morphometric Similarity (FMS) network from the anatomical and function images. Left: Morphometric measures are extracted from the cortical surface of the T1 image. Right: Functional connectivity is generated from the surface of the fMRI and node degrees (ND) are extracted via sparsification (thresholding the edge strength). The morphometric and functional measures are combined to generate the FMS. NS - Node Strength, @ - thresholded at. ........................................ 54

Figure 5.3: Top 10 functional connections identified in classification of Chess Masters vs Novice Players. ................................................................. 58
Figure 5.4: Top 10 morphometric connections differentiating Chess Masters from Novice Players. Axial view of the anatomical connections with nodes representing one of the 17 networks in the Yeo atlas. 59

Figure 5.5: Common Functional and Morphometric connections differentiating Chess Masters from Novice Players. Connections are derived from common connections from the morphometric similarity networks and functional connectivity network. 61

Figure 5.6: Top 10 Functional Anatomical connections differentiating Chess Masters from Novice Players. 62

Figure 6.1: B-Caps architecture: Represented by three different parameters; the number of capsules (C), the size of description vector (D), and the latent dimension (L). 66

Figure 6.2: Reconstruction results with different training strategies. 68

Figure 6.3: The distribution of digits in the latent space for basic VAE and B-Caps (C=16, D=64) on MNIST dataset. Although none of these distributions are optimal, B-Caps latent space has visibly better allocations of certain classes. 71

Figure 6.4: Improvement in reconstruction of MNIST digits as a function of latent variable length. 74

Figure 6.5: The distribution of the classes in latent space for basic VAE and B-Caps (C=8, D=64) on Fashion-MNIST dataset. 76
Figure 6.6: A comparison between reconstruction of Fashion-MNIST images using basic VAE and B-Caps. Left to Right: Original image, basic VAE reconstruction and B-Caps reconstruction, $L = 10$  

Figure 6.7: Improvement of reconstruction as a function of latent variable length.  

Figure 6.8: Variation in image reconstruction quality measured using SSIM for different latent variable dimensions.  

Figure 6.9: Variation of classification performance measured using F1 score for different latent variable dimensions.  

Figure 6.10: Variation of classification performance with latent variable dimensions where basic VAE has intermediate layer of dimension 1024.  

Figure 6.11: Comparison of models with and without Batch Normalization.
Table 3.1: Patient Demographics, grid placement, and Epilepsy status are summarized. 
Left hemisphere is dominant for language tasks. ............................. 16

Table 4.1: Patient demographics, clinical information, grid placement, and information 
about the number of analysed channels (electrodes) are summarized. All 
study participants were left hemisphere language dominant for language. . . . . 28

Table 4.2: Channel classification accuracy for different network architectures. AT- ac-
tive time, AR – auto regression. .................................................. 42

Table 4.3: Comparing channel classification accuracy for active and control data. AT-
active time, AR – auto regression, CT- control time. .......................... 42

Table 4.4: Channel classification performance parameters (with mean and variance) for 
active task data with individual hand-crafted time-domain features using the 
AT-AR³ model (training paradigm-I). ........................................... 43

Table 4.5: Channel classification performance parameters for different approaches using 
feature fusion from time and frequency domains (training paradigm-II). AT-
Active time, AR- autoregressive, EF- early fusion, LF- late fusion. ........... 44

Table 4.6: Channel classification performance (mean and variance) for individual active 
task data, using each story independently (training paradigm-III). ........... 44

Table 5.1: Subject demographics. ..................................................... 55
Table 5.2: Anatomically significant functionally defined regions and the corresponding metric of difference. * - FWE corrected, L - Left, R - Right . . . . . . . . . . 55

Table 5.3: 10-fold cross-validation accuracy using functional connectivity features. w/o - without, w - with . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 57

Table 5.4: Classification results of morphometric similarity networks (MSN), functional connectivity networks (FC) and functional morphometric networks (FMS) on 10-fold cross-validation. SA - Surface area, CT_avg - Average cortical thickness, CT_sd - Cortical thickness standard error, FI - Folding index, CI - Curvature Index, GC - Gaussian curvature, ND@ - Node degree at threshold level, . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . 59

Table 6.1: Comparison of reconstruction quality on MNIST while varying the capsule types (C) and description length (D). std - standard deviation. . . . . . . . . . . 73

Table 6.2: Comparison of reconstruction quality on FASHION- MNIST while varying the capsule types (C) and description length (D). . . . . . . . . . . . . . . . . . 75

Table 6.3: Trainable encoder parameters in the basic VAE with intermediate layers of 512 and 1024 and B-Caps. FC- fully connected layer. . . . . . . . . . . . . . . . . . 80
CHAPTER 1: INTRODUCTION

Traditionally, in healthcare, radiologists manually observe and analyze images to detect and diagnose tumors/diseases. Planning a therapy or surgery is based upon the observation of the radiologist. This manual process can be time consuming and prone to errors in certain conditions. With increasing population and the availability of imaging devices, the number of images to be analyzed by a radiologist is increasing exponentially. Computer Aided Diagnosis (CAD) can help ease the burden if sufficiently accurate. Improving medical technology and storage options has increased the availability of data and the need for artificial intelligence (AI) in healthcare. Disease variations and complex diagnoses make the application of AI in healthcare more challenging. This thesis focuses on developing novel computational algorithms to solve challenging neuroimaging problems in different dimensions - ECoG (1D), MRI (3D) and fMRI (4D). The overall organization of the dissertation is shown in Figure 1.1. The chapter organization is as follows:

Chapter 3: ML in ECoG - address the problem of high-dimensional data in datasets of limited size.

Chapter 4: DL in ECoG - address the problem of high-dimensional data in datasets and learning to combine multimodal data.

Chapter 5: Smart Thinkers - address the challenge of combining multimodal data in difficult classification problems

Chapter 6: B-Caps - to improve the latent space representation for high-dimensional data
1.1 Chapter 3 & 4: Eloquent Cortex Localization using Electro corticalography (ECoG)

**Clinical Motivation:** Epilepsy is a chronic neurological disorder characterized by recurrent, unpredictable seizures with over 65 million reported cases around the world [1]. Approximately 20% of these patients are diagnosed with drug-resistant epilepsy (DRE). The only possible treatment in a majority of these cases is surgical intervention. During epilepsy surgery the pathological brain tissue, which is associated with seizures, might be surgically removed. While epilepsy surgery is a curative option for drug-resistant epilepsy, neurosurgeons need to avoid removing tissues associated with language, sensory, and motor functions. This calls for an accurate identification and localization of these functionally significant brain regions. The surgical procedure can be performed more accurately with a precise localization for preventing any corresponding post-surgical neurological/functional deficits. This would ensure an improved and sustainable post-surgical quality of life for patients presented for brain surgery. The language function in the brain is processed in
several regions primarily, the Wernicke’s area and Broca’s area as demonstrated in Figure 1.2. The Wernicke’s area is located in the posterior section of the superior temporal gyrus and is responsible for the receptive language task i.e., language comprehension. The Broca’s area, on the other hand, is more involved in speech production. There exists an anatomical connection between these two regions, named the arcuate fasciculus, which could induce a response in one region owing to the other’s activation.

Electro-cortical stimulation mapping (ESM) has been considered as the gold standard for functional cortex localization in epilepsy surgery. ESM is an invasive procedure that uses electrodes placed on the surface of the brain (grid electrodes) or within the brain (depth electrodes). It is considered vital for reducing the risk of language deficits post-surgery and therefore, expanding surgical options. ESM has a long history of serving as the main modality for pre-surgical functional mapping of epilepsy patients. Acute electrical cortical stimulation was successfully performed in 1950 during epilepsy surgery by Penfield and colleagues [2, 3]. During ESM, pairs of electrodes
covering the region of interest (in our case - eloquent cortex) are stimulated by delivering a brief electric pulse. The stimulation temporarily disables/inhibits the cortical area of interest (creates a temporary functional lesion). Behavioral changes such as unusual sensation, involuntary movements, or language impairments (i.e., speech paucity), observed during stimulation indicates that the tested area is essential to that task performance, and its resection might lead to functional deficits. Ojemann et al. [4] studied language localization using ESM, on a large dataset of 117 patients. The study found that there was sufficiently large individual variability in the exact location of language function and concluded that there was a need for an improved language localization model. Much later, more standard and effective tasks for expressive language localization, such as verb generation [5] and picture naming [6], were tested with the increased use of ESM.

However, one major drawback of ESM is its potential to induce after-discharges [7], which could result in seizures. Since stimulation provoked seizures can occur rather frequently during ESM procedures [8], ESM tests often need to be repeated, leading to extended time and effort from medical professionals (neuropsychologists and/or neurologists). In some cases, the ESM procedure cannot be completed due to repeated seizure activity and/or its consequences. The current limitations of the ESM have created a strong need for establishing other independent functional mapping modalities to identify eloquent cortex. Unfortunately, as of now, none of the existing neuro-imaging modalities are flexible enough to provide functional mapping results in real time in the operating room. Therefore, the search for a stand-alone methodology for functional eloquent cortex localization has been continuing and resulted in attempts to use electrocorticography (ECoG) as a viable alternative. ECoG is the invasive version of electroencephalography (EEG) and sometimes also referred to as intracranial encephalography, demonstrating excellent temporal resolution like EEG. Importantly, ECoG equipment is portable and can be utilized both at the patients’ bed-side and intra-operatively. Unlike EEG, it overcomes the problem of poor spatial resolution, since the activity of interest is recorded directly from the cortical brain surface. It also avoids the
problem of electrical signal attenuation in EEG caused by the signal propagation through tissues surrounding the brain. There is thus, a strong need to establish ECoG as a stand-alone modality for eloquent language cortex localization.

1.2 Chapter 5: Smart Thinkers

Functional connectivity networks of brain have gained increasing popularity as a technique for diagnosis of diseases due to its high test-retest reliability and reproducibility [9]. Many computational methods were proposed to facilitate individual and group-wise differences and similarities [10][11][12]. The methods that use machine learning approaches for analyzing FC networks have been shown to be promising [13, 14, 15]. However, functional connectivity network based classification suffers from the high dimensionality issues, where there are more features than the available data. Feature selection algorithms with sparse learning is a common approach used recently to handle the dimensionality problem. For instance, sparse learning techniques have been successfully applied to the diagnosis of Alzheimer’s disease (AD) [16], attention deficit hyperactivity disorder (ADHD) [17], and epilepsy [18]. Whereas, there is evidence of developing such systems for studying various clinical conditions, FC networks have not been widely adopted to analyze two groups of healthy subjects.

Clinical Motivation: Structural and functional brain studies have shown that there are structural correlates of intelligence and for intelligent people the brain regions are known to be more connected [19]. Skill acquisition or long term practicing of a particular set of actions can lead to changes in the brain structure as is evident from neuro-science [20, 21, 22, 23]. There are evidences of structural and functional differences between the brain of two healthy groups of subjects for a particular skill set [21]. Although resting state magnetic resonance imaging (rs-fMRI) is ideally suited for studying such differences by helping in understanding functionally linked brain
regions when the brain is at rest, exploring the global and local *structural* changes observed using T1-weighted magnetic resonance imaging (MRI) could also benefit in understanding difference that exist between such groups of healthy subjects.

We propose to extend the applicability of sparse learning methods to a new application: brain analysis of different groups of healthy subjects where one group consists of professional chess players (with many years of experience), and a second group of novice subjects. Since chess is a very demanding board game, *we hypothesize that there are significant differences, both structurally and in localized brain connections, that could be modelled by modern statistical machine learning methods*. Towards this aim, we use imaging data curated from grandmaster-level chess-players and normal controls (amateur chess players) [24].

1.3 Chapter 6: Improving Latent Representations via Variational Capsule Encoders

**Clinical Motivation:** Neuroimaging data tends to be very high-dimensional and noisy. The 1D signals are recorded at high sampling rates, for several minutes duration leading to hundreds of thousands of datapoints and from several channels. They contain both, a common noise across all channels and a channel-specific noise. This generally needs an expert to identify artifactual channels and separate them. Following this, signal processing techniques need to be employed to extract useful signal representations prior to the application of machine learning algorithms. Similarly, N-D images contain millions of voxels that may be affected by different types of noise and provide redundant information for task. This again necessitates signal processing to learn a more usable signal representation. Domain knowledge may help identify the useful representations but often requires different features to be tested. There is thus a need to learn automatically, a better signal representation, with minimal supervision.
Autoencoders (AEs) have been around since the 1980s [25] and are used for encoding the input into a latent space that optimally represents the high dimensional data with lower dimensions by introducing a bottleneck layer in the encoding-decoding process. AEs can be used to extract features for various classification/detection tasks [26] [27]. In a typical AE, the input data is passed through a few or several neural network layers to obtain a reduced and more compact dimension. This manageable delineation, the encoding vector, encodes the different learned attributes. Thus, the learned latent space encodes descriptive attributes of the data and can be used for several purposes including classification, and reconstruction. To understand and model the variations associated with these attributes, a probabilistic distribution estimate of the latent variables can be used. Variational Autoencoders (VAE) were proposed to do just that [28]. VAEs generate a continuous latent space as opposed to a conventional AE. Despite having all the advantages, both AE and VAE are not viewpoint invariant; hence, they either require a large amount of data for precise modeling or certain bounds on the learning process for a fair representation [29].

**Capsule Networks**

The capsule networks, by replacing scalar neurons with vectors, assist in learning a relationship between objects and its parts [30]. Capsules make the underlying assumption of objects or entities being composed of parts, and ideally learning the part-whole relationship for these entities benefits the learning process by making it invariant to transformations and novel viewpoints. Although convolutional neural networks (CNNs) have been successful in a wide spectrum of classification and detection applications, their performance could suffer when data representations have varying or novel viewpoints. These variations in the appearance manifold could likely be learned by carefully designed data augmentation methods, albeit adding to the computational cost. Traditional VAE would also fail to model these relationships in the latent space and thereby reduces the invariance of such models under various image transformations. Learning models that are transformation
invariant to such manifolds has been a challenging task. In our proposed study based on capsule networks, we develop a new variational encoder algorithm to address these challenges. This will generate stronger latent space representation that can be used to improve channel localization and classification tasks. Additionally, the benefit with variational approaches is that it will enable us to generate a confidence value for every prediction which will benefit in decision making.

Our major contributions are:

- We compare the positive response and negative response channels in regards to language response instead of to a baseline recording. To the best of our knowledge, this is the first work to do so.

- We introduce machine learning and deep learning approaches to the ECoG language response channel classification problem. To the best of our knowledge, this is also a first.

- Another first, we analyze the complete ECoG signal frequency spectrum to identify signal characteristics for mapping language cortex.

- We enable new understanding of the brain changes by classifying two groups of healthy controls (based on their skill specialization).

- We propose a novel method to combine the anatomical and functional measures in MRI and fMRI to identify functional changes influenced by structural changes.

