TRANSFER LEARNING FOR BRAIN DECODING

A THESIS SUBMITTED TO
THE GRADUATE SCHOOL OF NATURAL AND APPLIED SCIENCES
OF
MIDDLE EAST TECHNICAL UNIVERSITY

BY

ERKIN ERYOL

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR
THE DEGREE OF DOCTOR OF PHILOSOPHY
IN
COMPUTER ENGINEERING

FEBRUARY 2023

Approval of the thesis:

TRANSFER LEARNING FOR BRAIN DECODING

submitted by **ERKIN ERYOL** in partial fulfillment of the requirements for the degree of **Doctor of Philosophy in Computer Engineering Department, Middle East Technical University** by,

Prof. Dr. Halil Kalıpçılar
Dean, Graduate School of **Natural and Applied Sciences** _____

Prof. Dr. Halit Oğuztüzün
Head of Department, **Computer Engineering** _____

Prof. Dr. Fatoş T. Yarman Vural
Supervisor, **Computer Engineering** _____

Examining Committee Members:

Assoc. Prof. Dr. Sinan Kalkan
Computer Engineering, METU _____

Prof. Dr. Fatoş T. Yarman Vural
Computer Engineering, METU _____

Assoc. Prof. Dr. Nazlı İkizler Cinbiş
Computer Engineering, Hacettepe University _____

Assist. Prof. Dr. Emre Akbaş
Computer Engineering, METU _____

Assoc. Prof. Dr. Tolga Çukur
Electrical and Electronics Engineering, Bilkent University _____

Date: 17.02.2023

I hereby declare that all information in this document has been obtained and presented in accordance with academic rules and ethical conduct. I also declare that, as required by these rules and conduct, I have fully cited and referenced all material and results that are not original to this work.

Name, Surname: Erkin Eryol

Signature :

ABSTRACT

TRANSFER LEARNING FOR BRAIN DECODING

Eryol, Erkin
Ph.D., Department of Computer Engineering
Supervisor: Prof. Dr. Fatoş T. Yarman Vural

February 2023, 120 pages

Understanding the human brain is a long-standing challenge in science. In this thesis, we focus on the brain decoding problem, where we estimate a cognitive state from functional magnetic resonance imaging (fMRI) images, to uncover the mechanisms in the brain-behavior relationship. However, due to the costly data acquisition process, fMRI studies are generally performed with a limited number of subjects in an experiment. Furthermore, the indirectly taken measurements introduce difficulties in the analysis of brain mechanisms.

With the increase in the available brain decoding datasets in recent years, transfer learning methods become applicable on brain decoding studies in neuroscience domain. In this thesis, we utilize the available data and knowledge in the neuroscience domain to improve the performance of a different but related brain decoding study, that we refer as transfer learning for brain decoding. We suggest two approaches on transfer learning for brain decoding.

In the first approach, we propose a novel Structured Multi-Layer Perceptron, utilizing a brain atlas. We observe that the Structured MLP model trained only on the target dataset has on-par classification and convergence time performance with the three

dimensional convolutional neural network model, that is pre-trained on a large source dataset.

In the second approach, we work on transfer learning between small-scale datasets that follows a common experimental paradigm. We propose Hierarchical Group PCA and its supervised variant for transferable feature generation that regards the session, subject and dataset relations. In the experiments, both methods outperform the state-of-the-art method, steadily on all transfer learning cases.

Keywords: transfer learning, feature alignment, learning with inductive bias, brain decoding, canonical correlation analysis

ÖZ

BEYİN ÇÖZÜMLEME İÇİN TRANSFER ÖĞRENME

Eryol, Erkin
Doktora, Bilgisayar Mühendisliği Bölümü
Tez Yöneticisi: Prof. Dr. Fatoş T. Yarman Vural

Şubat 2023 , 120 sayfa

İnsan beynini anlamak, bilimde uzun zamandır süregelen önemli bir problemdir. Bu tezde, beyin-davranış ilişkisindeki mekanizmaları ortaya çıkarmak için fonksiyonel manyetik rezonans görüntüleme (fMRI) verilerinden bilişsel bir durumu kestirmek üzere beyin çözümleme problemine odaklanıyoruz. fMRI görüntüleri üzerinde çalışmak, maliyetli veri toplama süreci nedeniyle zordur. Sonuç olarak, fMRI çalışmaları genellikle az sayıda denekle gerçekleştirilir.

Son yıllarda mevcut beyin çözümleme veri setlerinin artmasıyla birlikte, nörobilim alanındaki beyin çözümleme çalışmalarında transfer öğrenme yöntemleri uygulanabilir hale gelmiştir. Bu tezde, beyin çözümleme için transfer öğrenme olarak adlandırdığımız, farklı ama ilişkili küçük ölçekli beyin çözümleme çalışmasının performansını iyileştirmek için nörobilim alanındaki mevcut veri ve bilgiden faydalanıyoruz. Beyin çözümleme için transfer öğrenmeye ilişkin iki yaklaşım öneriyoruz.

İlk yaklaşımda, bir beyin atlası kullanarak özgün bir Yapılandırılmış Çok Katmanlı Algılayıcı önerdik. Yalnızca hedef veri seti üzerinde eğitilen Yapılandırılmış Çok Katmanlı Algılayıcı modelinin, büyük bir kaynak veri seti üzerinde önceden eğitilmiş

olan üç boyutlu evrişimli sinir ağı modeli ile eşit sınıflandırma ve yakınsama süresi performansına sahip olduğunu gözlemledik.

İkinci yaklaşımda, ortak bir deneysel paradigma ile elde edilmiş küçük ölçekli veri kümeleri üzerinde çalıştık. Bu yöntem, oturum, konu ve veri kümesi ilişkilerini dikkate alarak kümeler arasında transfer edilebilir öznitelikler üretmektedir. Deneylerde, önerilen her iki yöntemin de mevcut transfer öğrenme yöntemlerine göre daha iyi performans sağladığı gösterilmiştir.

Anahtar Kelimeler: transfer öğrenme, öznitelik hizalama, model varsayımı ile öğrenme, beyin çözümleme, kanonik korelasyon analizi

To my parents.

ACKNOWLEDGMENTS

I'm extremely grateful to Dr. Yarman Vural for her invaluable patience and feedback, carefully monitoring my progress, and providing direction throughout my Ph.D. program. I would also like to thank the thesis monitoring committee members, Dr. Emre Akbaş and Dr. Nazlı İkizler Cinbiş, who generously provided knowledge and expertise, and Dr. Sinan Kalkan and Dr. Tolga Çukur for taking part in the thesis jury and reviewing the thesis.

I am grateful to my cohort members and alumni for setting the high standards and forming the challenging and engaging research topic. I would also like to thank the ImageLab members for the inspiring seminars that kept me motivated under the COVID times. I am thankful to the Health Informatics department and the HASAT project members who formed a good working environment and supported my research during the earlier years of my Ph.D. journey.

Lastly, I want to thank my parents for their endless support all through my studies. Without them, it would be impossible for me to complete this study.

I acknowledge National Higher Education Council 100/2000 scholarship for their financial support and TÜBİTAK TRUBA for the computation service.

TABLE OF CONTENTS

LIST OF TABLES

TABLES

LIST OF FIGURES

FIGURES

LIST OF ABBREVIATIONS

ANN	Artificial neural network
BOLD	Blood oxygen level dependent signal
fMRI	Functional magnetic resonance imaging
HCP	Human Connectome Project
MMD	maximum mean discrepancy
MVPA	Multi-voxel pattern analysis
NIfTI	Neuroimaging Informatics Technology Initiative data format
ROI	Region of interest
SRM	Shared Response Model
SSTL	Shared Space Transfer Learning
SVM	Support vector machine
TL	Transfer learning

CHAPTER 1

INTRODUCTION

Understanding the human brain is one of the most important and long-standing challenges in science. The advancements in non-invasive measurement technologies have opened many new windows to observe the brain mechanisms related to our behavior in the last three decades. Functional magnetic resonance imaging (fMRI), developed in the early 90's, makes it possible to non-invasively measure the active regions in the brain. The fMRI is a big step ahead in measuring the brain activities with relatively higher space and time resolution, compared to the positron emission tomography technology. With the advancements of the new measurement technologies, neuroscience has become a vast and interdisciplinary research area.