- We propose a novel VAE architecture, called Bayesian Capsules (or B-Caps), which combines VAE and capsule networks by utilizing the variational Bayes approach for learning a better signal representation.
CHAPTER 2: BACKGROUND AND LITERATURE REVIEW

2.1 Eloquent Cortex Localization using Electrocorticography

ECoG-based approaches have been used successfully for motor cortex localization [31, 32, 33, 34, 35]. In comparison, the localization of functional language cortex appears far more complex and challenging [36]. Current localization approaches are based on detecting positive response channels (called active channels or active electrodes) among the set of all channels. A baseline recording of each channel at resting-state is used to determine signal characteristics at specific frequency ranges. Most often, power of the ECoG signal lies within the alpha (α), beta (β), and (primarily) high gamma (high-γ) (70Hz-170Hz) frequency bands [37, 38]. These values are compared with the signal power measured during the execution of language task. The results of this approach for language mapping have not achieved desirable accuracy. For example, Arya et al. [39] studied high-γ response from ECoG recording of 7 patients during spontaneous conversation. The results showed low specificity and accuracy. In a follow up study, Arya et al. [40] demonstrated high-gamma modulation for the story-listening task and achieved high specificity, but sensitivity remained low. Korostenskaja et al. [41] showed that similar to the results for motor cortex, ECoG-FM can be used for eloquent language cortex localization as a complimentary technique with ESM, but not as a stand-alone modality. It has also been demonstrated that ECoG-FM can be used as a guiding tool for ESM, thereby reducing the time of ESM procedure and decreasing the risk of provoked seizures [42, 43]. Despite their potential, the current ECoG-FM approaches are not found capable enough to be used as a stand-alone methodology for accurate language mapping. To address these challenges and provide ECoG-FM more independence in eloquent language cortex localization, we fill in the following currently existing methodological gaps:
Available approaches compare a channel’s signal with its resting-state (baseline) recording and do not compare the channels’ characteristics to other recorded channels.

The signal characteristics in the frequency range beyond high-\(\gamma\) band have not been explored yet to the best of our knowledge.

More recently, ECoG-based deep learning applications have been gaining attention owing to the new BCI challenge dataset [44] inspired by the idea to interpret two-dimension finger movement from ECoG data in humans [45]. Though the challenge saw the top performers using machine learning approaches, the top performing team used a larger frequency spectrum beyond the high-\(\gamma\) band. Du et al. [46] proposed an LSTM based deep learning model to predict the finger flexion with \(\approx 83\%\) accuracy. Similarly, in [47], LSTM was again used to classify motor imagery movements. [48] proposed using deep learning to detect and decode responses during auditory stimulus in an animal model using micro-ECoG.

However, there has been limited work on validation of ECoG-FM for language cortex localization; hence, there are more evidences needed for utilization of ECoG-FM in the clinics.

2.2 Smart Thinkers

Skill acquisition or long term practicing of a particular set of actions can lead to changes in the brain structure and/or function. Rioult-Pedotti et al. trained rats in a reaching task with a single forepaw and found an increase in the strength of horizontal connections in the motor cortex [20]. The effect of long-term skill acquisition on human subjects was studied by evaluating the change in the gray matter volume using voxel based morphometry (VBM), on a dataset of expert musicians and amateur musicians [22]. and found volumetric changes in the auditory, motor and visual-spatial brain regions. In another study, VBM was used to study the morphological changes in the...
brain associated with the complexity of navigation induced learning in taxi and bus drivers and found increased posterior gray matter volume in the hippocampus of taxi drivers[49]. Di et. al. studied the alterations in functional connectivity in a group of professional badminton players and found altered functional connectivity between the left superior parietal and frontal regions [21] . Driemeyer et. al hypothesized that even short-term skill acquisition such as learning to juggle for three months can lead to detectable changes in the brain [23]. The authors found increased gray matter in the occipito-temporal cortex which comprises the motion sensitive area. Hence there are evidences of structural and functional differences between the brain of two healthy groups of subjects for a particular skill set. In a study by Wan et. al, the precuneus was found to be activated during the perception of chess board patterns [50]. To compare anatomical regions in the brains of chess masters and novice players, Duan et. al performed voxel by voxel volumetric comparison using two-sample t-tests [51]. Decreased gray matter volumes in the left and right caudate regions were found for chess masters compared to the matched novice controls. Additionally, a resting state analysis with respect to connections from the caudate also found increased correlations to posterior cingulate cortex (PCC) and bilateral angular gyrus (AG), important parts of the default mode network. In another study on a different chess player dataset, Hanggi et. al found that chess experts demonstrated specific anatomical features in the caudate nucleus, occipito-temporal junction and the precuneus that played important roles in the neural network activation during chess playing [52]. A graph theory analysis on the rs-fMRI chess data was performed in [53] where they thresholded the functional connectivity matrices at different edge strengths to achieve desired network sparsity and found enhanced small-worldness in chess experts compared to the novice players. From these few studies it can be inferred that there may exist differences between chess masters and novice players in brain topologies and functional organization. However, the influence of the global topological changes on functional connectivity or vice-versa is not know.

In a step away from the conventional anatomical analysis, Sabuncu et. al. proposed to use brain
morphology to explain phenotypic variations as a method of identifying a global statistical association between brain morphology and observable traits [54]. More recently, a morphometric similarity network (MSN) was proposed to map the network architecture of anatomically connected regions in the brain, where a correlation between morphometric measures between regions of the brain was computed [55]. It was found that MSN modules reiterated known cortical cytoarchitectonic divisions establishing MSN as a valid measure. Structural connectivity can be considered as the basis of functional connectivity. Several works have studied the relationship between these two in mice [56], in humans using simulations [57] and for real data [58], and for specific tasks such as cognition [59]. However, there has been limited work on analyzing the combined effect of anatomical and functional differences. In most cases, anatomical differences are identified and used to localize the search for functional differences.

### 2.3 Improving Latent Representations via Variational Capsule Encoders

Since its introduction in 2013, several variations to VAEs have been proposed to cater to different tasks and domains. Among many, some of the representative works are summarized as follows. A deep CNN (encoder) and deep generative deconvolutional network (decoder) was proposed for modeling images and their labels/captions [60]. The learned model was able to run in a semi-supervised setting in test cases where the labels were not available. In natural language processing, Kusner et. al. proposed a grammar VAE to incorporate knowledge about the structure of data and applied this model to parse trees [61]. In [62], a variational lossy AE was proposed to learn more global representations while dropping local ones. The authors combined the VAE with recurrent neural networks to achieve this goal. Along similar lines, Habibie et. al. proposed to learn the manifold of human motion from motion capture dataset using a recurrent VAE [63]. It was observed for deep stochastic models, that starting with the reconstruction loss before introducing the KL loss
was important for convergence [64]. It was also noted that batch normalization played an important role in these networks. A shape VAE was proposed, which modeled the distribution of object parts, locations of surface points, and the normal associated with these points [65]. The modeling of the distribution of object parts and locations, in a way attempted to model the part-part relationships. On the contrary, capsule networks took this a level up by modeling an object-part relationship and transforming the AE into a classification network [30]. Following this, the capsule networks were applied to a range of applications from text classification [66] and action detection [67], to brain tumor classification [68] and explainable medical diagnoses [69]. In certain domains, where the availability of labeled data is scarce, VAEs worked well to support semi-supervised learning [70]: the proposed architecture combined the latent space and reinforcement learning to enable learning for data with limited labels. For an effective inference from the latent variables in generative modeling, a Bayesian approach towards learning the latent representation could play a significant role. In a recent study, a routing algorithm was proposed for capsules inspired by the variational encoder architecture [71]. However, there is still little evidence of work which transforms a VAE such that the latent space representation makes use of the part-whole relationship probabilities. We argue that such an interpretable representation could be inferred by learning a latent embedding using capsules during the encoding process.
CHAPTER 3: LANGUAGE CORTEX LOCALIZATION USING MACHINE LEARNING

Related Publication:


3.1 Overview

We address the accuracy limitation of the current Real time functional mapping (RTFM) signal estimation methods by analyzing the full frequency spectrum of the signal in frequency domain instead of the restricted high-γ band analysis and by replacing signal power estimation methods with machine learning algorithms, specifically random forest (RF), as a proof of concept. We train RF with power spectral density of the time-series RTFM signal in supervised learning framework where ground truth labels are obtained from the gold-standard Electrocortical stimulation mapping (ESM). The details of the proposed method are presented in the following sections.
Figure 3.1: The language localization framework with RTFM approach include the following steps: ECoG signal recording, data transfer, storage, research and clinical paths, and tasks. Note that, RTFM signals are obtained from subdurally implanted grid electrodes.

3.2 Materials and Methods

3.2.1 Data Collection and Experimental Setup

RTFM is inter-cranial electroencephalogram (EEG) that tests the electrical activity in the brain. The basic setup is shown in Figure 3.1. To record ECoG signals, a craniotomy (removal of the skull section: bone flap) is performed and the dura is opened to access the brain tissue. The arrays of grid of electrodes (Figure 3.1, left) are then placed on the exposed cerebral cortex. The ECoG signals from the implanted subdural grids are split into two streams: one for continuous
clinical seizure monitoring and the other for ECoG-based functional mapping (ECoG-FM). The tool used to record the incoming ECoG signal is BCI2000 [72]. A baseline recording of the cortical activity is first acquired to capture the "resting-state" neuronal activity of the regions. The motor localization task using RTFM has seen considerable work with good accuracy [73][74], while the language localization task has proved to be more challenging [39]. The language function in the brain is processed in several regions primarily, the Wernicke’s area and Broca’s area as illustrated in Figure 1.2. The Wernicke’s area is located in the posterior section of the superior temporal gyrus and is responsible for the receptive language task i.e. language comprehension. The Broca’s area on the other hand is more involved in speech production. There exists an anatomical connection between these two regions, the arcuate fasciculus, which could induce a response in one region owing to the other’s activation. Following the baseline recording step, paradigms similar to those employed in ESM or functional Magnetic Resonance Imaging (fMRI) are employed to record the ECoG signal during cognitive tasks. One such paradigm is illustrated in Figure 3.2 which shows the experimental setup for the language comprehension task. Alternate 30 second blocks of control and task are recorded continuously at a fixed sampling rate of 1200Hz.

Table 3.1: Patient Demographics, grid placement, and Epilepsy status are summarized. Left hemisphere is dominant for language tasks.

<table>
<thead>
<tr>
<th>Subject #</th>
<th>Age (yrs)</th>
<th>Sex</th>
<th>Epilepsy Focus</th>
<th>Grid Placement</th>
<th>Epilepsy Onset (yrs)</th>
<th>Channels Tested/PRC / NRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>19</td>
<td>M</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>16</td>
<td>54 / 22 / 32</td>
</tr>
<tr>
<td>2</td>
<td>33</td>
<td>F</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>10</td>
<td>32 / 5 / 27</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>M</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>6</td>
<td>127 / 16 / 111</td>
</tr>
<tr>
<td>4</td>
<td>22</td>
<td>F</td>
<td>Parietal</td>
<td>Lateral</td>
<td>20</td>
<td>30 / 19 / 11</td>
</tr>
<tr>
<td>5</td>
<td>32</td>
<td>F</td>
<td>Temporal</td>
<td>Bilateral</td>
<td>26</td>
<td>48 / 10 / 38</td>
</tr>
<tr>
<td>6</td>
<td>52</td>
<td>M</td>
<td>Temporal</td>
<td>Lateral</td>
<td>30</td>
<td>48 / 5 / 43</td>
</tr>
</tbody>
</table>

For the language comprehension task, the active task is listening to a story while the control task is listening to broadband noise. Another associated paradigm is the reading comprehension task
wherein the subject reads sentences and replies with a "True" or "False" response. The system records information from all 128 channels.

3.2.2 Pre-processing

As a first step to preparing the data, non-task/control time points in the signal are eliminated. These correspond to the spontaneous activity recording before the 0-min in Figure 3.2 and any trailing signals at the end of the experiment. The use of the power of the signal is proposed in [38] as a discriminating feature between the baseline signal and paradigm experiment signal. Similar to that work, we propose to estimate the power spectral density (PSD) of the signal using the covariance method. This is an autoregressive (AR) based approach. The AR approach is a parametric approach and reduces the number of parameters in the model. It assumes that the $n^{th}$ sample is correlated

Figure 3.2: Subdural grid localization and position of ECoG electrodes on the brain surface are illustrated (left). For a sample of 1 min duration, both control and active tasks are illustrated (right).
with the $p$ previous samples. The problem thus translates into predicting the reduced number of parameters, the AR parameters. Herein, the AR parameters, $\tilde{a}[n]$, are estimated by forward linear prediction coefficients and then, the spectral estimate is calculated as

$$\hat{P}(f) = \frac{T \tilde{\rho}}{1 + \sum_{n=1}^{p} \tilde{a}[n] e^{-i 2\pi f n T}}$$

(3.1)

where $T$ is the inverse of the sampling rate ($f_s$), $\tilde{\rho}$ is the estimated noise variance and $p$ is the order of the AR process.

This approach gives us $\frac{f_s}{2} + 1$ frequency components. The PSD estimates are computed for each block (task/control) of each channel.

### 3.2.3 Classification Model

To differentiate positive response channel (PRC) from negative response channels (NRC), we focus on identifying structured signal patterns in signal blocks, which are not visible to the human eye. We hypothesize that the features of the active and control tasks within a PRC and NRC are similar between PRC and NRC but include substantial differences. This hypothesis can be visually tested and partially confirmed in Figure 3.2 where the PSD of the active and control blocks of PRC are visibly larger than those of NRC.

In order to test our hypothesis and provide scientific evidences of separability of active and control regions we propose to use RF Classifiers [75] that is capable of extracting structured local signal patterns as representative features. RF is an ensemble classifier technique wherein multiple decision trees are trained and the output is the mode of all the predicted classes. It has been shown in various different areas that RF is an efficient classifier [76][77][78]. In RF, briefly, each new
Figure 3.3: PRC vs NRC in different frequency bands. a,b show an example of the difference between PRC and NRC in high-γ band. c,d show the same samples in a higher frequency range.

tree is created and grown by first randomly sub-sampling the data with replacement. An ensemble of algorithms are used so that the sub-trees are learned differently from each other. For a feature vector \( \mathbf{v} = (v_1, v_2, \ldots, v_d) \in \mathbb{R}^d \), where \( d \) represents feature dimension, RF trains multiple decision trees and the output is determined based on combined predictions. In each node of decision trees, there is a weak learner (or split function) with binary output: \( h(\mathbf{v}, \theta) : \mathbb{R}^d \times \mathcal{T} \rightarrow \{0, 1\} \), where \( \mathcal{T} \) represents the space of all split parameters. Note that each node is assigned a different split function. RF includes hierarchically organized decision trees, in which data arriving at node \( j \) is divided into two parameters.

Overall, RF treats finding split parameters \( \theta_j \) as an optimization problem \( \theta_j = \theta \in \mathcal{T} I(\mathbf{v}, \theta) \), where
Figure 3.4: Auto-classification workflow: First the signals are split into its contributing blocks. After, power spectral density (PSD) of the signal is estimated and the blocks are stacked from all channels. Finally, a random forest (RF) classifier is used for discriminating positive response channels (PRC) and negative response channels (NRC).

$I$ is the objective function (i.e., split function) and $v$ represents the PSD coefficients in this particular application. As the tree is grown (Figure 3.4), an information criterion is used to determine the quality of a split. Commonly used metrics are Gini impurity and Entropy for information gain. To overcome potential over-fittings, a random sample of features is input to the trees so that the resulting predictions have minimal correlation with each other (i.e., minimum redundancy is
achieved). In our experiments, we have used linear data separation model of the RF.

In our experiments, we use full spectrum of RTFM signal (0-600 Hz) in frequency domain instead of restricted $\gamma$-band. Moreover, we stack the signal to enhance the frequency specific features rather than concatenating them. Each channel has 10 blocks (Figure 3.2) and the final channel classification is based on a majority voting (Figure 3.4) on the classified sub-blocks. For the tested data point (feature) $v$, the output is computed as a conditional distribution $p(c|v)$ where $c$ represents the categorical labels (positive vs. negative response). Final decision (classification) is made after using majority voting over $K$ leafs: $p(c|v) = \frac{1}{K} \sum_{k=1}^{K} p_t(c|v)$.

Model parameters

Number of trees, number of features, and data size fed to each tree with or without resampling and the information metric for data splitting are some of the RF parameters that need to be optimized. To achieve this, the model was repeatedly tested under different combinations of the above parameters. For the total number of trees, an incremental update approach was used where we increased the total number of trees till the increase in performance was negligible. Similarly, the number of features was set as the square root of the number of input variables. For the choice of splitting function, Gini impurity was used as for a binary classification problem, both measures yield similar results [79].

3.3 Experiments and Results

With IRB approval, ECoG data were recorded from six adult patients with intractable epilepsy. Table 3.1 summarizes the patient demographics and also highlights the number of channels that were tested per patient. The number tested varies based on grid placement, the epilepsy focus and
Subjects 1, 2, 3, 5 & 6 were tested with the language comprehension paradigm as shown in Figure 3.2. Subject 4 on the other hand underwent the reading comprehension test which involved listening to sentences and responding with True or False to the questions asked. Since this test also incorporates speech which would incite a response from the Broca’s area, channels corresponding to this area were not included in our experiments. The ground truth for all tested channels is from ESM (the gold standard). There are 77 PRCs and 262 NRCs in total. Since the aim is auto-classification of channels, the problem is a binary classification problem and hence, each block of data in a channel is assigned the same label. With the recording being 5 minutes long, we have 5 blocks of control and active tasks each per channel and hence, 3390 data samples in total. Owing to the large imbalance in data, 77 NRCs are randomly chosen from the 262. In total we have 1540 blocks of data. To learn a classifier for this data, we use 10-fold cross-validation and the mean over a 100 iterations is reported.

First, we tested whether the raw time signal data has sufficiently discriminating features i.e. in the time-domain. For this experiment, a RF model with 100 trees was used. The resulting classification accuracy was 61.79% with sensitivity and specificity around 60%. While this is marginally better than the simple flip of a coin scenario, it is insufficient to encourage the use of RTFM over ESM.

The raw data was then transformed to a different space, the frequency domain. To do this, the time signal data of each block was converted using the preprocessing step in Section 3.2. The order of the AR process is set to $\frac{\text{SamplingRate}}{10} = 120$. The PSD estimate is of length $f_s/2 + 1, 601$. Additionally, the PSD is log normalized and these log normalized features are used to train a RF classifier. An ensemble of 200 bagged classification trees is trained on 9 folds of the data and tested on the last fold.

In order to validate the use of classification of control & active task blocks for channel classifica-
Figure 3.5: Classification scores on Language Comprehension Task. for E1, E2 & E3 which are the experiments on the full frequency spectrum, $\mu,\beta,\gamma$ bands and high-$\gamma$ band only respectively.

Three different tests were performed to understand the contribution of the different frequency bands to the channel classification problem:

E1. Classification using full signal spectrum

E2. Classification using $\mu,\beta,\gamma$ sub-bands

E3. Classification using only the High-Gamma sub-band

In these experiments, the blocks are classified and a majority voting is applied to classify a channel
Figure 3.6: Comparison of proposed approach, Random Forest (RF) and current state-of-the-art, ECoG-EM on language comprehension task.

as PRC/NRC. Figure 3.5 summarizes the results of the above experiments on the language comprehension task. In concordance to what is observed in the ECoG-EM approaches [38][74], we find that the lower frequency bands, $\mu$ and $\beta$, do not contribute largely towards classification and the high-$\gamma$ band achieves good classification accuracy. It can also be seen that the full signal spectrum based classification has higher classification accuracy, sensitivity and specificity than the sub-band approaches indicating that the full spectrum has more information to offer.

Additionally, we tested the use of smaller blocks of data by further dividing each control/active task block into 10 sub-blocks. Each sub-block of data is the power spectrum representation of 3 seconds of recording. The classification was done based on a majority voting of the classified sub-blocks within a channel. The resulting classification accuracy of 78% is higher than the block-based classification indicating that there is more local information to be extracted from the signal.
The state-of-the-art approach in RTFM, ECoG-EM has previously been tested on motor localization tasks [37], but not on language localization. To have a fair comparison, we ran ECoG-EM on the frequently tested sub-bands - $\mu, \beta$ and high-$\gamma$, as well as on the frequency bands beyond and upto 350 Hz. In order to perform a fair comparison, ECoG-EM results from only the randomly selected NRCs and all the PRCs were used. The results are shown in 3.6. While ECoG-EM approach provides a high specificity, it has a much lower accuracy and sensitivity. This is a strong validation of our hypothesis that inter-channel comparison is a promising approach as compared to the intra channel (using baseline) approach.

3.4 Summary

Discriminating between the response in the eloquent language cortex regions based on the associated task is a challenging problem. In the current study, we developed a novel machine learning-based framework towards the ECoG-based eloquent cortex localization. This approach compared the positive response and negative response channels in regards to language response instead of comparing to a baseline recording. We analyzed the complete ECoG signal frequency spectrum to identify signal characteristics for mapping language cortex. This methodology resulted in 78% accuracy on channel classification which is a significant improvement in comparison to the 55% accuracy of the state of the art ECoG-based functional mapping.
CHAPTER 4: IMPROVING LANGUAGE CORTEX LOCALIZATION USING DEEP LEARNING APPROACHES

Related Publication:


4.1 Overview

Having established the efficacy of machine learning methods in ECoG signal classification, we extend our study by designing deep learning algorithms instead of conventional ML classifiers for the channel response classification task. We also explore all the underlying architecture designs for ECoG data and provide comprehensive comparisons for better data analysis algorithms. Specifically, we combine multimodal signal data, time-domain and frequency-domain, and fuse the features via learning. In the following sections the proposed algorithms are explored in detail.

4.2 Materials and Methods

An overview of the proposed architecture is illustrated in Figure 4.1. First, based on the evidences that both time and frequency domain signals carry important information for ECoG signal classification, we intend to explore both signals with new (potentially better) classification systems and provide a learning based fusion algorithm to improve the overall process. We pre-processed the ECoG signals using temporal filtering such that the selected samples were synchronized with the
start and end points of the task resulting in equal length blocks. We then divided the equal length ECoG signal blocks into overlapping sub-blocks of data. Our aim was to learn discriminative signal patterns and eventually reduced the computational load (Step 1). We learned different sets of signal features independently: frequency domain (i.e. auto-regression) and time-domain (deep learning-based features) in Step 2 and Step 3, respectively. After we combined the learned features, we trained a recurrent neural network (RNN), a class of deep learning algorithm suited for analyzing sequential data, to classify sub-blocks of signals (Step 4). Finally, we used the majority voting technique to combine these sub-block labels and determine an overall (PRC or NRC) channel label (Step 5). In the following sub-sections, we will describe each module of our proposed system in detail.

4.2.1 Dataset

We recruited eleven patients with drug-resistant epilepsy, who underwent pre-surgical evaluation with ECoG grid implantation. All patients provided their written informed consent to participate
in this study. The patients were teenagers and adults with an average age of 23.18 ± 11.61 years. Varying number of electrodes were tested for the patients for a total of 637 electrodes across all patients. (See Table 4.1 for summary of the patient demographics). Patients were recruited under IRB approved protocol 276487. Only patients 13 years or older with English as the dominant language and already assigned to undergo ESM as part of a standard of care pre-surgical evaluation are included in this study.

Table 4.1: Patient demographics, clinical information, grid placement, and information about the number of analysed channels (electrodes) are summarized. All study participants were left hemisphere language dominant for language.

<table>
<thead>
<tr>
<th>Subject #</th>
<th>Age (yrs)</th>
<th>Sex</th>
<th>Epilepsy Focus</th>
<th>Grid Placement</th>
<th>Epilepsy Onset (yrs)</th>
<th>Channels Tested/PRC / NRC</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>19</td>
<td>M</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>16</td>
<td>54 / 22 / 32</td>
</tr>
<tr>
<td>2</td>
<td>33</td>
<td>F</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>10</td>
<td>32 / 5 / 27</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>M</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>6</td>
<td>127 / 16 / 111</td>
</tr>
<tr>
<td>4</td>
<td>22</td>
<td>F</td>
<td>Parietal</td>
<td>Lateral</td>
<td>20</td>
<td>30 / 19 / 11</td>
</tr>
<tr>
<td>5</td>
<td>32</td>
<td>F</td>
<td>Temporal</td>
<td>Bilateral</td>
<td>26</td>
<td>48 / 10 / 38</td>
</tr>
<tr>
<td>6</td>
<td>52</td>
<td>M</td>
<td>Temporal</td>
<td>Lateral</td>
<td>30</td>
<td>48 / 5 / 43</td>
</tr>
<tr>
<td>7</td>
<td>15</td>
<td>M</td>
<td>Temporal-Parietal</td>
<td>Lateral</td>
<td>12</td>
<td>43 / 10 / 33</td>
</tr>
<tr>
<td>8</td>
<td>17</td>
<td>M</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>7</td>
<td>72 / 20 / 52</td>
</tr>
<tr>
<td>9</td>
<td>15</td>
<td>F</td>
<td>Medial-Frontal</td>
<td>Lateral</td>
<td>7</td>
<td>80 / 19 / 61</td>
</tr>
<tr>
<td>10</td>
<td>17</td>
<td>M</td>
<td>Frontal-Temporal</td>
<td>Lateral</td>
<td>16</td>
<td>63 / 10 / 53</td>
</tr>
<tr>
<td>11</td>
<td>13</td>
<td>F</td>
<td>Temporal</td>
<td>Lateral</td>
<td>2</td>
<td>40 / 6 / 34</td>
</tr>
</tbody>
</table>

4.2.2 Pre-processing ECoG Signal (Step 1)

As a first step of preparing the recorded data for deep learning-based analysis, non-task/control time points in the signal were eliminated using temporal filtering. We recorded the signals from the ECoG grid electrodes for a few seconds prior to the task initiation. Similarly, we recorded the signals for a few seconds post task completion. This led to padding of data, that may not contain useful information but was necessary to ensure that signals were located at the boundary locations.
Thus, as a first step of preparing the recorded data for ML based analysis, non-task/control time points in the signal were eliminated: these correspond to the spontaneous activity recording before the 0-min and any trailing signals at the end of the experiment. We recorded the ECoG signals over a few minutes at a sampling rate of 1200Hz, that was a few hundred thousand data points per electrode (1200 samples/s × 300s). Due to this large sample size, we used the sliding window approach to generate features and analyzed the whole signal through these feature representations. One drawback of this approach might be a potential loss of long term dependencies; however, we believe that we have minimized the potential loss of periodic information because of the paradigm involving different stories and a wise choice of data preparation.

### 4.2.3 time-domain RNN (Step 2)

Our goal was to find discriminative signal patterns from ECoG signals, which are time varying and non-stationary 1D sequences. They are non-stationary because task based ECoG recordings can have signal statistics which depend on the time relative to the events. Inspired by the effectiveness of recurrent neural networks in sequence classification tasks in different domains ([80, 81, 82]), we have developed RNN based deep neural network algorithms to extract discriminative features from time-domain ECoG data. We hypothesized that limitations of the conventional spectral (frequency-based) or time-based signal analysis methods can be overcome with RNN based methods. In RNNs outputs from previous time steps are taken as input for the current time step, thereby forming a directed cyclic graph. RNNs thus learn the relationships in sequential data thereby retaining higher contextual information.

The ECoG signal consists of over a thousand samples every second; to learn signal characteristics from such a noisy data, we first used popular EEG features [83, 84, 85]. A sliding window approach was applied to extract features, which were then concatenated into a single feature vector.
to represent the control and active-task blocks in the signal. The extracted features included mean, skew, kurtosis, peak to peak value, and Hjorth values.

- **Mean** – The mean signal intensity within the window of a channel’s signal block is used as a feature. This can help detect the change in activation across windows and is measured as:

\[
m_i = \frac{\sum_{j=t}^{t+n} X_j}{n},
\]

where \(i = i^{th}\) window, \(n=\) size of window, \(t=\) time point in the window of channel signal \(X\).

- **Skew** – This is a measure of the symmetry in a distribution and is measured as

\[
s_i = \frac{\sum_{j=t}^{t+n} (j=t)(t+n)(X_j - m_i)^3}{\sigma^3},
\]

where \(\sigma\) is the standard deviation within the window \(i\).

- **Kurtosis** – This is a measure of how peaked around the mean a distribution is and is measured as

\[
k_i = \frac{\sum_{j=t}^{t+n} (X_j - m_i)^4}{\sigma^4}
\]

- **Peak-to-peak (P2P)** – This measures the difference between the minimum and maximum values in the distribution and is measured as

\[
p2p_i = \max X_{t:t+n} - \min X_{t:t+n}
\]

- **Hjorth features** – define the signal in terms of amplitude, time scale and complexity.

  - **Activity** – this is a measure of the mean power of the signal and is computed as the variance of the signal.
Figure 4.2: Deep network structure for extraction of time-domain features from the input signal. Note that these features are combined with frequency domain features (Auto-Regressive) later for final prediction of the channels.

- Mobility – this represents the mean frequency and is defined as the square root of the ratio of the variance of the first derivative and the amplitude.

- Complexity – this represents the frequency change and is defined as the ratio between the mobility of the first derivative of the signal and the mobility of the signal itself.