In this thesis, among many problems in Neuroscience, we focus on the brain decoding problem, which requires the development of computational models for representing the cognitive models based on the neural data, such as fMRI signals. The mathematical tools in Computer Science, specifically the state of the art Machine Learning methods offer very powerful representation techniques for brain data. Thus, the research on the computational models of human brain lies at the intersection of Computer Science and Neuroscience.

Brain decoding experiments are designed to capture the active brain regions about a cognitive state from brain signals recorded as fMRI images to uncover the mechanisms in the brain-behavior relationship, as illustrated in figure 1.1. As it can be seen from this figure, a subject is exposed to a series of cognitive stimulus, such as listening to music or watching a series of images or videos. During this time span, at each time instant, the brain activities in a brain volume are recorded, generating a spatio-temporal fMRI data for the set of stimulus. Then, the brain decoding problem

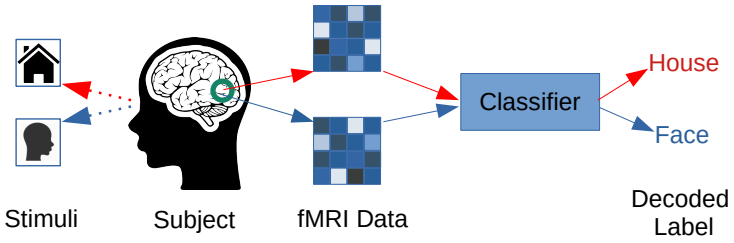


Figure 1.1: Brain decoding involves a set of stimuli presented to a subject, and the estimation of the stimuli from the fMRI signals. In the figure, the subject is presented two images; a house and a face image. Corresponding brain signals are input to a classifier to distinguish the presented images.

involves estimating the input stimuli from the fMRI recordings.

Brain decoding is an important tool in many neuroscience applications. Brain decoding methods help the analysis of higher cognitive functions, that involve multiple brain regions and highly varying brain responses. Furthermore, successful estimation of brain activity that corresponds to an external stimuli is the main tool in brain-computer interfaces. Brain decoding is also used in the discovery of biomarkers for detecting diseases and monitoring their progression, that is a non-invasive probe to measure the change in brain activity to a disease related stimuli.

Unfortunately, the fMRI data acquisition process is quite costly, time-consuming and it requires technical supervision, due to the limitations of fMRI technology and the magnetic field created by the fMRI equipment. The constraints imposed by the experimental setups limit the number of samples for each cognitive process, which creates serious training problems of the data hungry Machine Learning algorithms. Furthermore, the fMRI technology can only capture the brain activities as indirect measurements of blood oxygenation level. The measurements of the same stimulus on different subjects result in a large data discrepancy. This is also observed on different sessions of the same subject. The variations of the fMRI recordings for the same external stimulus results in a large data discrepancy in the analysis of brain mechanisms.

In Neuroscience domain, one of the most significant problems is the reproducibility problem (Marek et al., 2022). We expect the outcome of a brain decoding experiment

to hold, when the experiment is reproduced under the same conditions. However, obtaining the same experimental conditions is not possible, since there is a great deal of variation in brain signals, even when the same subject repeats the same experiment. As a result, findings in one neuroscience experiment may not hold, when it is reproduced in a similar environment. Reproducibility in a similar environment is referred as generalization problem in the Machine Learning literature.

There are two main challenges in the generalization performance of an estimated computational model for decoding the brain from fMRI data. The first challenge stems from the insufficient number of the fMRI data samples, where the number of samples measured in each cognitive state is much smaller than the dimension of the feature space. The fMRI image consists of a large number of volumetric elements ($N \approx 10^7$), called voxels, each of which corresponds to a dimension of the feature space. On the other hand, due to the technological limitations, for each cognitive class the number of samples are at most around couple of hundreds. The second challenge is rooted in the noise embedded in the fMRI samples. It is expected that the set of active neuron populations under the same external condition should not vary significantly among different sessions, subjects and datasets. However, there is a large variation in the set of active neuron populations, even between two sequential sessions of a single subject. The difference in the active neuron population set increases between sessions of two different subjects, and further increases between sessions of subjects from two different datasets.

Transfer learning (TL) methods apply the data and knowledge, gained in one problem domain to another related problem domain, where the former domain is referred as the source domain and the latter is referred as the target domain. Transfer learning includes a wide range of applications, depending on the form of source domain representation, transferred to the target domain.

One application of transfer learning is based on a model representation, where the performance of a model on target dataset is improved with the help of a model trained on source dataset. In this application type, the trained model is the form of source domain representation. Another application of transfer learning utilizes the available domain knowledge representation as a set of priors on the target domain.

As the amount of statistically sufficient data increases in the source domain, model based transfer learning becomes more viable. In the limited data regime, priors based on source domain knowledge have a higher impact on target model performance, compared to the model based transfer learning.

Transfer learning is prone to negative transfer, where the source domain representation degrades the performance of the model on the target domain. There are two main causes of negative transfer learning. The first cause is learning features that are unrelated to the problem at hand, called spurious features. The spurious feature problem is often addressed by constraining the model to learn under domain-specific prior information. The distribution discrepancy between source and target datasets is managed by adapting or aligning the distributions of the source and target datasets. The second cause is the discrepancy in data distributions between source and target datasets.

From the data perspective, we observe two main problems in the neuroscience domain. Firstly, most of the previous studies are carried out on small scale datasets, that have a limited number of subjects ($N < 30$ subjects). There are numerous small scale datasets, however their limited sample size becomes a limiting factor in the reproduction of the findings in these studies (Turner et al., 2018). Considering the high dimension of the feature space, which is about 10^7 voxels, these datasets are far from statistical sufficiency to learn and generalize a cognitive state. Fortunately, some of the available small scale datasets are obtained from studies that investigate a common cognitive **task** in the experiment. Secondly, a large sample size is a natural remedy for the reproducibility problem. However there are only a few large scale datasets, that include both a large number of subjects and tasks. Human Connectome (Van Essen et al., 2013) and UK BioBank (Sudlow et al., 2015) projects are examples that host a large-scale dataset. Yet, the diversity of tasks is not sufficient. The cognitive task in a given target domain may not match a common cognitive task in these large scale datasets.

In this thesis, our major goal is to utilize the data and domain-specific knowledge in the Neuroscience literature to improve the performance of a different but related brain decoding studies, that we refer as transfer learning for brain decoding. In the context

of this thesis, we follow two transfer learning approaches. In the first approach, we incorporate domain-specific structural information on large scale fMRI datasets and we propose the Structured Multi-Layer Perceptron. The large sample size enables applying the data hungry multi-layer perceptron for the brain decoding problem. We inform the model of the voxel labels, defined in a given anatomical brain map, called a brain atlas. In the second approach, we reduce inter-session, inter-subject and inter-dataset differences to generate a large scale source dataset. The proposed model learns to generate features on source datasets, and the model improves brain decoding performance in a related target dataset.

1.1 Motivation and Contribution of this Thesis

In the last decade, there is a drastic increase in the number of online repositories, which follow open science practices. These repositories to host various small-scale datasets with diverse tasks. Popular examples include, OpenNeuro (Markiewicz et al., 2021, formerly named OpenfMRI Poldrack and Gorgolewski, 2017) which hosts the raw data, and the Neurovault project (Gorgolewski et al., 2015), which hosts the statistical maps of numerous neuroscience studies. There are few large-scale projects, namely Human Connectome Project (HCP) (Barch, 2013) and UK Biobank (Sudlow et al., 2015) project, that focus on the ground up acquisition of fMRI data. These projects lead to an increase in the size and number of datasets available to the researchers, paving the way for the transfer learning methods to be utilized in the brain decoding problem. Frégnac, 2017 argues that the trend of increase in the available studies may help the neuroscience field reach new breakthroughs in understanding the brain-behavior relationship.

The size of the datasets has a great impact on the methodologies, developed for the brain decoding problem. Based on the size and type of the datasets, the research studies on transfer learning methods for brain decoding can be grouped under two headings.

1. Across-tasks TL; a large-scale model with a rich set of features trained on vast number samples of varying brain mechanisms, that learns from a diverse set of

tasks and re-calibrated to a target task.