The time-domain features were fed to the learning module illustrated in Figure 4.2. The complete ECoG signal contained both control and active task signals, thus; the sub-blocks of control signals were ignored and the input to this time-domain module was sub-blocks of active task signals. Recently, 1D convolutional networks have been shown to perform well in time series forecasting and classification tasks [86, 87]. We designed the module to have two paths comprising of 1D convolutional layers and long-short term memory blocks. LSTM, introduced in 1997 [88], is a type of RNN that has the ability to learn long-term dependencies of data. In literature, LSTM and its variants have primarily been used in 1D sequence classification tasks [89, 90] and prediction tasks [91, 92]. In our experiments, we used multiple LSTM layers in different exploratory configurations.
to learn a more complex feature representation of the input signal.

### 4.2.4 Frequency Domain Features (Step 3)

One of the objectives of our study was to analyze multi-domain (time and frequency independently) and hybrid-domain (time and frequency combined) signal characteristics. This kind of thorough comparison has never done before to the best of our knowledge for ECoG-FM. In this step (Step 3), we focused on the spectral characterization of signals. Conventional ECoG signal classification approaches are based on frequency-domain, where spectral analysis of the signal is performed to identify the channel response. Traditionally, spectral estimation of the signals is performed by fitting a parametric time-domain model to the ECoG signals. One of the most commonly employed models/approaches in this category is the autoregressive model. An AR model for a discrete signal is represented as,

$$
x[n] = - \sum_{k=1}^{p} a_p[k] x[n-k] + w[n],
$$

(4.5)

where $a_p[k]$ are the AR coefficients, $p$ is the order of the AR model, $w[n]$ is a zero mean white noise process with a variance. Once the model in Eq. 4.5 is solved, the resulting AR parameters were used for characterization of the ECoG signal from frequency-domain perspective. Methods to solve for the AR parameters are diverse and can be classified into three main categories[93]:

- **Correlation Function Estimation method** - AR parameter estimation from autocorrelation sequence estimates of Yule-Walker equations [93]. These equations relate the autocorrelation sequence for lags 0 to $p$, to the variance of a driving noise signal. This approach has the benefit of producing a stable model, however, in some special cases, involving nearly periodic signals, it may produce incorrect estimates.

- **Reflection coefficient estimation methods** - AR parameters are constrained to satisfy a recur-
sive relationship. The Levinson-Durbin recursion [93] relates the AR parameters of order \( p \) to the AR parameters of order \( p - 1 \) as

\[
a_p[n] = a_{p-1}[n] + k_p a_p^*[p-n],
\]

(4.6)

where, \( k_p \) is the reflection coefficient. The benefit of this approach is that it produces stable models.

- Least Squares Linear Prediction Estimation methods - AR parameters are estimated by the minimization of the forward (predicting the future) and backward (predicting the past) linear prediction squared errors. The minimization can be done separately or combined. The advantage of this approach is that it minimizes the residual variance however, this does not directly translate to minimizing the variance of the prediction error.

We used the reflection coefficient estimation-based methods, which require the estimation of \( k_p \) from the autocorrelation function for lags 0 to \( p - 1 \). These approaches are based on minimizing the least square error of the forward (predicting the future) and backward (predicting the past) linear prediction with respect to the reflection coefficient. Other known approaches in this category includes the geometric algorithm, which aims to minimize the geometric mean of the squared error predictions, the harmonic algorithm that minimizes the arithmetic mean instead of the geometric mean, and the maximum likelihood approach. Among these solutions, we used the popular Harmonic algorithm, also known as Burg method [93] which produces comparable estimates to the least square linear prediction estimation method. In the Burg estimation approach, the reflection coefficient \( k_p \) was found by,

\[
k_p = \frac{-2 \sum_{n=p+1}^{N} e_{p-1}^f[n] e_{p-1}^{bs}[n-1]}{\sum_{n=p+1}^{N} |e_{p-1}^f[n]|^2 + \sum_{n=p+1}^{N} |e_{p-1}^{bs}[n-1]|^2}
\]

(4.7)
where $e_p^f, e_p^b$ are the forward and backward prediction errors and are expressed as

$e_p^f[n] = e_{p-1}^f[n] + k_p e_{p-1}^b[n-1]$  \hspace{1cm} (4.8)

$e_p^b[n] = e_{p-1}^b[n-1] + k_p^* e_{p-1}^f[n]$  \hspace{1cm} (4.9)

Once the errors in Eqs. 4.8 and 4.9 were computed recursively, the reflection coefficient was computed by minimizing the arithmetic mean of the squared forward and backward prediction errors. This allowed us to solve for the final AR parameters, $a_p[k]$ in Eq. 4.6, which were used to characterize the ECoG signals from frequency-domain perspective.

### 4.2.5 Fusion for Hybrid Domain (Step 4)

Using LSTMs in Step 2, we learned a different set of features (i.e., time-domain) than the AR features that were generated in Step 3. In domain fusion step (Step 4), these two (largely) complementary features were combined to obtain a hybrid signal representation model with a new deep network setting, Domain Fusion Network (DFN) (See Figure 4.3). Although the merging of the two feature vectors can be done in multiple ways, we used a concatenation approach to get full benefit of each domain (time vs. frequency).

In concatenation, we assumed independence of features; hence, we did not use element-wise multiplication or other approaches for data merging. Since convolution helps identify local patterns and reduce redundant information in the data, the complete feature vector (after concatenation) was then passed through multiple layers of 1D convolutions with an activation function, to weight each feature based on its contribution to the classification problem (PRC vs. NRC). Following the 1D convolution layers, the output feature maps were spatially averaged using Global Average
Figure 4.3: Deep network structure of the fusion module. AR (i.e., frequency) features (orange) and time-domain features (green) are concatenated and classified.

Pooling [94], making the DFN more robust to spatial translations of the input data and introducing structural regularization to the feature maps. Finally, we inserted a single fully connected layer into the DFN and used a sigmoid activation to perform the final classification.

4.2.6 Majority Voting (Step 5)

The output of the domain fusion model was a label for the input signal, which was a sub-block. Signal from each channel/electrode was made up of hundreds of sub-blocks of the signals with reasonable overlapping. Therefore, for classifying a channel as either PRC or NRC, we hypothesize that the output that is observed more commonly is assigned as the final label. For this purpose, we apply majority voting on the output for each sub-block. For instance, if a channel included 354 sub-blocks and more than 50% of sub-blocks indicated a positive response, that channel was labelled as a PRC. As a rule, whenever the number of negative and positive responses are equal,
the channel will not be assigned any label. Although, we did not observe any such channel in our experiments.

4.3 Experiments and Results

4.3.1 Task Paradigm

ECoG signals from the implanted subdural grids are split into two streams: one for continuous clinical seizure monitoring and the other for ECoG-FM (Figure 1A). The tool used to record the incoming ECoG signal was BCI2000 [72]. A baseline recording of the cortical activity was first acquired to capture the “resting-state” neuronal activity. Following this baseline recording step, paradigms similar to those employed in ESM or functional magnetic resonance imaging (fMRI) were used to record the task-related ECoG signal for functional mapping [95]. Figure 3.2 shows one such paradigm, mimicking the exact details of the experimental setup we have used for the language comprehension task. Alternate 30 second blocks of ECoG data during “control” and “active” conditions were recorded continuously at a fixed sampling rate of 1200 Hz. For the language comprehension task, the active condition implies listening to a story, while the control task involves listening to broadband noise [96]. For the active condition (i.e., listening to a story), a different story was selected for each block in order to keep the patient attentive and responsive. Both control and active sequences would activate sense of hearing, but the story listening task will particularly activate the language function. We hypothesize that this would suffice in eliciting the desired response for mapping eloquent cortex related to language function and our results have verified this hypothesis. For this purpose, the system recorded information from 128 ECoG channels (128 electrodes in Figure 3.2) by using g.USBamp bio-signal amplifiers (g.tec Medical Engineering GmbH, Austria) with subdural ground and reference electrodes.
4.3.2 Training Paradigms

Our overall goal was to successfully (and automatically) identify positive response channels (PRCs) and negative response channels (NRCs) in ECoG-FM data using new machine learning models, specifically based on deep neural networks. The ground truth (i.e., reference standard) was inferred from the gold standard ESM results. Owing to the large imbalance in the number of PRCs and NRCs (NRCs outnumbering PRCs by 3:1), we randomly selected equal number of NRCs to balance the data and avoid potential data imbalance problem when training deep learning models. Each channel’s signal comprised of blocks of active task data and control data, where each active task block was from a different story. The discriminative power of these stories in the classification task was unknown. There is a possibility that features from one story could play a more significant role than others. Additionally, the discriminative power of any particular feature is unknown. To ascertain the role of these, we divided our experimental evaluation approaches into three main categories for data classification. Each approach depended upon the way active task data was included, and the features used in the training process. This structured experimental procedure helped us in determining the usefulness of each component of the signal and provided insights into the response of brain regions (through channel responses) to different signals. We performed experiments with different features and architectures. A detailed block diagram showing the steps involved in the proposed methodology is shown in Figure 4.4.

4.3.3 Our Proposed Deep Network Architectures

In task-based experiments, a response is generally expected only during the active task period and not in the control or rest period. We used fully convolutional network (FCN) and LSTM architectures in the time-domain module, since these have shown success in various time-series classification problems [97, 98]. We built our network by first analyzing the effect of using time-domain
Figure 4.4: A overview of the steps involved in the proposed methodology showing ECoG data recording in response to story listening task, time and frequency domain feature extraction, different training paradigms and majority voting to classify positive and negative response channels. AT: Active Time, AR: Auto Regressive; CT: Control Task, EF: Early Fusion, LF: Late Fusion, PRC: Positive Response Channel, NRC: Negative Response Channel.

features during the active task (represented as Active Time- AT). We tested our proposed time-domain module by varying the network. We used an FCN (represented as AT$^1$) and then added LSTM module to the network (represented as AT$^2$). For frequency domain analysis, we added the auto regressive (AR) features to the frequency domain module by passing it through a fully connected layer (represented as AT-AR$^1$). Figure 4.5(a) shows the architectures (the superscripts indicate the variation within an architecture) including AT$^1$, AT$^2$ and AT-AR$^1$. In the domain fusion module, we tested different combinations of 1D convolutions (represented as AT-AR$^2$) and fully connected layers (represented as AT-AR$^3$). We also varied the depth of the frequency domain module by adding an additional fully connected layer in the network (represented as AT-AR$^4$). The network structures including AT-AR$^2$, AT-AR$^3$, and AT-AR$^4$ are shown in Figure 4.5(b). We empirically determined the number of epochs required to train the network such that to avoid over-
fitting. Our experimental paradigms used time and frequency domain features individually and also in a hybrid manner (combined). We also analyzed the effect of active and control task data.

4.3.4 Model Validation

In supervised machine learning approaches, where a model is trained using ground-truth labels, the goal is to maximize predictive accuracy. However, therein lies the risk of memorizing the data rather than learning the optimal features. This problem of memorizing the data or learning the structure of the data to be the noise in the data is often referred to as overfitting [99]. It is important for a classification model to be able to generalize to unseen data and avoid the problem of overfitting. The method of testing how the analysis/model generalizes to an independent test dataset is known as cross-validation. When a completely independent dataset is not available, as is generally the case, the available data is split into training data and validation/test data. There are different types of cross-validation approaches such as leave-one-out cross-validation, hold-out method, k-fold cross-validation, to name a few [100].

Shuffle-split Cross Validation Previously, due to the time-consuming nature of training a deep learning model, we applied the hold-out method to validate our proposed models [101]. In this method, the model is trained on a part of the available data, while the remaining data is held for testing/validating the model. For effectively testing the generalization and robustness of our proposed models, we validated them using the shuffle-split cross-validation approach. In the shuffle-split cross-validation method, the data is randomly sampled and split into training and testing splits iteratively, similar to the hold-out method. The results are averaged across the number of iterations. This can be seen as repeating the hold-out method k times, such that the data for training and validation is randomly sampled each time. The use of shuffle-split method allows sampling different data combinations rather than a single sampling as in k-fold cross-validation. Since the blocks were randomly assigned to the training and test folds, we ensured that no data from an electrode (channel) was represented in both training and testing
folds simultaneously. Hence, if a block of data was assigned to a particular set (training/test), then all blocks belonging to that channel were assigned to the same (training/test) set. This ensured a fair evaluation with better generalization accuracy and helped in avoiding overfitting. For 30-fold cross validation, we repeated the experiments 30 times and used each of these distinct and non-overlapping training-testing sets to evaluate our model accuracies. Prediction accuracy was then calculated by averaging the results of these 30 experiments.

4.3.5 Training on individual features on active task data (Training Paradigm-I)

In this approach, we assumed that the channel response was similar for different stimuli (story) used in this study. Our experimental paradigm consisted of 5 different stories and thus, in this approach, no distinction was made with regards to the story. All of the active task data (i.e., five different story tasks) from a channel were used together for training the network. Among the time-domain features (See Section: Methods), we found using random forest method that activity feature gave the best results. Therefore, all our proposed deep learning architecture (Figure 5) were first tested using the activity feature (Table 4.2). The addition of LSTM improved the performance of the time-domain module. This was further improved by the addition of the frequency domain features using the domain fusion module. We found that increasing the depth of the frequency domain module did not have any obvious benefit in classification performance. We also tested our hypothesis that the story listening task (active task) was more discriminative in identifying the eloquent cortex. To compare information present in the active and control task data, we replicated our best performing models (ATAR$^2$ and AT-AR$^3$) and fed it with control task data (represented as - CTAR$^2$ and CT-AR$^3$, where CT represents control time). We observed that sensitivity of the control data model was lower than that of active data model, indicating a lower discriminative power (Table 4.3) and confirmed our hypothesis.
Figure 4.5: Deep network structure of the various AT and AT-AR models. AT- Active Time, AR-Autoregressive.
To identify the best features for the channel classification task, we fed the best performing model (AT-AR\(^3\)), with different hand-crafted features and performed cross-validation. The performance with different features was found to be similar (Table 4.4). The mobility feature showed the best performance with high sensitivity and accuracy compared to the other features.

4.3.6 Training with multiple features on active task data – feature fusion (Training Paradigm-II)

We hypothesized in this experiment that different features can provide complementary information and can be combined to enhance the model performance. The top performing features from indi-

---

Table 4.2: Channel classification accuracy for different network architectures. AT- active time, AR – auto regression.

<table>
<thead>
<tr>
<th>Model</th>
<th>Block Accuracy (%)</th>
<th>Sensitivity %</th>
<th>Specificity %</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT(^1)</td>
<td>63.33</td>
<td>83.33</td>
<td>58.33</td>
<td>70.83</td>
</tr>
<tr>
<td>AT(^2)</td>
<td>65.83</td>
<td>66.66</td>
<td>74.99</td>
<td>70.83</td>
</tr>
<tr>
<td>AT-AR(^1)</td>
<td>77.5</td>
<td>91.67</td>
<td>58.33</td>
<td>74.99</td>
</tr>
<tr>
<td>AT-AR(^2)</td>
<td>80.83</td>
<td><strong>99.99</strong></td>
<td>66.67</td>
<td>83.33</td>
</tr>
<tr>
<td>AT-AR(^3)</td>
<td><strong>83.33</strong></td>
<td>91.67</td>
<td><strong>74.99</strong></td>
<td><strong>83.33</strong></td>
</tr>
<tr>
<td>AT-AR(^4)</td>
<td>83.33</td>
<td>91.67</td>
<td>74.99</td>
<td>83.33</td>
</tr>
</tbody>
</table>

Table 4.3: Comparing channel classification accuracy for active and control data. AT- active time, AR – auto regression, CT- control time.

<table>
<thead>
<tr>
<th>Model</th>
<th>Block Accuracy (%)</th>
<th>Sensitivity %</th>
<th>Specificity %</th>
<th>Accuracy %</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-AR(^2)</td>
<td>75.93</td>
<td>77.97</td>
<td><strong>79.66</strong></td>
<td>78.81</td>
</tr>
<tr>
<td>AT-AR(^3)</td>
<td><strong>78.13</strong></td>
<td><strong>86.44</strong></td>
<td>72.88</td>
<td><strong>79.66</strong></td>
</tr>
<tr>
<td>CT-AR(^2)</td>
<td>73.9</td>
<td>76.27</td>
<td>72.88</td>
<td>74.58</td>
</tr>
<tr>
<td>CT-AR(^3)</td>
<td>73.05</td>
<td>72.88</td>
<td>72.88</td>
<td>72.88</td>
</tr>
</tbody>
</table>
Table 4.4: Channel classification performance parameters (with mean and variance) for active task data with individual hand-crafted time-domain features using the AT-AR\textsuperscript{3} model (training paradigm-I).

<table>
<thead>
<tr>
<th>Features</th>
<th>Sensitivity $\mu \pm \sigma$</th>
<th>Specificity $\mu \pm \sigma$</th>
<th>Accuracy $\mu \pm \sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mean</td>
<td>83.33 ± 10.09</td>
<td>81.11 ± 9.61</td>
<td>82.22 ± 6.71</td>
</tr>
<tr>
<td>Skew</td>
<td>81.67 ± 9.47</td>
<td>81.67 ± 9.95</td>
<td>81.67 ± 5.54</td>
</tr>
<tr>
<td>Kurtosis</td>
<td>82.78 ± 11.97</td>
<td>79.17 ± 10.03</td>
<td>80.97 ± 7.81</td>
</tr>
<tr>
<td>P2P</td>
<td>82.22 ± 10.91</td>
<td>80.00 ± 9.77</td>
<td>81.11 ± 7.11</td>
</tr>
<tr>
<td>Activity</td>
<td>82.22 ± 9.31</td>
<td>79.44 ± 10.70</td>
<td>80.83 ± 7.88</td>
</tr>
<tr>
<td>Mobility</td>
<td><strong>84.17 ± 9.22</strong></td>
<td><strong>81.11 ± 8.85</strong></td>
<td><strong>82.64 ± 6.55</strong></td>
</tr>
<tr>
<td>Complexity</td>
<td>84.17 ± 8.70</td>
<td>79.44 ± 10.70</td>
<td>81.81 ± 6.67</td>
</tr>
</tbody>
</table>

Table 4.4: Channel classification performance parameters (with mean and variance) for active task data with individual hand-crafted time-domain features using the AT-AR\textsuperscript{3} model (training paradigm-I).

Individual feature training (Table 4.4) – mobility, skew, mean, peak-to-peak (P2P), were used to test the hypothesis. The other three features were not used on the basis that they had a marginally lower specificity. Different approaches to feature fusion were tested in the form of early fusion and late fusion. In the early fusion approach, different features were used as input channels to the best performing network architecture (AT-AR\textsuperscript{3}). In late fusion, we tested two different approaches: first, separate time-domain models were retrained for each hand-crafted feature, and a single frequency domain module was trained. The domain fusion module was used to combine these time and frequency domain modules (represented as AT-AR\textsuperscript{3}-LF\textsuperscript{1}). Secondly, we experimented by combining the frequency domain module prior to the feature fusion layer (represented as AT-AR\textsuperscript{3}-LF\textsuperscript{2}). The performance of these models is presented in Table 4.5.
Table 4.5: Channel classification performance parameters for different approaches using feature fusion from time and frequency domains (training paradigm-II). AT- Active time, AR- autoregressive, EF- early fusion, LF- late fusion.

<table>
<thead>
<tr>
<th>Features</th>
<th>Sensitivity $\mu \pm \sigma$</th>
<th>Specificity $\mu \pm \sigma$</th>
<th>Accuracy $\mu \pm \sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>AT-AR$^3$ – EF</td>
<td>79.99 ± 9.76</td>
<td>83.89 ± 11.37</td>
<td>81.94 ± 6.48</td>
</tr>
<tr>
<td>AT-AR$^3$ – LF$^1$</td>
<td>82.22 ± 10.70</td>
<td>80.83 ± 9.42</td>
<td>81.53 ± 7.43</td>
</tr>
<tr>
<td>AT-AR$^3$ – LF$^2$</td>
<td>85.83 ± 7.80</td>
<td>80.27 ± 11.07</td>
<td>83.05 ± 6.35</td>
</tr>
</tbody>
</table>

Table 4.6: Channel classification performance (mean and variance) for individual active task data, using each story independently (training paradigm-III).

<table>
<thead>
<tr>
<th>Features</th>
<th>Sensitivity $\mu \pm \sigma$</th>
<th>Specificity $\mu \pm \sigma$</th>
<th>Accuracy $\mu \pm \sigma$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Mobility$_{story}$</td>
<td>85.55 ± 10.07</td>
<td>79.44 ± 10.91</td>
<td>82.5 ± 6.21</td>
</tr>
<tr>
<td>Mean$_{story}$</td>
<td>83.61 ± 10.42</td>
<td>78.88 ± 11.53</td>
<td>81.25 ± 7.35</td>
</tr>
<tr>
<td>Activity$_{story}$</td>
<td>83.05 ± 9.73</td>
<td>80.55 ± 11.45</td>
<td>81.80 ± 6.93</td>
</tr>
</tbody>
</table>

4.3.7 Training with individual features on individual stories/active task data (Training Paradigm-III)

In previous experiments so far, we assumed that the channel responds in a similar manner to different stimuli (stories). However, it is plausible that the channel may respond differently to different stimuli. In this training approach, we now assumed that the channel responds differently to different stimuli (stories). The task paradigm consisted of 5 different stories corresponding to 5 different task blocks. We hypothesized that PRCs respond differently to the NRCs for each of these stories. We separated the signals based on the story and train the network in a similar manner as in section 3.6. Each story was trained with its own time-domain and frequency domain modules using the AT-AR$^3$ network. These different networks were then combined and fed through a fully
connected layer. Deep networks with different features as inputs were trained and the performance is compared in Table 4.6, where the features column shows the particular value (mobility, mean, and activity) computed for each story.

4.4 Summary

In this paper, we proposed novel deep learning architectures to classify the channel response of ECoG signals. The results showed the state-of-the-art classification accuracy of 83.05% with high specificity and sensitivity of 80.3% & 85.8%, respectively, in determining whether the channel was positive (has a response) or negative (has no response) in relation to the task stimulus. The different features and fusion approaches have given us the flexibility in maximizing different metrics, where as an example we can improve the specificity to 83.9% (with a 1% drop in accuracy). In general, with AT-AR$^3$-LF$^2$ is our best performing model with values >80% for all performance metrics including accuracy, sensitivity and specificity. In a feasibility study towards using machine learning for ECoG-FM, a random forest classifier was used in detecting positive and negative response channels with an accuracy of 78% [102]. Traditionally, the accuracy of ECoG-FM is high for mapping sensory and motor function, but relatively low for language modality. On an average, ECoG functional language mapping had a lower sensitivity (62%) and higher specificity (75%) to detect language-specific regions (for a comprehensive review, see [95]). This is in contrast to the results for hand motor (100% sensitivity and 79.7% specificity) and hand sensory (100% sensitivity and 73.87% specificity) ECoG-based mapping [74]. The results of our current study demonstrate that the accuracy for mapping eloquent cortex using ECoG-FM can now be comparable to both sensory and motor ECoG-FM accuracies. The language ECoG-FM accuracy values we have achieved are the highest among those reported so far [40]. Although a number of studies have demonstrated successful utilization of the ECoG-FM as a complimentary tool for ESM [42, 43]. There was not
enough evidence to support the use of ECoG-FM as a stand-alone methodology for functional language mapping due to its relatively low accuracy compared with ESM [74, 95]. The outcome of our research has indicated the potential of ECoG-FM, to be considered as a stand-alone modality for eloquent language cortex localization. Our experimental results show performance comparable to what is achieved with ESM for eloquent cortex mapping. Based on this we believe that with our trained models the proposed scheme can be used independent of ESM in surgery planning. To establish the method as ready to be used in clinical practice, a patient/subject-wise analysis followed by blind test evaluation on the models will be performed as we continue to collect more clinical data.
CHAPTER 5: SMART THINKERS

Related Publication:


5.1 Overview

In this study, we move from 1D signals to 3D and 4D to explore more advanced algorithms handling multidimensional and multimodal signals of brain imaging. More specifically, we explore the functional and anatomical differences between novice and professional chess players because this gives us an opportunity to test computational algorithms when brain signals are extremely similar (as there is no disease), and still identify discriminatory signal features. Towards this end, we utilize resting state functional magnetic resonance images (MRI) to generate functional connectivity (FC) information. We also use T1 weighted MRI to estimate morphometric similarity network (MSN) connectivity information. We further combine the functional and anatomical features into a
new connectivity matrix, the function morphometric similarity (FMS) connectome. To reduce the dimensionality of extracted information and perform successful classification, statistical features selection is employed and support vector machines are used for classification. Single modality and multi-modality classification is performed. In the following sections, the approach is presented in detail.

5.2 Materials and Methods

5.2.1 Data

The chess masters dataset was used [24]. The dataset consists of both high-resolution anatomical images and T2-weighted rs-fMRI images acquired at the MR Research Center of West China Hospital of Sichuan University, Chengdu, China. The anatomical images were acquired with repetition time (TR), 1900 ms; echotime (TE), 2.26 ms; flip angle=12; whole head: 176 sagittal slice; slice thickness, 1.0 mm; voxel size=111 mm). rs-fMRI images were obtained at TR, 2000 ms; TE, 30 ms; flip angle=90; whole head: 30 axial slice, each 5 mm thick (without gap); voxel size=3.753.755 mm with subjects instructed to relax with their eyes open and visual fixation on a cross-hair centered in the screen. The scan comprised of 205 volumes in total.

Preprocessing T1 images

For the extraction of anatomical features from the high-resolution MRI images, Freesurfer [103] was used in our experiments. Freesurfer performs surface parcellation via the following steps:

- The MRI volume is registered with the MNI305 atlas using an affine registration.
- Bias field correction and skull stripping are performed.
• Cutting planes approach is used to remove white and gray matter.

• An initial white surface, is generated for each hemisphere which refined to follow the intensity gradients between the white and gray matter. From this surface, the pial surface is generated by following the intensity gradients between the gray matter and CSF and surface labeling is done as in [104].

Parcellation of the cortex for each subject was based on the 17 Network functional parcellation [105]. The metrics of interest, extracted from the cortical parcellations are the following:

1. Surface Area (SA)

2. Gray Matter Volume (GM)

3. Cortical Thickness - is measured as the shortest distance from the white matter to the pial surface.
   - Average Cortical Thickness ($CT_{avg}$)
   - Standard Error of Cortical Thickness ($CT_{sd}$)

4. Curvature - is defined as the inverse of the radius of the circle passing through a point $p$ and a pair of infinitesimally close points on the same curve.
   - Rectified Mean Curvature ($MC$) - is an extrinsic measure of the curvature of a surface and is measured as
     \[ K_M = \frac{\kappa_1 + \kappa_2}{2}, \]

     where $\kappa_1, \kappa_2$ are the maximum and minimum curvatures of the surface.
   - Rectified Gaussian Curvature ($GC$) - is an intrinsic measure of the curvature of a surface and is measured as
     \[ K_G = \kappa_1 \times \kappa_2. \]
5. Folding Index ($FI$)- gives a measure of the local gyrification and is computed as

$$FI = |\kappa_1| \times (|\kappa_1| - |\kappa_2|).$$

6. Intrinsic Curvature Index ($CI$) - is measured as the maximum intrinsic curvature across all points in the surface.

**Preprocessing rs-fMRI**

For the extraction of functional connectivity networks, AFNI software is used [106]. The first 5 volumes of the rs-fMRI image were discard to account for the time taken for molecules to reach steady-state upon application of the magnetic field. Following this, slice timing correction was performed to account for the time difference when each slice was acquired. Motion correction was then performed by aligning all volumes to a reference volume to account for the head movement during the course of the recording. The functional image was then registered/aligned with the corresponding anatomical (T1 image) and smoothing was performed to enable better group-wise analysis. The data was then mapped onto the surface. White matter and ventricle maps from Freesurfer outputs were used as tissue regressors to detect non-BOLD signals in the data. Motion threshold of 0.3mm was set and volumes/time-points with motion that exceeded the threshold were censored. Spatial blurring was then performed to further reduce the noise and add up coherent signals locally. Finally band-pass filtering was performed to further isolate the noise with a frequency range of $0.01\text{Hz} - 0.1\text{Hz}$. 
5.2.2 Morphometric Similarity Network

Morphometric Similarity Network (MSN), is the connectivity matrix designed from anatomical measures. Anatomical measures are extracted from parcellation regions. The MSN generated differs from that used in [55] in that the authors use diffusion metrics such as fractional anisotropy, mean diffusivity and magnetization transfer in addition to the anatomical features SA, GM, CT, CI, FI, MC and GC. Here, we use only the anatomical features and separate the cortical thickness measure into two: the mean cortical thickness \((CT_{avg})\) and the standard error in cortical thickness \((CT_{sd})\) for a total of 8 features. Correlation requires feature vectors of 2 or more dimensions. We thus, build several MSNs using combinations of 3 or more anatomical feature vectors. The features have different range of values with the GM and SA \(> 100\) but GI, FI, etc \(< 10\). Thus, each feature \(A\) of a region \(i\) is z-score normalized across all regions as:

\[
A_i^z = \frac{A_i - \mu_A}{\sigma_A}
\]  

Figure 5.1: Generating the Morphometric Similarity Network (MSN) from the anatomical images. Morphometric measures are extracted from the cortical surface of the T1 image and pearson’s correlation is used to build the MSN.
Pearson’s correlation between the feature vectors of the different regions (17 networks) is computed to build each MSN. Since the atlas used is the Yeo 17 network parcellation map, there are 34 regions of interest across both hemispheres. The dimensions of the MSN are $34 \times 34$.