2. Single-task TL; a deterministic model trained on a small scale datasets, that generalizes to a target task with a common underlying brain mechanism.

In this thesis, we propose solutions based on spatial priors and aligned features to improve transfer learning on the brain decoding problem.

Brain decoding studies on fMRI data either focus on a subset of voxels, called the region of interest, or consider coarse-grained statistics of whole-brain data. It is a new research direction to work on fine-grained whole-brain fMRI images. Furthermore, transfer learning on the fine-grained whole-brain data introduces new problems. Due to the limited fMRI data in the brain decoding domain, coarse grained statistics improve the transferability of a model trained on this new data form.

In our first study, we propose a novel model on fine-grained whole-brain fMRI data, called Structured Multi-Layer Perceptron. In this model, prior information of the fMRI data structure, called a brain atlas, is utilized. We compare the Structured MLP with a three-dimensional convolutional neural network model on transfer learning experiments. In these experiments, we observe that the Structured MLP model trained only on the target dataset has on-par classification and convergence time performance with the three dimensional convolutional neural network model, that is pre-trained on a large source dataset.

In the second study, we propose a transfer learning solution between datasets that follow a common experimental paradigm. The datasets are acquired from the subjects that perform a common set of tasks. We assume that there is a relatively small task-related pattern variation in these datasets, hence a series of linear transformations on the data representation can reduce the task-related pattern variation between these dataset, as opposed to nonlinear transformations. We further assume that there are multiple small-scale source datasets that follow the same experimental paradigm of the small-scale target dataset. We propose an improvement on generalized canonical correlation analysis objective function for the feature alignment based transfer learning problem, where we align source and target samples to obtain a common transferable hierarchical data representation. Each fMRI recording sample is ob-

tained in a data acquisition session from a subject of an experiment(dataset), hence each sample carry a session-subject-dataset tag. The subsumption relation between the datasets, subjects and sessions ($session \subset subject \subset dataset$) also forms the expected similarity of brain responses, namely two different sessions of the same subject are expected to be more similar compared to two sessions from two different subjects. We obtain generalizable features by extending the previous state of the art model (Yousefnezhad et al., 2020), incorporating brain regions as an invariant in the feature generation process. We further improve this representation with covariance profiles of brain regions. We achieve a substantial improvement compared to the state of the art brain decoding studies.

1.2 Summary of the Thesis

The thesis is organized in five chapters.

In chapter two, we define the concepts regarding functional magnetic resonance imaging technology. The steps that raw fMRI images go through, called preprocessing, are defined in this chapter. Furthermore, we give the mathematical notation used throughout the thesis.

In chapter three, we explain our Structured Multi-layer Perceptron model. We overview the literature on recent transfer learning methods for brain decoding with a focus on imposing a spatially structured bias on MLP models. We define the Human Connectome Project (HCP) Barch et al., 2013 dataset in detail. We compare the experimental results on HCP for the baseline 3D convolutional model and our Structure Multi-Layer Perceptron model in classification and convergence time criteria.

In chapter four, we develop a "feature alignment" model to generate transferable features. We overview the literature on hyperalignment in neuroscience domain, critique the shortcomings in recent work and give a background on generalized canonical correlation analysis. The recent benchmark in single-task transfer learning for brain decoding and our suggested method are defined in this chapter. We show that the suggested models improve the recent work in single-source and multi-source transfer learning experiments.

In chapter five, a summary of the thesis is given. We discuss the strengths and weaknesses of the two approaches of transfer learning for the brain decoding problem. We give future research directions for each of the proposed models in this thesis.

CHAPTER 2

NATURE OF FMRI DATA AND PREPROCESSING METHODS

In this chapter, details of datasets, preprocessing of fMRI data, acquisition of fMRI data and analysis of fMRI data are overviewed. In "datasets" section, we explain the neuroscience experiment types and the dataset, used in the following chapters. The raw fMRI recording goes through a series of steps, called preprocessing, that improve the fMRI image quality and standardize each fMRI recording to a common criteria. In the preprocessing of the fMRI data section, we define the intermediate steps that remove the irrelevant components of the fMRI recording. In acquisition of the fMRI data section, we define the terms related to the data acquisition process. In the analysis of the fMRI data section, we briefly explain the traditional methods used for fMRI analysis.

2.1 Nature of fMRI Data

It is known that an active brain volume consumes relatively more oxygen in blood than a passive brain volume (deoxygenation) (Poldrack et al., 2011). The fMRI device scans for changes in blood oxygenation, called blood oxygenation level dependent (BOLD) signal, as a measure of brain activity. The oxygenation-deoxygenation process follows a specific pattern, called haemodynamic response, where the consumed oxygen is over-compensated for, creating a peak, and falls back to a balance level after a short amount of time.

The fMRI device controls and measures the magnetic field. The magnetic field of the fMRI device forces atoms to be aligned in a specific polarization. At the event of neutralization of the magnetic field, the atoms retain their initial polarization over

a time period. Different brain tissues are distinguished by the time they obtain their initial polarization, measured by the fMRI device. This time dependent process makes the fMRI technology less effective in temporal resolution but more effective in spatial resolution.

In the context of this thesis, we refer to a dataset as the blood oxygen level dependent (BOLD) signal recordings in an fMRI experiment. The design of an fMRI experiment can be categorized into three groups; resting state experiment, task related experiment and naturalistic paradigm experiment.

The neutral brain state is referred as the resting state. Hence resting state experiments investigate patterns that distinguish subjects based on their neutral brain states.

On the other hand, the task related experiment refers to introducing an external stimuli to investigate the related brain activation patterns. There are two approaches in the task related experiment; event related task and block related task. The choice between event and block related task depends on the timing of successive stimuli. Due to the haemodynamic response curve, BOLD signal peaks in ≈ 5 seconds after the application of the stimuli and returns to the resting state in ≈ 15 seconds. Therefore, in an experiment with two or more different stimuli, switching from one stimuli to the other requires a time-period for the BOLD signal to turn back to the resting state. In a block task experiment, a stimuli is introduced in blocks of time intervals, that reduces the required resting period before switching to another stimuli. In an event related task experiment, the stimuli can be in an arbitrary.

In the naturalistic experiment type, the subjects in the experiment goes through the same experience, i.e. an story audio or video clip, rather than a specific timed stimuli.

Based on the order and duration of the stimuli during the experiment, there are two types of datasets. The term **homogeneous dataset** refers to the experiment design, where the stimuli sequence and duration of each stimuli is the same for all data acquisition sessions. Naturalistic experiments, where all subjects watch the same video clip or listen to the same audio clip, produce homogeneous datasets. The **heterogeneous** dataset term refers to an arbitrarily applied stimuli sequence and duration. Most of the task-fMRI datasets are in this category.

In the third chapter of this thesis, we work on datasets from the Human Connectome Project (HCP) Van Essen et al., 2013. There are seven different tasks in the HCP project, each comprised of a set of subtasks. Further details can be found in chapter 3.

In the fourth chapter of this thesis, hierarchical feature alignment for single task transfer learning, we work on four response-inhibition datasets, acquired in two separate studies. Further details can be found in chapter 4.

2.2 Preprocessing of fMRI Data

In the following, we briefly explain the several sources of noise in the data acquisition process, stemming from both the human factors and the data acquisition technology. The embedded noise is partially reduced by a series of initial steps, that form the preprocessing pipeline. This section briefly summarizes the common preprocessing steps in the Handbook of Functional MRI Data Analysis by Poldrack et al., 2011.

fMRI technology scans each unit of brain volume (voxel) as a function of time. In other words, at each time instant a single measurement of BOLD signal is recorded at a voxel, which consists of several thousand neurons. In order to generate the BOLD signals of the entire brain volume, we need to synchronize the measurements of the BOLD signals, recorded sequentially. A single three dimensional image is obtained by scanning one two-dimensional slice of the three-dimensional volume at a time. The time it takes to scan the whole brain is called the repetition time (TR). At each repetition duration, the timing of slices can slightly change. Synchronization of slices over repetitions is called slice timing correction.

Head movements of a subject generates a heavy noise in the measured brain volume consistency. This noise is handled by the registration of consecutive brain snapshots through time. One traditional registration method estimates a rigid body transform for three dimensional translation and rotation.

As mentioned in slice timing correction, a three-dimensional fMRI image is composed of two-dimensional sequential slices. A higher number of two-dimensional

slices lead to a higher spatial resolution. However, the time required to record all two-dimensional slices increases as the number of slices increase. Hence, the time resolution decreases as the number of slices increases. This forms a trade-off between the space and time resolution. The fMRI device allows to improve temporal resolution at the cost of spatial resolution by skipping two-dimensional slices, resulting in less two-dimensional scans per three-dimensional volume and a lower TR. A common strategy to benefit from high spatial and temporal resolution involves, firstly, taking a high spatial resolution snapshot. Secondly, images that have a low spatial resolution and high temporal resolution, are registered onto the static high-spatial resolution image.

Among subjects, although the brain anatomy is common, there are variations in various tissue shapes and sizes. Therefore, a registration of individual brain anatomy among subjects is required, called the co-registration step. In this step, brain volume of multiple subjects are registered among each other.

Several projects, i.e. Talairach, MNI, have worked on obtaining an anatomical common brain atlas from a large number of subjects. Registering of a subject's anatomy (high spatial resolution image) to the common brain template is referred as normalization.In order to further increase the signal to noise ratio of the spatio-temporal image, a Gaussian filter is applied to the entire three-dimensional brain volume. This step is referred as smoothing.

There are two common types of artifacts in the time domain; linear trends and low frequency artifacts. These artifacts are reduced by linear detrending and high-pass filtering referred as the filtering stage.

The above steps are minimal preprocessing stages, and can be further extended with quality control of output signals at each preprocessing stage.

2.3 Acquisition of fMRI Data

The data acquisition involves a subject entering the magnetic field of the functional magnetic resonance imaging (fMRI) device. In a task-fMRI experiment, a **subject** en-

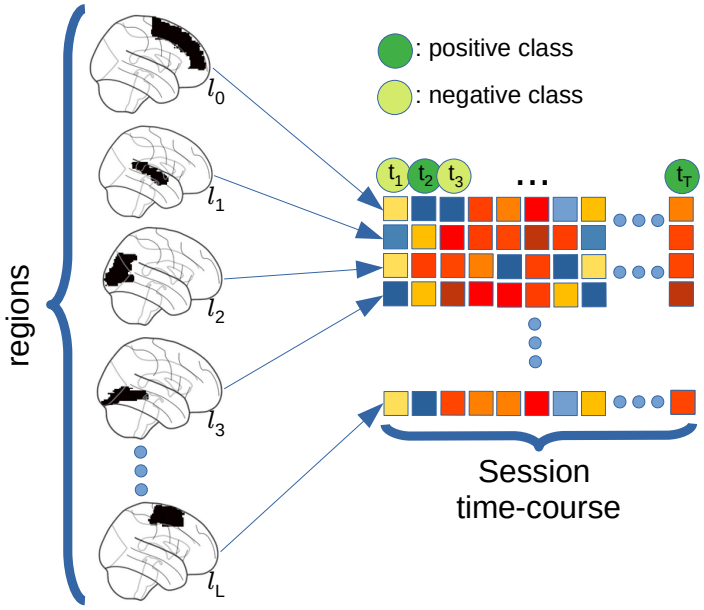


Figure 2.1: Voxel average intensities in a region l, at a time point $t \in \{0, .., T\}$ forms the regional time-series signal. Each time-point t belongs to a class, depending on the external stimuli in the experiment. Regions $l \in \{0, .., L\}$ are defined by a given brain atlas with L regions.

ters the magnetic field of the fMRI device, and conveys a given **task**, that is related to the hypothesis of the experiment. The experiments are carried out in sessions, where a **session** refers to the uninterrupted time interval that the brain signals are recorded based on the design of the experiment. In the context of this thesis, a **dataset** refers to the fMRI recordings with the associated task labels, obtained in an experiment.

A single session includes sequential brain images from a single subject, where each image is labeled with the task-related cognitive state, a single brain volume is a three dimensional recording of BOLD signals, and the smallest volumetric measurement unit is called a **voxel**. A brain atlas is a parcellation of voxels into regions, where voxels in each region is grouped under a region name. The brain atlas can be acquired either based on anatomical differences among brain tissues or based on activation patterns of voxel groups.

Region time-series data, illustrated in figure 2.1, is the average voxel intensity per region over the session time-course. During a session, each time-point in the session time course is assigned a cognitive state, related to the external stimuli.

2.4 Notation for Formal Representation of fMRI Data

Throughout the thesis, we use the following notation to represent various quantities of fMRI data.

In mathematical notation, the smallest spatial measurement unit is called a voxel, $X(w, h, d) \in \mathbb{R}$, where $w \in W$, $h \in H$, $d \in D$ refer to indices in width, height and depth dimensions, and W, H, D refers to the size of each dimension. There are $W \times H \times D$ voxels in a single image. Each session $r \in \mathbb{Z}$ is composed of $t \in T_r$ time-points of three dimensional images, forming a four-dimensional matrix, $X_r(w, h, d, t)$, where, T_r is the time duration for session r, and R is the set of all sessions.

In an experiment, there are multiple sessions, r, per subject, s, and multiple subjects, s, in a single dataset, d, indexed $X_{r,s,d}$. Note that each session data, r, belongs to a subject, s, and each subject is part of a dataset, d, thus forming an hierarchy among sessions.

In the normalization step of the preprocessing pipeline, each subject's brain volume is registered onto a common brain anatomy, called a brain atlas. The normalization step maps the voxels of a brain atlas A, represented as a three dimensional matrix of the same size of a single brain image X. Each voxel of the brain atlas $A(w, h, d)$ is assigned a label $l \in 0, ..., L$, where l is the identifier of an anatomical brain region. A brain region with identifier l involves a set of voxels $X(w_l, h_l, d_l)$, where (w_l, h_l, d_l) are voxel indices, such that $A(w_l, h_l, d_l) = l$. Brain regional time series data is formed by taking the average intensity of voxels with the same brain region identifier, illustrated in figure 2.1.

2.5 Chapter Summary and Conclusion

In this chapter, we defined the basic terminology that we use throughout the thesis. The fMRI technology related details are explained in the "Nature of fMRI Data" section. In the next section, "Preprocessing of fMRI Data", we explain the steps in the pipeline that takes the raw fMRI device measurements to the design matrix values

Table 2.1: Notation used in the thesis

Variable	Explanation
$W \in \mathbb{Z}, H \in \mathbb{Z}, D \in \mathbb{Z}$	Width, height, depth dimension sizes
$X \in \mathbb{R}^{W \times H \times D}$	3D BOLD signal snapshot
$X(i,j,k) \in \mathbb{R}$	BOLD intensity value of the voxel indexed at (i,j,k)
r, s, d	Session, subject and dataset index
T_r	Number of time-points in session r
$X_r \in \mathbb{R}^{T_r \times W \times H \times D}$	4D data from session r with T_r time-points
$X_{r,s,d}$	4D data from session r, subject s, dataset d
$A \in \mathbb{R}^{W \times H \times D}$	Brain atlas, a parcellation of the voxels in X
$A(i,j,k) = l \in \mathbb{Z}$	The brain region label l of the voxel indexed at (i,j,k)
$X(w_{l_i}, h_{l_i}, d_{l_i}) \in \mathbb{R}$	ith voxel BOLD intensity in the brain region indexed l
$X_r^l \in \mathbb{R}^{T_r}$	Mean BOLD time-series data of brain region indexed l, in session r
$l_i \in L$	L is the set of brain region labels, there are $\|L\|$ different labels

that we use in the analysis of the fMRI data. In the "Acquisition of fMRI Data" section, we define the terms related to the acquisition of fMRI data. In the "Notation" section, we explain the mathematical definition of the common terms referred in the remainder of the thesis. In "Analysis of fMRI data for Brain Decoding" section, we overview the two traditional methods to analyze the fMRI data for the brain decoding problem.

In the next chapter, we propose an ANN model that incorporates a brain atlas that jointly reduces session differences and classifies the cognitive states.