### 5.2.3 Functional Connectivity

Functional Connectivity (FC) is the correlation of time series of different voxels or different group of voxels. FC can be computed from task based or resting-state fMRI images. In resting-state fMRI (rs-fMRI), no external stimulus is provided and the BOLD signal is recorded for the duration of no stimulation while in task-based signal, an external stimulus (finger-tapping, listening to story, etc.) is established alternating with no stimulus as the experimental design paradigm. For this study, the resting-state paradigm is preferred so as to study the general functional differences between the two groups. Once the image has been preprocessed as in Section 5.2.1, the Yeo 17 network parcellation map was overlayed and the average time-series in each functional parcel was extracted. This step amounts to averaging the time-series across all nodes in the parcel. Pearson’s correlation was then used to compute the correlation between the different parcels and generate the functional connectivity network. The overall network generation process in shown in Fig. 5.1.

### 5.2.4 Functional Morphometric Similarity (FMS) Connectome

While morphometric and functional metrics are two independent measures, since structural connectivity is the basis of functional connectivity [58], we hypothesize that there exist non-linear relations between the functional and anatomical features that reveal differences between two distinct study groups. We propose a novel approach to combine these two modalities to indirectly build and identify the relationship as outlined in Fig. 5.2. Towards this, we first extract the morphometric measures (Fig. 5.2 Left). We then extract the FC network. The network topology is
better understood by analyzing graph metrics. Thus, from the FC, we compute the following graph metrics:

- **Node strength (NS)** - computed as the sum of the weights of edges to the node.
- **Node Degree (ND)** - computed as the number of edges to the node.

Towards this, we generate sparsified connectivity matrices by thresholding the edge strength of the FC between $[0.4, 0.8]$ at steps of 0.1. This sparsification approach leads to the generation of several undirected connectivity matrices. We then use the NDs from these and the NS from the original FC as the functional measures in place of the pearson’s correlation (Fig 5.2 Right). Finally, we combine the morphometric and functional measures into a single node feature vector and z-score normalize as in Eq 5.1. The correlation between the different nodes is computed using this feature vector to build the new connectivity matrix, the Functional Morphometric Similarity (FMS) network.

### 5.3 Experiments and Results

All experiments were performed using Freesurfer [103], AFNI [106] and Python [107] on a linux base with 2.2GHz CPU and 16Gb RAM. BrainNet Viewer [108] is used to visualize the connectivity networks.

#### 5.3.1 Dataset

After preprocessing the data, subjects with number of censored volumes greater than 10% the total number of volumes (post-stabilization) in rs-fMRI were discarded. This leaves us with $N$
Figure 5.2: Generating the Functional Morphometric Similarity (FMS) network from the anatomical and function images. Left: Morphometric measures are extracted from the cortical surface of the T1 image. Right: Functional connectivity is generated from the surface of the fMRI and node degrees (ND) are extracted via sparsification (thresholding the edge strength). The morphometric and functional measures are combined to generate the FMS. NS - Node Strength, @ - thresholded at.

Subjects. To ensure that equal number of volumes/time-points were considered for the construction of the functional connectivity network, equal number of time-points, 90% of all volumes were considered. The subject demographics can be found in Table 5.1.
<table>
<thead>
<tr>
<th>Group</th>
<th>Age ($\mu \pm \sigma$)</th>
<th>Education ($\mu \pm \sigma$)</th>
<th>Male (M) / Female (F)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Chess Masters</td>
<td>28.67 ± 9.06</td>
<td>13.71 ± 6.42</td>
<td>16 / 8</td>
</tr>
</tbody>
</table>

Table 5.1: Subject demographics.

5.3.2 Statistical Significance of Morphometric Features

In [51], voxel based morphometry was used to identify statistically distinct regions in the chess masters compared to the novice players. Similarly, a two-sample t-test for each region and each metric was performed to identify whether these morphometric measures derived from functionally defined anatomical regions were distinct between chess masters and novice players. The significance value was set at $p < 0.05$ using a family-wise error (FWE) correction based on the Holm-Sidak method [109]. This resulted in only two significant regions and the metric of interest being Cortical Thickness. Table 5.2 shows the regions and the FWE corrected p-values. The identified regions were different from those identified in literature showing the benefit of using a functional parcellation.

<table>
<thead>
<tr>
<th>ROI</th>
<th>Metric</th>
<th>Hemisphere</th>
<th>$p_{value}^*$</th>
</tr>
</thead>
<tbody>
<tr>
<td>Somatomotor A</td>
<td>CT_sd</td>
<td>L</td>
<td>0.0443</td>
</tr>
<tr>
<td>Peripheral Visual</td>
<td>CT_avg</td>
<td>R</td>
<td>0.0137</td>
</tr>
</tbody>
</table>

Table 5.2: Anatomically significant functionally defined regions and the corresponding metric of difference. * - FWE corrected, L - Left, R - Right

5.3.3 Classification

With limited statistically different regions, we turn to the connectivity matrix to identify further differences between the two healthy groups of individuals. To ensure a strong classification model,
The $k$-fold cross-validation approach \cite{101} is used with $k = 10$. Cross-validation is an approach to do out-of-sample testing with limited data wherein the data is split into training and testing sets to enable model validation. In $k$-fold cross-validation, the data is split into $k$ different training and test sets and the average performance across all metrics is the overall model evaluation. Further, we use a stratified 10-fold cross-validation wherein, the data splitting attempts to ensure that there are similar number of samples of all classes in the test set.

The different connectivity matrices/connectomes generated were all symmetrical matrices i.e., they are identical across the diagonal. The diagonal itself represents a self-correlation and is always 1. Thus, only the upper-right triangle was extracted from these connectivity matrices and the resulting feature vector was used in the classification task. For an $m \times m$ matrix the feature vector dimension is computed as $\frac{m \times (m-1)}{2}$. Since the networks are $34 \times 34$, the extracted feature vector is of dimension 561. The feature dimension is $>$ the size of the training data and thus, feature selection was used to reduce the feature size. Towards this, a simple t-test was performed and only the significant features ($p < 0.05$) were chosen. The selected features were used to train a support vector machine (SVM) \cite{110} with a linear kernel, a regularization parameter $C = 1$ and scale parameter $\gamma = 1e-4$. The performance metrics used are:

- **Accuracy**: Is defined as the ratio of the total number of correctly classified items to the total number items. Computed as:

$$\text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN},$$

where TP - True positives, FP - False positives, TN - True negatives and FN - False negatives.

- **Precision**: Defined as the ratio of correctly predicted positive results to the total predicted
<table>
<thead>
<tr>
<th>Approach</th>
<th>Accuracy (%)</th>
<th>Precision (%)</th>
<th>F1-score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>w/o t-test</td>
<td>64.83</td>
<td>62.67</td>
<td>64.83</td>
</tr>
<tr>
<td>w t-test</td>
<td>76.33</td>
<td>76.83</td>
<td>74.05</td>
</tr>
</tbody>
</table>

Table 5.3: 10-fold cross-validation accuracy using functional connectivity features. w/o - without, w - with

positive results. It is computed as:

\[
Precision = \frac{TP}{TP + FP}.
\]

• F1-Score: Defined as the harmonic mean of precision and recall, is computed as:

\[
F1 - Score = 2 \cdot \frac{precision \times recall}{precision + recall}.
\]

5.3.4 FC based classification

We start with the FC based classification. To test the effect of the feature selection step, 10-fold cross-validation was first performed using all the features extracted from the FC. Using the 561 features, the SVM was trained and the classification performance can be seen in Table 5.3. Then, feature reduction/selection via t-test was performed prior to training the SVM in each fold. This resulted in an average accuracy of \( \approx 76\% \), an improvement of over 11%. The common significant connections across all 10 folds of the cross-validation are found and Fig. 5.3 shows these connections. Based on the average p-value obtained, the top-10 functional connections are shown in Table yy.
5.3.5 **MSN based classification**

Having established the benefit of the feature selection towards the classification task and our baseline classification accuracy, we perform the connectivity feature based classification using features from the MSN networks. Since we consider the construction of MSNs using 3 or more measures, the total number of MSNs is computed as:

\[ MSNs = \sum_{k=3}^{8} \binom{n}{k}, \]

where 8 is the total number of morphometric measures. There are a total of xx MSNs and thus, xx trained SVM models. Out of these xx models, the best 10-fold accuracy was achieved using SA, CT_sd and CI as the morphometric features of interest. The MSN representing the connections
for this model is shown in Fig. 5.4. The performance measures of the top performing models are shown in Table 5.4.

![Figure 5.4: Top 10 morphometric connections differentiating Chess Masters from Novice Players. Axial view of the anatomical connections with nodes representing one of the 17 networks in the Yeo atlas.](image)

<table>
<thead>
<tr>
<th>Approach</th>
<th>Model</th>
<th>Accuracy (%)</th>
<th>Precision (%)</th>
<th>F1-Score (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>MSN</td>
<td>SA+CT_sd+CI</td>
<td>73.83</td>
<td>77.5</td>
<td>70.92</td>
</tr>
<tr>
<td>MSN</td>
<td>SA+CT_sd+FI+CI</td>
<td>71.83</td>
<td>73.5</td>
<td>69.33</td>
</tr>
<tr>
<td>FC</td>
<td>FC</td>
<td>76.33</td>
<td>76.83</td>
<td>74.05</td>
</tr>
<tr>
<td>Pseudo-FMS</td>
<td>Majority Voting</td>
<td>80.17</td>
<td>84.67</td>
<td>79.33</td>
</tr>
<tr>
<td>FMS</td>
<td>CT_sd+GC+FI+CI+ND@0.6+ND@0.8</td>
<td>88.00</td>
<td>91.58</td>
<td>87.52</td>
</tr>
<tr>
<td>FMS</td>
<td>CT_avg+GC+FI+CI+ND@0.4+ND@0.8</td>
<td>87.17</td>
<td>91.17</td>
<td>86.57</td>
</tr>
</tbody>
</table>

Table 5.4: Classification results of morphometric similarity networks (MSN), functional connectivity networks (FC) and functional morphometric networks (FMS) on 10-fold cross-validation. SA - Surface area, CT_avg - Average cortical thickness, CT_sd - Cortical thickness standard error, FI - Folding index, CI - Curvature Index, GC - Gaussian curvature, ND@ - Node degree at threshold level,
5.3.6 FMS based classification

We first propose a pseudo-FMS on the basis of a simple majority voting strategy, combining the best performing MSN-based classification models and FC-based classification. The common connections across the majority voting model is shown in Fig. 5.5. The salient ventral attention network, dorsal attention network and central visual networks are common across the models. The resulting model performance has an accuracy of 80%, a 6% and 3% improvement over the best performing MSN and FC models respectively. This indicates possibly complimentary information in the MSN to the FC. We thus, combines the MSN and FC metrics to generate the FMS. The FC was thresholded at edge strengths of \([0.4, 0.5, 0.6, 0.7, 0.8]\) and NDs were computed for each of these sparse undirected networks. NS was computed directly from the FC. The total number of features was now 14 (8 morphometric measures and 6 functional measures). Similar to MSN-based classification, we tested all possible combinations and this resulted in \(14C_3\) models. Table 5.4 shows the best performing FMS-based classification models. Highest accuracy of 88% was achieved and Fig. 5.6 shows the common significant connections.

Across all three approaches - FC, MSN and FSM, among the top-10 connections identified was the connection between a part of the saliency and ventral attention network and a part of the control network. The saliency and ventral attention network is known to be the hub between the default mode network and the control network (central execution network). This could mean that the skill specialization could have influenced how the brain switches between the default mode and the cognitive/execution mode. The default mode network was also identified in pairs of connections significantly different between the groups of chess masters and novice players. From the literature on chess player brain image analysis, we know that the default mode network plays an important role [51, 50].
5.4 Summary

In this work we have proposed functional morphometric similarity (FMS) networks, an approach to combine anatomical and functional metrics, to classify chess masters from novice players. Our trained SVM model achieved a classification accuracy of 88% showing drastic improvements over the standard functional connectivity based approach. Additionally, we have used a functional parcellation that combines functionally coupled regions into the same networks. This allowed us to study whether broadly defined functional networks show morphometric differences. We thus identified two such coupled regions with statistically significant differences in cortical thickness.

Figure 5.5: Common Functional and Morphometric connections differentiating Chess Masters from Novice Players. Connections are derived from common connections from the morphometric similarity networks and functional connectivity network.
Figure 5.6: Top 10 Functional Anatomical connections differentiating Chess Masters from Novice Players.
CHAPTER 6: IMPROVING LATENT REPRESENTATIONS VIA
VARIATIONAL CAPSULE ENCODERS

Related Publication:

6.1 Overview

In brain image analysis, and most medical imaging tasks, identifying a strong signal representation is integral to enhancing the algorithm’s performance. So far, we have seen signal processing approaches in both time- and frequency- domain being performed to accomplish this. Learning a strong signal representation with minimal supervision, would greatly benefit medical image analysis. In this study, for a better signal representation, an understanding of the part-whole relationship for objects could be significant, which traditional encoder-decoder architectures fail to achieve. We propose to combine capsule networks and VAEs in a novel capsule network based variational encoder architecture, called Bayesian capsules (B-Caps), to influence the mean and standard deviation of the sampling distribution in the latent space. We show the ability of B-Caps to learn a better representation than the traditional VAE using MNIST and FASHION-MNIST datasets and lay the foundation for further exploration in more complex datasets.
6.2 Methods

Herein, we briefly outline the VAE and capsule network layers before presenting our proposed fusion of these two concepts.

6.2.1 Variational Autoencoder

Unlike the vanilla AE, the VAE generates two outputs in the encoder, a vector of means and a vector of standard deviations. They form the parameters of a vector of random variables from which samples are generated. This helps the encoder learn a potentially different mean for each class while the standard deviation controls its spread and reduces the overlap with other classes.

VAEs are typically trained using two losses - a generative loss, which measures the accuracy of the reconstructed image, and a latent loss which measures how closely the latent variables are distributed to a unit Gaussian. The latent distribution loss is controlled using the Kullback-Leibler (KL) divergence:

\[
D_{KL}(P||Q) = - \sum_{x \in \chi} P(x) \log \frac{Q(x)}{P(x)},
\]

(6.1)

where \( \chi \) = probability space,

\( P & Q \) = known distributions.

In the VAE, \( P \) is the latent variable and \( Q \) is a unit Gaussian. To enable random sampling and support backpropagation for optimizing the KL loss, a reparameterization trick is employed. The variable \( z \), using the samples from the encoder outputs - including mean (\( \mu \)) and standard deviation (\( \sigma \)), can be sampled from the standard deviations such that the mean is added afterwards:

\[
z = \mu + \sigma \cdot \epsilon,
\]

(6.2)
where $\epsilon = \mathcal{N}(0, 1)$. A more comprehensive summary of the VAEs can be found in [111].

6.2.2 Capsule network layers

Capsules are a group of neurons that represent an object or object part in an image. Unlike a fully connected layer, every capsule layer is either 2-dimensional or in the case of convolutional capsules, $N + 1$ dimensional, where $N$ is the convolution dimension [112]. The additional dimension converts the scalar filter into a vector representation, thereby enabling the encoding of pose and orientation information. The vector length along this dimension gives the probability (that an object/object-part exists) and the orientation with respect to its parent layer (i.e. input image or previous capsule layer). Dynamic routing is utilized to enable grouping of capsules such that similar lower level capsules are grouped together to a higher level capsule. Considering a lower level child capsule ($c_i$) and a higher level parent capsule ($c_j$), the description vector $u_i$ of $c_i$ is related to $c_j$ via transformation matrix weights $W_{ij}$, which are trained via backpropagation. Hence, the child capsule predicts the output of the parent capsule as:

$$\hat{u}_{j|i} = W_{ij}u_i.$$  

The output $u_j$ of $c_j$ is computed as,

$$u_j = \sum k_{ij}\hat{u}_{j|i},$$

where $k_{ij}$ is a coupling coefficient. The output description $u_j$ is normalized to $[0, 1]$ via a squashing function [30]. In this way, the description vector can be seen as the probability of detecting a feature with a given orientation. This builds the basic object (input capsule) and object-parts (child capsules/group of capsules) relationship. Different routing algorithms such as expectation maximization (EM) routing [113], self routing [114], and dynamic routing with min-max normal-
ization [115] have been proposed to reduce the computational burden and improve the coupling between capsules. Exploring different routing algorithms are kept outside the scope of current work.

![B-Caps encoder architecture](image1)

![Decoder architecture](image2)

Figure 6.1: B-Caps architecture: Represented by three different parameters; the number of capsules ($C$), the size of description vector ($D$), and the latent dimension ($L$).

### 6.2.3 Variational Capsule Encoder

In this study, we investigate the ability of capsules for learning feature variations in latent space. For this, we aim to build strong relationships between the image and the object parts even in a shallow VAE network where the encoder part includes capsule layers. Traditionally in capsule networks, a convolutional layer is employed to generate features which are then converted into a primary capsule layer. In this way, the channels serve as the description vector of the single capsule. Herein, we propose to skip this step for small images like those in the MNIST and Fashion-MNIST datasets. Instead, we treat the flattened image as a description, thereby converting the whole image into a capsule. We posit that individual pixels are descriptive of the image and thus can be treated...
as a vector description for a single image capsule.

The variational capsule encoder architecture for our proposed B-Caps is shown in Figure 6.1(a). The output of our proposed capsule encoder consist of two groups (mean and standard deviation) of capsule layers, each with \( L \) capsules. The length (\( L \)) of these layers defines a vector of random variables of length \( L \) in the latent space. The network architecture is represented by three different parameters: the number of capsules (\( C \)), the size of description vector (\( D \)), and the latent dimension (\( L \)). The flattened input image is routed to \( C \) capsules of vector length \( D \). These are then routed to the latent layer having \( L \) capsules, whose vector norm forms the latent vector.

Traditionally, in capsule based networks designed for classification or segmentation task [69] [112], an additional reconstruction loss can be included to encourage the capsules to encode the inputs’ instantiation parameters (such as the pose information). To take advantage of this, a fully connected decoder network (illustrated in Figure 6.1(b)) is used in our implementation to reconstruct the flattened image.

The B-Caps network is trained to minimize the mean squared error (MSE) loss and the Kullback-Leibler divergence loss, thereby simultaneously optimizing the latent space as well as the image reconstruction. To enable sampling and use of KL in backpropagation, a resampling trick is used to approximate the random normal posterior. The input to the latent sampling layer is a vector Euclidean norm of the mean and standard deviation capsules. The norm of the description vector \( u \) is computed as:

\[
    u_i = \sqrt{u_{i1}^2 + u_{i2}^2 + \cdots + u_{i64}^2},
\]

where \( i = 1 \cdots L \). The latent vector \( z \) is sampled as in Eq. 6.2 and the KL loss is computed as:

\[
    D_{KL}(\mathcal{N}(\mu, \sigma)||\mathcal{N}(0, 1)) = 0.5 \times \sum_{L}(exp(\sigma) + \mu^2 - 1 - \sigma). \tag{6.3}
\]
6.3 Experiments

6.3.1 VAE architecture

We devised the basic VAE architecture with two fully connected layers in the encoder and two fully connected layers in the decoder. The output of the encoder were the mean ($\mu$) and standard deviation ($\sigma$) that are fully connected layers with $L$ units, where $L$ is the latent vector dimension. The decoder network used (Figure 6.1(b)) takes as input the sampled latent vector $L$ and outputs the reconstructed image. In our experiments, we fix the dimensions of the fully connected layer in the encoder and decoder as 512. Batch normalization was included after the first fully connected layer, in both the encoder and decoder.

![Reconstruction results with different training strategies](image)

Figure 6.2: Reconstruction results with different training strategies

6.3.2 B-Caps Architecture

Our B-Caps architecture follows the VAE formulation and has a depth of just 2 layers. Each of these capsule layers is a fully connected capsule, also known as DigitCaps [30]. The primary capsule layer comprises of $C$ capsule types of vector length $D$. The last layer comprises of the two
outputs, the mean ($\mu_c$) and the standard deviation ($\sigma_c$) capsules, having $L$ capsule types of description length $D_1$. The complete variational capsule encoder is represented as $\{\{C, D\}, \{L, D_1\}\}$. In our experiments we fixed $D_1 = 64$, and varied the number of capsule types in the layer prior to the output layer in the encoder part by varying $C$, hence evaluated the effect of the description length $D$. Since capsules encode part-whole relationships, the effect of increasing $C$ can change with varying $L$. Therefore, we set $L = 2$ for this set of experiments which also allowed visualizing class separation in the latent space.

### 6.3.3 Effect of latent dimension on performance

By increasing the length of the latent attribute vector, the quality of the generated/reconstructed images can improve [28]. In our proposed B-Caps, the length of the latent attribute vector $L$ translates to the number of capsule types. By increasing this value, we tested the number of part-whole relations that can be learnt in a shallow network. This also enabled us to test whether learning more part-whole relationships improves the performance.

### 6.4 Results

#### 6.4.1 Summary of Main Results

We trained the proposed B-Caps on MNIST and Fashion-MNIST data and observed that the model does not converge when the variance was sampled from a standard normal distribution. We hypothesized that the feature distribution in capsule layers does not follow simple Gaussian distribution; therefore, we trained B-Caps using a data driven approach to observe the distribution of data along capsule layers. We call this approach psuedo-MCMC due to its similarities with the Markov Chain Monte Carlo (MCMC) approach. We also showed that with increasing dimension of the latent
variables, the proposed B-Caps outperformed the basic VAE, indicating that the learned attributes have a stronger relation to their preceding layers in B-Caps. In the following, we present our experimental results in detail.

6.4.2 Training B-Caps

We first trained our proposed B-Caps in a similar way as used for training regular VAEs using the standard normal distribution with the reparameterization trick. We noticed that backpropagation failed and the loss function was unable to converge. A possible reason for this is that the mean and standard deviation vectors in B-Caps are driven by the length of the vectors $\mu_c$ and $\sigma_c$ respectively, which are always non-negative. Based on this observation, to better initialize the latent space sampling, we replaced the standard normal distribution with a normal distribution $\mathcal{N}(\mu = 0.5, \sigma = 0.5)$. As seen in Figure 6.2, the trained model converged albeit to a poor reconstruction. The random normal sampling comes marginally closer to approximating the true distribution of the data. This also indicates that the variance cannot be sampled directly from a standard normal distribution. Based on this, we modified the random normal distribution to be data-driven by allowing the distribution to be modulated by $\mu_c$ and $\sigma_c$. Although this modulation may break the backpropagation with the absence of the independent random sampling (which the reparameterization trick entails), we can alleviate the potential of exploding gradients by using batch normalization along the description vector dimension in the capsule layers. It’s because batch normalization adds a form of regularization to the network and helps accelerate the training [116]. However, it should also be noted that since the description length varies from [0,1], there is a very low chance of a “bad” variance in sampling that could break the backpropagation.
Figure 6.3: The distribution of digits in the latent space for basic VAE and B-Caps (C=16, D=64) on MNIST dataset. Although none of these distributions are optimal, B-Caps latent space has visibly better allocations of certain classes.

In this data-driven approach, the samples are being drawn as,

$$z = \mu_c + \sigma_c \cdot \epsilon,$$  \hspace{1cm} (6.4)

where $\epsilon = \mathcal{N}(\mu_c, \sigma_c)$. This strategy translates to creating samples similar to each data point and repeating the process several times, similar to the MCMC process. Hence, with the $\mu_c$ and $\sigma_c$ changing with every update, the latent space is learnt in an MCMC manner with the reconstruction loss being learnt via optimization. As mentioned earlier, we call this training as psuedo-MCMC. Our experimental results have shown that the presented data-driven approach converges with a lower total network loss compared to the random normal distribution.
6.4.3 MNIST Reconstruction

The MNIST dataset consists of images representing hand-written digits (0 – 9) with dimension 28 × 28 [117]. The data comprises of 60,000 training and 10,000 test images. We normalized all images within the [0, 1] range and flattened them before feeding into the B-Caps network.

In all our experiments, the networks were trained with a batch size of 128 for 100 epochs. Adam optimizer was used with an initial learning rate of 1e−3 [118]. During the testing phase, image reconstruction quality was evaluated using the mean squared error (MSE) and the structural similarity index metric (SSIM). MSE is the mean difference between the squared pixel-wise errors between the original image and the model estimate (reconstruction). It is computed as:

\[
MSE(A, B) = \frac{1}{N} \sum_{i=1}^{N} (A_i - B_i)^2,
\]

where \( A, B = \) actual image, reconstructed image, 
\( N = \) number of pixels.

SSIM, on the other hand, is a perceptual image quality metric assessing the effect of luminance, contrast, and image structure. SSIM is computed as a product of the aforementioned variables as follows:

\[
SSIM(A, B) = \frac{(2\mu_A \mu_B + C_1)(2\sigma_{AB} + C_2)}{(\mu_A^2 + \mu_B^2 + C_1)(\sigma_A^2 + \sigma_B^2 + C_2)},
\]

where \( A, B = \) actual image, reconstructed image, 
\( \mu_A, \mu_B = \) means of A & B, 
\( \sigma_A, \sigma_B = \) standard deviations of A & B, 
\( \sigma_{AB} = \) cross-variance of A & B, 
\( C_1, C_2 = \) constants to avoid instability.
<table>
<thead>
<tr>
<th>Model</th>
<th>Capsule types (C)</th>
<th>Description length (D)</th>
<th>SSIM mean±std</th>
<th>MSE mean±std</th>
</tr>
</thead>
<tbody>
<tr>
<td>Basic VAE</td>
<td>—</td>
<td>—</td>
<td>0.555 ± 0.154</td>
<td>0.041 ± 0.019</td>
</tr>
<tr>
<td>B-Caps</td>
<td>8</td>
<td>64</td>
<td>0.541 ± 0.144</td>
<td>0.043 ± 0.020</td>
</tr>
<tr>
<td>B-Caps</td>
<td>16</td>
<td>64</td>
<td><strong>0.580 ± 0.133</strong></td>
<td><strong>0.040 ± 0.017</strong></td>
</tr>
<tr>
<td>B-Caps</td>
<td>32</td>
<td>64</td>
<td>0.573 ± 0.147</td>
<td>0.041 ± 0.019</td>
</tr>
<tr>
<td>B-Caps</td>
<td>8</td>
<td>128</td>
<td>0.529 ± 0.152</td>
<td>0.046 ± 0.020</td>
</tr>
<tr>
<td>B-Caps</td>
<td>16</td>
<td>128</td>
<td>0.577 ± 0.129</td>
<td>0.040 ± 0.018</td>
</tr>
</tbody>
</table>

Table 6.1: Comparison of reconstruction quality on MNIST while varying the capsule types (C) and description length (D). std - standard deviation.

We compared the basic VAE at a latent dimension of $L = 2$ against various configurations of the B-Caps architecture. For B-Caps, we first fixed the value of description length (D) and varied capsule types (C). Later, we increased the value of D and repeated the experiments by varying C. We observed that too many capsule types in the intermediate layers and a larger description length (D) do not help in improving the reconstruction (see Table 6.1). Compared to the basic VAE, at latent dimension $L = 2$, B-Caps \{16, 64\}, \{2, 64\} performs better. We visualized the latent space in Figure 6.3: as illustrated, while the basic VAE gets close to the standard normal distribution with different classes radiating outwards in the 2D space, the latent space for B-Caps shows a clear separation of most classes.

As mentioned earlier, increasing the number of latent variables improves performance when using the basic VAE architecture. We tested how this fact can be translated into B-Caps. First, we chose B-Caps \{8, 64\}, \{L, 64\} as our base model and varied $L$ from 2 $\rightarrow$ 10. The effect of varying latent dimension is seen in Figure 6.4, where we observe a poor initial guess, which started to improve at higher dimensions with performance superior to that of VAE. We also compared the effect of latent dimension in image reconstruction quality on our proposed B-Caps and baseline VAE (Figure ??). We observed that the B-Caps network outperformed the baseline with increasing
Figure 6.4: Improvement in reconstruction of MNIST digits as a function of latent variable length. 

$L$ and that the improvement was more discernible beyond a latent dimension of $L = 4$. This adds credence to our hypothesis that the latent representation (in higher dimensions) of B-Caps is more powerful than conventional VAEs.

6.4.4 Fashion-MNIST Reconstruction

Fashion-MNIST dataset is similar to MNIST, having images of dimension $28 \times 28$ split into 10 classes related to fashion products [119]. There are 60,000 training images and 10,000 test images. Similar to our experiments for MNIST, we normalized the data to the range of $[0, 1]$ and flattened them to the length of 784 prior to our experiments.

Similar to the MNIST reconstruction experiments, we also tested different B-Caps architectures as well as the basic VAE at a latent dimension 2. In Fashion-MNIST experiments the basic VAE performed marginally better than the B-Caps architectures at lower latent dimension (see Table 6.2), indicating that the number of capsule types is more important than the description length. But the performance improves by increasing the latent dimension. We hypothesize that for a more
complex dataset (Fashion-MNIST), using capsules in the encoder is more significant. Using higher
dimension could reflect learning better part-whole relationships, and hence better performance.