CHAPTER 3

STRUCTURED MULTI LAYER PERCEPTRON FOR ACROSS-TASK TRANSFER LEARNING

Brain decoding introduces a way to probe the mechanisms that lead to a cognitive behavior. In the brain decoding problem, we record the fMRI data of a subject, who responds to different external stimuli. We train a classifier on the recorded data, where the external stimulus is the class information. The trained model allows researchers to analyze the role of brain regions that leads to a behavior under the same stimuli type.

The non-invasive exploration of functional relation between brain readings and subject behavior is an important asset of brain decoding research. Furthermore, brain decoding enables designing markers of certain diseases and allows tracking the progress of these diseases.

The variation in brain readings make the solution of the brain decoding problem a great challenge. It is difficult to estimate the subject-specific variation of the brain readings. Hence, an important obstacle is the estimation of individual-to-individual and time-to-time differences in brain readings. Transfer learning methods are proposed in the literature to reduce the negative effect of the variation in brain readings. In transfer learning, we utilize the source data or domain knowledge, to improve the performance of a model on a related target dataset.

In this chapter, we propose a model that jointly learns to normalize the differences between different sessions of data and classify the brain readings. In the context of this chapter, we treat the domain knowledge as the source information in our model for the transfer learning problem, opposed to a source dataset.

Multiple studies (Gao, 2019; Wang, 2020) have shown that transfer learning improves model performance in brain decoding. In this work, we adapt a recent multi-layer perceptron (MLP) model (Tolstikhin et al., 2021) to the brain decoding problem. In our model, called Structured Multi-Layer Perceptron (MLP), we suggest a flexible model, based on the MLP-Mixing block (Tolstikhin et al., 2021), with the following properties. Structured MLP decomposes an fMRI image into volumetric patches. The patched representation allows treating each patch individually. In other words, we can both discard irrelevant patches and group together relevant patches. Recall that, a brain atlas is a three dimensional fMRI image mask that defines the volume of each brain region. The structured MLP utilizes a brain atlas, called Automated Anatomical Labeling (Tzourio-Mazoyer et al., 2002), in the normalization layer.

Structured MLP enables training on the whole brain fMRI images with approximately 10^7 voxels, where traditional methods are not practically applicable on such large amount of parameters. On the downside, it is hard to stabilize the convergence of the structured MLP model, and it is difficult to interpret the converged model, due to the black box structure of the non-linear neural network.

Furthermore, we re-implement a recent study by Wang, 2020, that proposes a three dimensional convolutional neural network for the brain decoding problem. We show the superior performance of our structured MLP on the convergence speed compared to the recent study.

In the following sections, firstly, we review the literature of structured learning and transfer learning methods, that inspired us to develop the suggested MLP algorithm. Secondly, we define our Structured MLP model in detail. Thirdly, we compare our method to the baseline three dimensional convolutional neural network. In the final section, we summarize our work and discuss the strengths and weaknesses of our model.

3.1 Literature Overview

In this section, we explain the transfer learning method, called fine-tuning. Then, we survey how spatial information is integrated into a model, where we overview the

methods in the general ML research and the neuroscience applications.

3.1.1 Fine-tuning for Transfer Learning

Yosinski et al., 2014 shows that early layers of a convolutional neural network learn "Gabor-like" weights, which are linear filters used in texture analysis. The linear filter-like early layer weights are general features that can generalize across many vision tasks. Besides, the final convolution layers learn dataset specific weights. This suggests that there is a transition from general to task specific features from initial to last layers. They measure the "generality-specificity" of each layer of a convolutional neural network model by freezing the first k convolutional layers of a pre-trained model with N layers, and remaining layers are retrained on target dataset. This routine evaluates the generality of the first k layers of the neural network.

Fine-tuning is the transfer learning method in the brain decoding application, that we compare with our proposed model performance.

Negative transfer refers to the case where applying the transfer learning method worsens the performance of a model on a target dataset. In the following, we explain the reasons behind negative transfer.

3.1.2 Reducing the Negative Transfer Between the Source and Target Datasets by Co-Registered fMRI Recordings

In Neuroscience domain, neural network models with a large training set are applied in the following studies; Wang, 2020, Y. Zhang et al., 2020, Gao, 2019 and Thomas and Samek, 2019. Transfer learning experiments on these models are limited to fine-tuning of pre-trained models on large scale datasets. However, fine-tuning is prone to negative transfer due to two reasons. Firstly, the learned features might fail to capture the common activation patterns of the task at hand on both source and target datasets, yet still match on nuisance features, for instance the head motion related error in the BOLD signal in both datasets. Secondly, although the learned features are a good representation of the mechanism of the task on the source distribution,

they may worsen the performance on the target dataset, due to the high discrepancy between the distributions of the datasets.

In the next subsection, we survey models that incorporate the available spatial prior information, which may reduce negative transfer learning. In the next subsection, we survey the methods on incorporating spatial prior information, that we refer as "spatially structured bias".

3.1.3 Survey on Incorporating Spatially Structured Bias

In domains with a spatial structure, spatial regularization methods allow specialization of model parameters to the spatial coordinates. Spatial regularization has been an active topic in Neuroscience, on neural network models (Q. Wu et al., 2016), on support vector machines (Sun et al., 2019), on graphical models (Cai et al., 2020) and on linear models (Beer et al., 2018). It has also attracted attention in machine learning field (Bach et al., 2012; Hernández-garcía and König, 2016; Kim, 2018; Kong et al., 2016; Scardapane et al., 2017; R. Wu and Kamata, 2018).

General-purpose Computer Vision methods presented in this section can be trivially generalized to the three dimensional spatial processing in the brain decoding problem. We group the methods in this section as non-local methods, context encoding with conditional random fields, set constraints in supervision and non-regular spatial methods.

Wang et al., 2018 introduces non-local networks to avoid the drawback of standard convolution. Assuming the image is partitioned into non-overlapping, equal size windows in a lattice form, a patch is the single window of image where patch neighborhood is the set of other patches that share a border with the center patch. In standard convolution, distant (non-neighboring) patches of image are related only after multiple convolution layers. Authors formulate

$$y_i = \frac{1}{C(x)} \sum_{\forall j} f(x_i, x_j) g(x_j), \tag{3.1}$$

where x_i is an input patch at each position i, g is a unary embedding, f is a pair-

wise embedding, j are non-neighbor patch indices and $C(x)$ normalizes over all non-neighbor patch indices j, $C(x) = \sum_{\forall j} f(x_i, x_j)$.

f and g are shown to be general functions, where f can be a kernel embedding or a Gaussian embedding, and g can be any local function, for instance convolution operation.

Context encoding with conditional random fields is another method to impose spatially structured bias. Lin et al., 2016 aim to explicitly model object and background pairs. They employ convolutional sub-networks to model unary and pairwise relations between patches, and extract multi-scale features with convolution modules.

The brain anatomical regions are formed by a set of voxels. Voxels in each brain region are assumed to be permutation invariant, such that the order of voxels inside a brain region are arbitrary. A recent work by Fayyaz and Gall, 2020 proposed constraining temporal transformer with sets, that are permutation invariant, for action recognition problem.

Another approach to impose a structural bias is through normalization of the artificial neural network. The normalization methods are applied in two distinct approaches. In the first approach, Ioffe and Szegedy, 2015 and Li et al., 2019 apply normalization to the tensor feature map, which is the activation function output. In the second approach, Qiao et al., 2020 normalizes the weights of the convolution kernel. Figure 3.1 illustrates the recent normalization methods, used in convolutional models, where batch norm (Ioffe and Szegedy, 2015), layer norm (Ba et al., 2016), instance norm (Ulyanov et al., 2017), group norm (Y. Wu and He, 2018) operate on the tensor feature map and weight standardization (Qiao et al., 2020) normalizes the convolution kernel weights.

In Z. Zhang et al., 2019 and Wang et al., 2019, authors impose structure by grouping the indices of the channel dimension in a convolutional neural network. L. Chen et al., 2020 and Norouzi, 2022 impose structure as stochastic weight pruning via modifying the dropout regularization (Hinton et al., 2012) method.

An important feature of Alexnet (Krizhevsky et al., 2012) is the distribution of the computation to multiple lanes via grouping the channels.

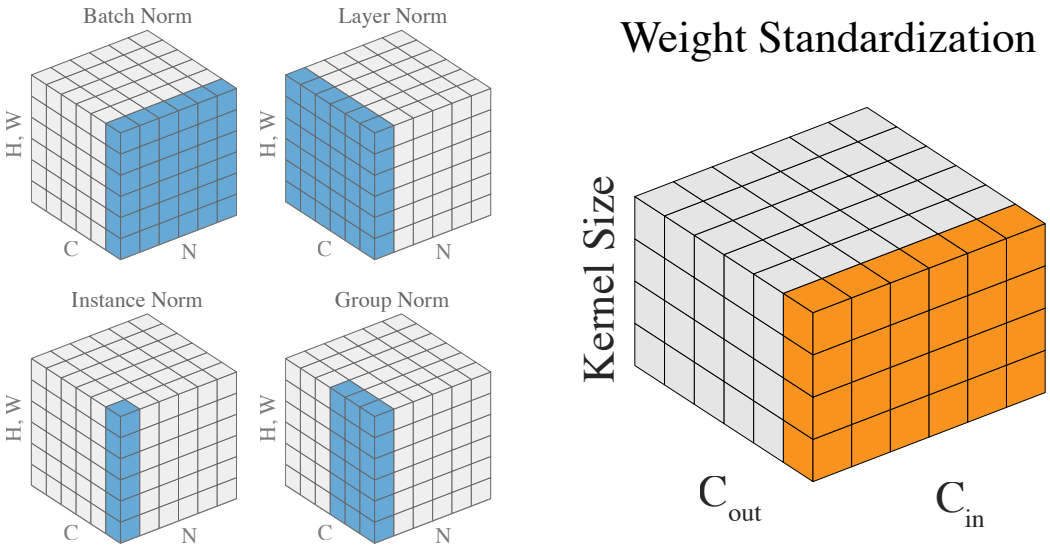


Figure 3.1: Recent normalization methods. Batch norm operates all samples in a batch. Layer norm operates on the channel dimension of each sample. Instance norm operates on a single channel of a single sample. Group norm operates on a uniform group of channels in a single sample. Weight standardization normalizes the kernel itself, and the rest operate on the feature tensor map. Illustration from Qiao et al., 2020.

Z. Zhang et al., 2019 generalizes grouped convolutions with an adjacency matrix to represent the relationship between input and output set of channels. They expose the adjacency matrix as a learnable set of parameters.

Batch normalization (Ioffe and Szegedy, 2015), illustrated in 3.1, estimates the channel mean and standard deviation of all batch samples. Each sample is normalized by the channel mean and standard deviation. At test time, running average statistics is used for normalization. A drawback of batch normalization is the dependency between batch size and tensor feature statistics, which causes a distribution mismatch between training and test sets, in addition to the test set data distribution difference.

Another normalization method, layer normalization, finds standardization parameters over all channels of a single sample in a minibatch.

Instance normalization (Ulyanov et al., 2017) operates over a single channel and single sample.

Weight normalization (Qiao et al., 2020) method standardizes filters rather than the channels. Batch normalization does not work well on small batch size. Grouped normalization works similar to layer normalization, where the standardization is not on all but groups of channels for a single sample. Hence, this operation is not affected by the drawbacks of the batch normalization.

Grouped normalization (Y. Wu and He, 2018) mitigates the drawback of Batchnorm (Ioffe and Szegedy, 2015), where the change of minibatch size in training and test sets degrades the model performance. Qiao et al., 2020 improves grouped normalization with weight standardization, where they simply standardize the weight matrix in the convolutional layer .

Li et al., 2019 proposes positional normalization, that defines the normalization parameters (μ, σ) as follows,

$$\begin{aligned} \mu_{b,h,w} &= \frac{1}{C}\sum_{c=1}^{C} X_{b,c,h,w}, \\ \sigma_{b,h,w} &= \frac{1}{C}\sqrt{\sum_{c=1}^{C}(X_{b,c,h,w} - \mu_{b,h,w})^2 + \epsilon}, \end{aligned} \tag{3.2}$$

where tensor features $X_{b,c,h,w}$ are normalized across the channel dimension $c \in C$, and normalization parameters $\mu_{b,h,w}$ and $\sigma_{b,h,w}$ are defined for each spatial location (h, w) and batch index b. The normalization parameters are estimated by marginalizing out the channel dimension.

In standard grouped convolution, there are equal number of channels per group, $\frac{C}{G}$, where C is the number of channels and G is the number of groups. We denote the output features of a layer as $o_{i,j}$. The channels $o_{i,j}$ are split into each group $o^g_{i,j}$. We define $o_{i,j}$ in terms of $o^g_{i,j}$. The output feature at coordinate (i,j) for group g, $o^g_{i,j}$, is found as follows,

$$o^g_{i,j} = \sum_{m=0}^{k-1}\sum_{n=0}^{k-1} f_{(i+m)(j+n)} w_{mn}, \tag{3.3}$$

where f is the feature map, $i \in \{1, .., H\}$, $j \in \{1, ..., W\}$, (W,H) are width, height dimensions, w_{mn} are the kernel weights, and m,n are indices on the kernel of size $k \times k$.

The output features $o_{i,j}$ are defined as,

$$o_{i,j} = o_{i,j}^{1} \cup o_{i,j}^{2} \cup ... \cup o_{i,j}^{G}, \tag{3.4}$$

where $i \in \{1, .., H\}$, $j \in \{1, ..., W\}$, (W,H) are width, height dimensions and $o_{i,j}^{g}$ is the gth output feature group for $(i, j)th$ coordinate.

Dynamic grouping relaxes the group norm by allowing arbitrary connections between input and output channels. The dynamic grouping (Z. Zhang et al., 2019) is given below,

$$o_{i,j} = \sum_{m=0}^{k-1} \sum_{n=0}^{k-1} f_{(i+m)(j+n)}(U \odot w_{mn}), \tag{3.5}$$

where the same notation in equation 3.3 is applied and, additionally, $U \in \{0, 1\}^{C^{in} \times C^{out}}$ is the binary grouping matrix, that maps the input to the output channels.

The soft grouping method in the literature is based on stochastic weight pruning method, called dropout. Dropout is used for stochastically pruning network weights to reduce overfitting and stabilizing training. Originally, this method is applied with constant probability on all image or intermediate feature locations. Spatial constraints is imposed on dropout in L. Chen et al., 2020 and Norouzi, 2022.

In this thesis, we work on a spatially structured bias, defined by a brain atlas. The recent studies on the brain decoding problem apply the spatially structured bias on artificial neural networks. In the following, we overview the recent methods on the brain decoding problem that incorporates a brain atlas on artificial neural networks.

3.1.4 ANN with a Spatially Structured Bias on fMRI Data

Habeeb and Koyejo, 2020 and Aydöre et al., 2019 propose methods to incorporate a brain atlas for brain decoding problem on fMRI data.

In Aydöre et al., 2019, fully connected layer weights are masked with a binary grouping matrix. Rather than learning the grouping dynamically, they use a clustering method to generate a bank of candidate matrices via Ward clustering, before training. One sample is taken from the bank and used for masking the weights. One drawback of this method is that the grouping matrix is only applied at the initial layer, rather than intermediate ones.

Habeeb and Koyejo, 2020 proposes fixed grouping layer as a modified MLP layer. The fixed grouping layer takes $x \in \mathbb{R}^{n_i n, c_i n}$ as input, where $n_i n$ is the number of input vectors and $c_i n$ is the input vector length. The layer output $z \in \ltimes_{\rtimes \cong \approx}, {}_{\rtimes \cong \approx}$ is calculated as,

$$z = A(xv \odot u) + b,$$

where A is a binary matrix of size n_{out}, n_{in} that maps the number of input vectors to the number of output vectors, $v \in c_{in}, c_{out}$ maps the input vector length to the output vector length, $u \in n_{in}, c_{out}$ and $b \in n_{out}, c_{out}$ is the bias. The pattern in the binary matrix A defines the spatial grouping of samples. Parameters u, v and b are learned via backpropagation algorithm. Fixed grouping layer is evaluated on GLM contrast maps of the Human Connectome Project dataset. This method is shown to outperform three-dimensional convolution on the brain decoding problem, where the input and the kernel are three dimensional matrices. Furthermore, as a baseline method, they propose to apply CoordConv (Liu et al., 2018) on 3D convolution, which incorporates brain region labels as an additional channel information.