<table>
<thead>
<tr>
<th>Model</th>
<th>Capsule types (C)</th>
<th>Description length (D)</th>
<th>SSIM mean ± std</th>
<th>MSE mean ± std</th>
</tr>
</thead>
<tbody>
<tr>
<td>basicVAE</td>
<td></td>
<td></td>
<td>0.577 ± 0.161</td>
<td>0.029 ± 0.016</td>
</tr>
<tr>
<td>B-caps</td>
<td>8</td>
<td>64</td>
<td>0.568 ± 0.149</td>
<td>0.032 ± 0.024</td>
</tr>
<tr>
<td>B-caps</td>
<td>16</td>
<td>64</td>
<td>0.568 ± 0.142</td>
<td>0.032 ± 0.022</td>
</tr>
<tr>
<td>B-caps</td>
<td>32</td>
<td>64</td>
<td>0.564 ± 0.157</td>
<td>0.032 ± 0.020</td>
</tr>
<tr>
<td>B-caps</td>
<td>8</td>
<td>128</td>
<td>0.547 ± 0.155</td>
<td>0.034 ± 0.020</td>
</tr>
<tr>
<td>B-caps</td>
<td>16</td>
<td>128</td>
<td>0.570 ± 0.142</td>
<td>0.032 ± 0.026</td>
</tr>
</tbody>
</table>

Table 6.2: Comparison of reconstruction quality on FASHION-MNIST while varying the capsule
types (C) and description length (D).

Among the B-Caps models, we noted that \{8, 64, \{L, 64\}\} performs the best but all other models
had almost similar performance in terms of SSIM and MSE. The \{8, 128, \{L, 64\}\} model per-
formed poorly, indicating that the number of capsule types is more important than the description
length. The latent space for the B-Caps model \{8, 64, \{L, 64\}\} along with the baseline VAE is
shown in Figure 6.5. We can see that the baseline VAE is separating the classes with sample points
in the latent space radiating outwards, while B-Caps, which shows similar class-separation, does
not center the distribution.

We also evaluated the performance of our proposed models with varying dimension of the latent
space. We obtained an improvement in the performance with an increase in latent space dimension.
We compared the reconstruction of the model \{8, 64, \{2, 64\}\} against the basic VAE for latent
dimension \(L = 10\). Figure 6.6 shows this comparison for two different clothing categories. While
both approaches capture the overall shape of the objects, B-Caps appears to capture more of the
texture information within these images.

Figure 6.7 highlights this improvement of B-Caps with increasing latent dimension. B-Caps im-
proves from an incorrect reconstruction to a generic class representation (i.e., rectangular bag) at
a latent dimension of \(L = 5\) (Figure 6.7). Further increasing the latent dimension allowed B-Caps
Figure 6.5: The distribution of the classes in latent space for basic VAE and B-Caps (C=8, D=64) on Fashion-MNIST dataset.

Figure 6.6: A comparison between reconstruction of Fashion-MNIST images using basic VAE and B-Caps. Left to Right: Original image, basic VAE reconstruction and B-Caps reconstruction, $L = 10$
to capture more variations and reconstruct the trapezoidal shape of the bag. Unlike B-Caps, basic VAE does not improve beyond \( L = 5 \).

Figure 6.7: Improvement of reconstruction as a function of latent variable length.

For Fashion-MNIST, in our qualitative evaluations, B-Caps showed better reconstruction quality when the latent variables dimension is high and this trend is similar to what we obtained for MNIST.

Figure 6.8: Variation in image reconstruction quality measured using SSIM for different latent variable dimensions.
We evaluated the performance of B-Caps in a classification setting using the reconstructed images. F1-score, the measure of a test’s accuracy was used to assess the classification performance. F1-score is computed as:

\[
F1 = 2 \cdot \frac{\text{precision} \times \text{recall}}{\text{precision} + \text{recall}}.
\]

We trained a support vector machines (SVM) based classifier on the MNIST and Fashion-MNIST datasets. A grid search was performed to identify the best settings for the SVM for each of these datasets. For both datasets, the radial basis function (RBF) based kernel was used with the kernel
coefficient of $\gamma = 0.01$. The regularization parameter $K$ was set to 100 and 10 for MNIST and Fashion-MNIST, respectively. Once trained, the reconstructed images were evaluated in a classification setting. The proposed B-Caps model $\{\{8, 64\},\{L, 64\}\}$ and basic VAE, with $L$ from 2 → 10 were evaluated comparatively. Figure 6.9 shows the variation of F1-scores with respect to the latent dimension. For both MNIST and Fashion-MNIST classification, B-Caps outperformed the baseline VAEs for $L \geq 3$. Further, we can see that in Fashion-MNIST, the performance improvement is greater, indicating that B-Caps was able to capture more variations than VAEs.

One needs to note that, an increase in the length of the latent dimension resulted in an increase in parameters for B-Caps encoder by $\approx 60,000$ (per latent variable). Whereas, for basic VAE the parameters increase by $\approx 1,000$ per latent variable. To account for the difference in parameter space, we replaced the intermediate layer in the basic VAE with a 1024 dimension fully connected layer. The resulting comparison of the trainable encoder parameters is shown in Table 6.3. With a more comparable number of trainable network parameters, we repeated the comparison of the classification performance to check whether this increase in parameters for the basic VAE would help change the plateauing effect. Based on the results from this new experimental setting, we conclude the overall performance was not significantly effected by increasing parameters for the VAE model (see Figure 6.10). In VAE, the performance started plateauing beyond $L = 5$ similar to Figure 6.9. These results revealed that the total number of parameters was not the reason for B-Caps performing better than basic VAE. Instead, it is because B-Caps learns richer attributes (in the latent space) owing to capsule layers used in the encoding process. The data driven sampling from the latent space also augmented this learning process with a better inference during the reconstruction process.

Additionally, we tested the effect of batch normalization (BN) on training the B-Caps and basic VAE architectures. BN was applied after the first capsule layer in the B-Caps and after the first fully connected layer in the basic VAE. BN was also used in the decoders for both networks. The
Table 6.3: Trainable encoder parameters in the basic VAE with intermediate layers of 512 and 1024 and B-Caps. FC- fully connected layer.

<table>
<thead>
<tr>
<th>Latent Dimension</th>
<th>basic VAE FC-512</th>
<th>basic VAE FC-1024</th>
<th>B-Caps C=8, D=64</th>
</tr>
</thead>
<tbody>
<tr>
<td>2</td>
<td>405K</td>
<td>810K</td>
<td>532K</td>
</tr>
<tr>
<td>4</td>
<td>407K</td>
<td>814K</td>
<td>663K</td>
</tr>
<tr>
<td>6</td>
<td>409K</td>
<td>818K</td>
<td>794K</td>
</tr>
<tr>
<td>8</td>
<td>411K</td>
<td>822K</td>
<td>925K</td>
</tr>
<tr>
<td>10</td>
<td>413K</td>
<td>826K</td>
<td>1.05M</td>
</tr>
</tbody>
</table>

Figure 6.10: Variation of classification performance with latent variable dimensions where basic-VAE has intermediate layer of dimension 1024.

comparison is shown in Figure 6.11. We observe that for B-Caps both the SSIM (reconstruction quality) and the F1 score (classification performance) improves when batch normalization is used. Whereas, for basic VAE the performance is almost similar in terms of F1 score both with and without BN. The SSIM decreases when BN was used with basic VAE. We can conclude that BN helped in improving the performance of our proposed B-caps architecture. In basic VAE, the shallowness of the network potentially negated any benefit of the BN.
6.5 Summary

We have presented a novel capsule based variational autoencoder architecture, called B-Caps, for an effective representation learning in the latent space and compared its performance in the image reconstruction and classification tasks. We showed that the proposed B-Caps outperformed the baseline VAE with increasing dimension of the latent space. It is interesting to note that the systematic selection of latent dimension is also supported by the winning lottery ticket theory, a recent study where it is shown that with the right initialization a highly pruned network could perform similar to a very dense network [120]. Our proposed B-Caps architecture, although not learning a sparse network, is aimed towards learning a powerful representation in the latent space. This learning is conditioned in a manner to account for the part-whole relationship (using Capsules) and embed Bayesian inference (data driven sampling from the latent space).

Improving latent space representation is not necessarily a complementary task to classification as
it could potentially require different information for reconstruction than for classification. This can lead to lower classification accuracy than existing approaches. However, the benefit of learning a strong latent space lies in the ability to learn distinct attributes that can be manipulated to further analyze the effect.

The reparameterization trick using an independent normal distribution enabled backpropagation and attempts to address the problem of a “bad” variance estimate. However, in capsules, the variance is always between [0,1] as the length of the vector represents the probability of coupling between capsules. This helps handle the “bad” variance estimate problem but suffers from an initialization problem. The data-driven modulation of the latent space helped in handling this issue and stabilized the training. The shallowness of the proposed networks also enabled the data-driven approach. We hypothesize that in deeper networks, this data-driven approach will not be needed and a standard reparameterization trick can be used.
CHAPTER 7: CONCLUSION AND FUTURE DIRECTIONS

This dissertation focuses on improving the understanding of the human brain thereby improving the pre-surgical planning. Towards this, it proposes algorithms for handling noisy brain data, ECoG channel classification for eloquent language cortex localization and identifying functional and anatomical differences in the healthy brain.

7.1 Eloquent Cortex Localization using Electrocorticography

In the first part of our analysis, we proposed a frequency domain based machine learning algorithm for channel response classification. We first extracted the power spectral density of the time-series from each block in the experimental paradigm. We then used the unrestricted frequency spectrum, i.e., the full signal spectrum encompassing $[0 - 600]Hz$. Log normalized full spectrum features were used to train a RF classifier in a supervised learning framework where the ground truth labels were obtained from the gold-standard ESM. The final channel label was determined based on a majority voting among the different block labels. The proposed approach was evaluated on an ECoG dataset of 6 epileptic subjects.

In the next phase on the analysis, we designed deep learning algorithms to improve the channel classification accuracy. We analyzed the importance of time domain features that are popular in EEG tasks in ECoG application. We carefully extracted 7 different time domain features and evaluated the contribution of each of these features and identified the most promising feature among them all. We then designed a deep learning module to extract useful information from the time-domain features and performed domain fusion by combining these with deep features from the frequency domain. We performed data augmentation using sliding window approach to increase
the available data samples for training the deep learning network in an end-to-end manner. The final classification label was assigned based on majority agreement among the multiple windows of data in each channel. As accurate channel classification can determine the quality of life of a patient post-surgery, the automatic classification of channels using a safe method, such as ECoG, is a very important challenge.

7.2 Smart Thinkers

With regards to identifying functional and anatomical differences in the brains of different groups of healthy subjects based on skill specialization, we proposed functional morphometric similarity (FMS) networks, an approach to combine anatomical and functional metrics, to classify chess masters from novice players. We extracted morphometric measures from the surface of the T1-weighted images. We then computed the FC from the surface of the fMRI images. We extracted graph metrics from the FC that identify the importance of nodes and combined these with the morphometric features. The correlation of the fused features between different regions on the brain surface was computed to generate the FMS. We then trained an SVM model on the FMS features to classify subjects as chess masters or novice players. We also analyzed the regions that contribute to this difference and their relationship to the game of chess.

7.3 Improving Latent Representations via Variational Capsule Encoders

To address the challenge of learning strong signal representations with minimal supervision, we designed a novel capsule based variational autoencoder architecture. We first reshaped the input image to an autoencoder into a 1D input. This input was then fed to a capsule layer by treating the image features as its description/pose information. We used dynamic routing to train the capsule
layers. The length of the capsule layers is then used to improve sampling in the latent space in a data-driven manner which helped overcome initialization issues faced in traditional capsule networks. A fully connected decoder was used to reconstruct the input image. The network was trained in an end-to-end manner. We compared the reconstructed image quality as well as the classification of the reconstructed images against the output of a traditional VAE and established the ability of the proposed BCaps to learn better signal representations.

7.4 Future Directions

Despite the state-of-the-art results in ECoG-FM predictions, there are some limitations of our work to be noted. First, our experimental paradigm (Figure 3.2) involves five different stories being played to the subject. The responses to these stories, have some inherent similarities, but overall are different. Therefore, training the deep learning model with a single label for the whole channel could add noise to the model. Though we have tested the effect of training the network, while treating each story individually, this reduces the overall data available to train the model. We believe that these results can be improved by including additional data and then training the system individually for each story in the paradigm. Second, the subjects used in this initial validation study were a mix of teenagers and adults. [121] found that the effect of epilepsy and seizures on children and adults was different i.e., the rules learned about the behavior of the brain in adults is different for children. Hence, a more comprehensive study with focus on children/teenagers is needed. This is one of our future aims to test the proposed machine learning based approach in different patient populations; however, patient recruitment is difficult due to the involvement of surgery, and disease prevalence. Finally, in the proposed approach, due to the exploratory and research nature of our study, the classification was not performed in real-time. We have used retrospective data for validating the innovations and are currently working on the real-time clinical implementation of the
algorithm. We intend to extend this study for mapping functional language cortex in prospective subjects. We believe that implementing such a reliable technology will increase current presurgical and intra-operative functional mapping accuracy, expand surgical treatment opportunities, prevent post-surgical language morbidity, and improve patient outcomes.

The identification of important regions that differ between two groups of healthy subjects can be extended to compare healthy subjects with subjects with neurological disorders for early disease diagnosis. The networks in the parcellation used were spatially distributed. Thus, the morphometric measurements are from non-connected anatomical regions. In future work, a further splitting of the network labels can be used so as to analyze the morphometric features of individual regions within the functional cluster. Moreover, with the increased dimensionality of the feature vectors, deep learning algorithms can be utilized to perform classification and identify important connections. While MSN enabled us to combine structural information, as a potential extension of this work, one could combine structural information from diffusion tensor imaging based measures such as fractional anisotropy, mean diffusivity, and structural node degree into the morphometric measures to further strengthen the functional-morphometric relationship.

There is a lot of potential for research in developing uncertainty driven approaches for medical imaging applications. The proposed proof-of-concept BCaps can be extended to more complex datasets involving multi-dimensional medical images. A convolutional capsule approach can be utilized to handle 2D, 3D and 4D images with a moderate increase in parameters. The BCaps can be improved by employing different routing strategies such as the recent capsule routing via variation bayes [71]. In future, this approach, can be used to generate a confidence value for our predictions, which is an extremely desirable attribute for clinical implementation. This can potentially change the face of computational brain imaging.
LIST OF REFERENCES


90


[99] T. Dietterich, “Overfitting and undercomputing in machine learning,” ACM computing sur-


[102] H. RaviPrakash, M. Korostenskaja, K. Lee, J. Baumgartner, E. Castillo, and U. Bagei, “Au-
tomatic response assessment in regions of language cortex in epilepsy patients using ecog-
held functional mapping and machine learning,” in Systems, Man, and Cybernetics (SMC),


for subdividing the human cerebral cortex on mri scans into gyral based regions of interest,”

[105] B. Thomas Yeo, F. M. Krienen, J. Sepulcre, M. R. Sabuncu, D. Lashkari, M. Hollinshead,
cerebral cortex estimated by intrinsic functional connectivity,” Journal of neurophysiology,