3.1.5 Literature Survey for the Background of Suggested Structured MLP

In the suggested Structured MLP, we employ a multi-layer perceptron. Furthermore, we compared our structured MLP model with a three dimensional convolutional neural network.

In the following subsections, firstly, we overview the multi-layer perceptron model. Secondly, we define the convolutional neural network model. Thirdly, we define the three dimension convolutional neural network that is proposed for the brain decoding problem. The first two subsections, multi-layer perceptron and convolutional neural network definitions, are summarized from Duda et al., 2001 and Mitchell, 1997.

3.1.5.1 Multi Layer Perceptron

The perceptron is a single neuron that assigns a weight value to each dimension of the input and applies a threshold on the weighted input. The output of the threshold function forms a decision surface, defined in the input space, where each sample is assigned a label depending on which side of the decision surface it lies on. The perceptron function f is defined as follows,

$$f(x) = sgn(w \cdot x), \tag{3.6}$$

where $x = [1, x_1, ..., x_N]$ is the input vector, w are the weights $w = [w_0, w_1, ..., w_N]$, the dot product of w and x is $\sum_i w_i x_i$ and $sgn()$ is the sign function. The sign function for some scalar input s is

$$sgn(s) = \begin{cases} 1, & \text{if } s > 0, \\ -1, & \text{otherwise.} \end{cases} \tag{3.7}$$

The weights w that define the decision surface are learned from data. The perceptron training rule updates the weights w as,

$$w_i \leftarrow w_i + \eta(t - sgn(w_i \cdot x_i))x_i,$$

where η is a constant, called the learning rate and the second term updates the weights in the direction that reduces the distance between target t and perceptron output. The perceptron converges to the optimum solution when the samples are linearly separable and may not converge otherwise.

The delta rule asymptotically converges to the minimum error weights w in the linearly non-separable case. The delta rule searches for the best weight parameters w via gradient descent algorithm. This rule measures the training error J that is summed over all samples,

$$J(W) = \frac{1}{2}\sum_{i \in D}(t_i - w \cdot x)^2,$$

where t_i is the label of ith sample, $w \cdot x_i$ is the output for the ith sample in the training set D. The weights w are updated in the steepest decrease direction with respect to the training error,

$$\nabla J(w) = \left[\frac{\partial J}{\delta w_0}, \frac{\partial J}{\delta w_1}, .., \frac{\partial J}{\delta w_n}\right].$$

Therefore, the gradient descent update is

$$w \leftarrow w - \eta \nabla J(w).$$

Multi-layer perceptron (MLP), also referred as multi-layer neural network, is composed of three types of layers; an input layer, single or multiple hidden layers and an output layer. Hidden and output layers include multiple neurons. For a network with K output neurons, kth neuron output is as follows,

$$net_k = f(\sum_{j=1}^{H} a(o_j w_{kj} + bj)), \tag{3.8}$$

where a is the activation or thresholding unit function, H is the number of neurons in the previous hidden layer, j is the index of the neuron among the hidden layer neurons, o_j is the output of jth neuron's activation in the layer, w_{kj} is the weight between kth output neuron and jth hidden layer neuron, b_j is the scalar, called the bias of the neuron.

The error function J is computed starting from the weights of the output layer neurons. The error function J, sequentially propagates backward layer by layer with the chain rule, given below,

$$\frac{\partial J}{\partial w_{ji}} = \frac{\partial J}{\partial o_j}\frac{\partial o_j}{net_j}\frac{net_j}{\partial w_{ji}}, \tag{3.9}$$

where o_j is the jth hidden unit output.

3.1.5.2 Convolutional Neural Network

Some of the concepts in structured MLP are based on the convolutional neural network (CNN) architecture. We briefly define these concepts in this subsection.

In a convolutional neural network, a convolution kernel moves over spatial coordinates and weights of this kernel is optimized to reduce an error function J, in equation 3.9. In a convolutional neural network, the number of weight parameters is a function of the number of kernels. Compared to a fully-connected MLP, CNN reduces the number of parameters to optimize. For a whole spatial sliding operation, the kernel weights are shared at each location of the image. For two dimensional inputs, i.e. a 2D image, translations of the image produce the same shared weights and the same output. The two dimensional kernel is $k \in \mathbb{R}^{w_k,h_k}$, where the image patch of size $w_k \times h_k$ is called the receptive field of the first layer kernel. In the implementation of a CNN, each kernel produces a channel of the image that is the resulting kernel output at each spatial coordinate. The number of channels is equal to the number of kernels. In a CNN implementation, the number of channels is referred as the depth dimension, d. The number of kernels at each layer is not constant, hence for each layer, the number of inputs are formed by the output channels of the previous layer.

With no loss of generality, a convolutional neural network can be generalized to three dimensions, where a kernel has a 3 dimensional receptive field.

Note that a fully connected MLP layer becomes equivalent to a CNN, when the 2 conditions are met;

- the kernel size of a CNN layer is equal to the whole image/channel size,
- the number of kernels at one CNN layer is equal to the number of MLP neurons in the next layer.

These conditions form a one-to-all relation between channel output of one kernel and channel input of the next layer, hence channels are fully connected.

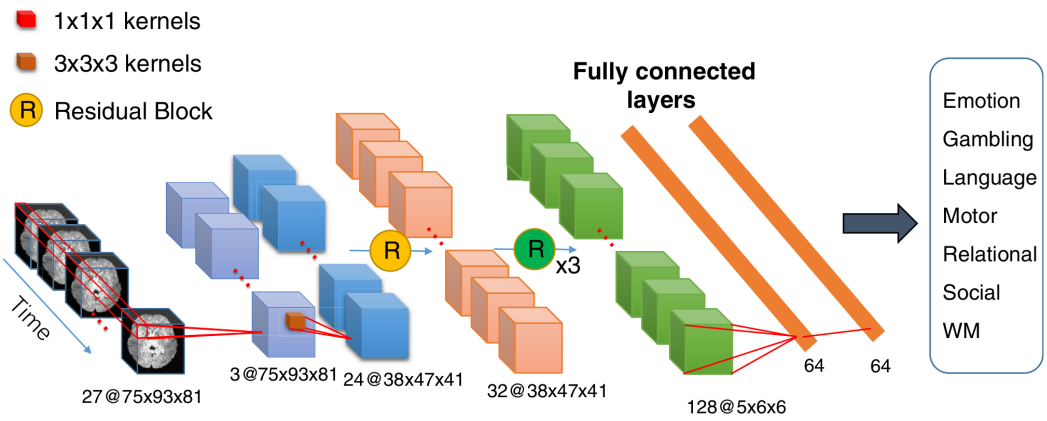


Figure 3.2: The architecture used in Wang, 2020 that implements a 3D CNN model. Image from Wang, 2020 is used under CC BY license.

3.1.6 3D Convolutional Neural Network Baseline Method

We reproduced the brain decoding method of Wang, 2020 that learns a set of spatially global filters. In the model, 3D brain snapshots of 27 sequential time points are used as input. They treat each time sample as an input channel and reduce the number of channels with 1x1x1 kernels. In temporal domain, 1x1x1 convolution is used to learn an intermediate semantic representation of 3 time-points in the first layer.

Next, 3D convolutional layers with skip connection, called residual blocks, are applied. Each residual block includes a convolution layer with 3x3x3 size kernels, Rectified Linear Unit (ReLU) activation and batch normalization (BN). Four residual blocks are applied in the proposed architecture.

Finally, two fully connected layers are used to generate the final output via softmax function.

The training routine involves a temporal data augmentation method where a different time sub-interval is sampled from the total time interval.

Spatial input size and number of channels in each layer is shown in the figure 3.2.

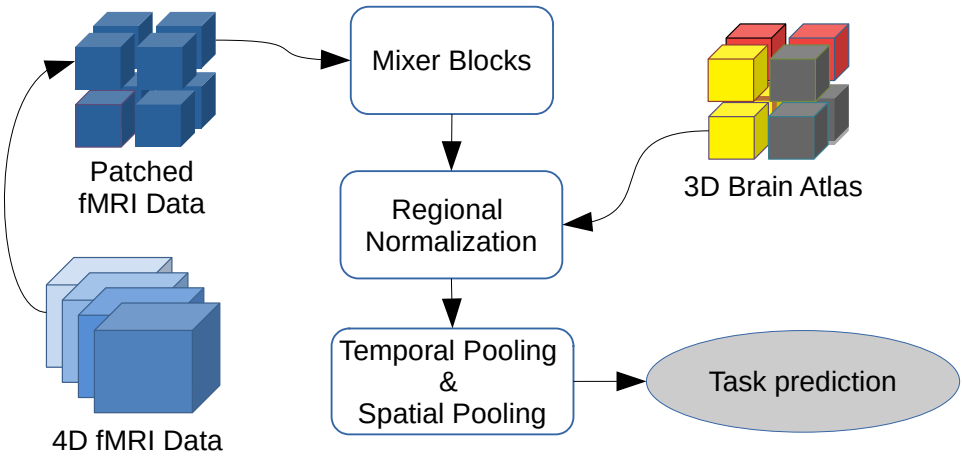


Figure 3.3: Structured MLP accepts four dimensional fMRI data. The model decomposes input into patches of non-overlapping, equal sized patches. It adapts mixer blocks (Tolstikhin et al., 2021) and applies regional normalization, where per-voxel region information is obtained from a brain atlas. Finally, the temporal and spatial dimensions are pooled sequentially.

3.2 Structured MLP for Across-Task Transfer learning

In our suggested structured MLP model, we engineer an Artificial Neural Network (ANN) model, that incorporates a common spatial map across subjects, provided by the brain anatomic atlas. The model is illustrated in figure 3.3.

As explained in chapter 2, fMRI images go through well-established co-registration steps that leave a common coordinate space among subjects. Furthermore, behavior-specific brain activation maps that are related to certain tasks, often has a form of spatial affinity to a voxel population. We aim to follow the spatial guidance and build a model that implements necessary inductive biases on whole brain fMRI images.

As illustrated in the figure 3.3, we firstly form volumetric patches of voxels. Secondly, we encode the spatial image through a series of mixer block layers. Thirdly, we apply regional normalization, where a region is a set of patches with the same region label, which are obtained from a brain atlas. Finally, we sequentially pool the temporal and spatial dimensions and predict the cognitive task.

3.2.1 Data Representation Challenges

Recall that fMRI technology generates a collection of voxel intensity values, which forms a brain volume at each time instant. Therefore, the data has four dimensions in width, height, depth and time.

In the following, we explain the difficulties in working with the four dimensional fMRI data and the model design challenges. We aim to address these challenges in our model.

Inferring functional similarities or dissimilarities of voxels from fMRI data depends on changes in signal intensities over time. However, 4D fMRI data has a much lower temporal resolution compared to the spatial resolution.

The statistically sufficient number of samples required to fit a model grows exponentially in the number of data dimensions. Unfortunately, the 4D fMRI data has a very high number of dimensions ($N \approx 10^7$), compared to the available number of samples ($N < 10^3$).

However, contrary to general computer vision problems, spatial registration and brain atlas labeling of 3D fMRI image coordinates provide an approximately common anatomical region label for each voxel at each time. Furthermore, certain stimuli excite a common brain activation pattern across different runs of data acquisition sessions. Therefore, applying a brain atlas as a static graph over the 3D coordinates of an fMRI image can generalize better compared to assuming a uniform lattice. Incorporating the brain atlas in the model definition introduces a model design challenge.

Allowing functional specialization of model parameters in a region, introduces an important challenge in formalizing the brain decoding problem. Convolutional neural networks share kernel parameters among all positions for translation invariance. The structure in the spatial domain of fMRI data contradicts the shared parameter assumption. Each anatomical/functional region has characteristic properties. Region specific non-shared parameters can address the characteristic spatial properties in the data.

However the number of required parameters increases in multiples of the number of regions, leading to an impractically large model. Therefore, keeping the number of

learnable parameters manageable introduces another challenge depending on the data and domain properties.

A further problem is the large variation in fMRI patterns across the subjects that perform the same cognitive task. Although the number and shape of the anatomical regions are assumed to be the same for all subjects, there are significant differences in functional role of these regions across subjects. Therefore, the dynamic structure needs to be accommodated in the architecture.

fMRI experiments often select a target set of voxels in the region of interest (ROI). This requires ways to omit/choose regions and work on a specific set of regions across subjects. The flexibility of the model in grouping or discarding voxels is a common requirement in the brain decoding problem.

In the next section, we address the data representation challenges and introduce the proposed Structured MLP model.

3.2.2 Structured MLP Model

In this section, we define the Structured MLP model, which has three main components. The first component decomposes the fMRI image into non-overlapping uniform volumes, called patches. The second component modifies the MLP-block (Tolstikhin et al., 2021) that regards the temporal order of three dimensional volumes in the channel dimension. The third component is the final fully connected layer that reduces the temporal and spatial dimension sequentially.

We start with the definition of the patch decomposition, illustrated in 3.4. The four dimensional fMRI image sample is reprepsented by a tensor, $X \in \mathbb{R}^{W,H,D,T}$, where W, H, D, T denotes width, height, depth and time dimensions respectively. We split the three dimensional coordinates W, H, D into non-overlapping cubic volumes, called patches. A patch P_i is defined such that $P_i \in \mathbb{R}^{p_w,p_h,p_d,T}$ and $p_w, p_h, p_d << W, H, D$. Therefore, an image sample X is given as,

$$X = \bigcup_{i=1}^{N_p} P_i,$$

where $N_p = W/p_w \times H/p_h \times D/p_d$ is the number of patches in an image.

Table 3.1: Chapter 3 - Structured MLP notation

Variable	Explanation
P, T	Number of patches and time-points in a session sample
$\bar{X} \in \mathbb{R}^{(T,P)}$	Design matrix of size $P \times T$
p_w, p_h, p_d	Width, height, depth dimension sizes of a patch p
$x_p \in \mathbb{R}^{p_w \times p_h \times p_d}$	3D patch data of $p_w \times p_h \times p_d$ voxels indexed p, where $X = \bigcup_p x_p$.
$N_p \in \mathbb{Z}$	Number of patches in a 3D fMRI image, $N_p = W/p_w \times H/p_h \times D/p_d$
X_{R_i}	Patches in the ith brain atlas region R_i
$A \in \mathbb{R}^{W,H,D}$	Brain atlas A
a_i	Set of voxel labels in patch i
l_{a_i}	Brain region label of brain atlas patch i
$\mu(.), \sigma(.)$	Column mean and standard deviation w.r.t. input matrix
γ, β	Scale and bias scalar values
M_k	kth mixer block
W	MLP layer weight
$ReLU(.)$	Rectified Linear Unit thresholding function, that assigns 0 to all negative values in the input.

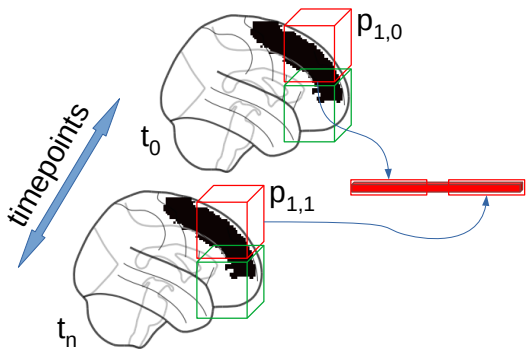


Figure 3.4: Illustration of time-series data for a single patch $p_1 \in \mathbb{R}^n$.

As we mention in section 2.3, brain atlas $A \in \mathbb{R}^{W \times H \times D}$ has the same three dimensional size as a single snapshot of the fMRI image X. $A(w, h, d) = l \in \mathbb{Z}$, where $l \in \mathbb{Z}$ is a brain region label. We partition the brain atlas A into patches as follows,

$$A = \bigcup_{i=1}^{N_p} a_i,$$

where $a_i \in \mathbb{R}^{p_w, p_h, p_d}$ is the ith patch.

We set the label of a brain atlas patch, as follows,

$$l_i = mode(a_i),$$

where mode is the most recurring element in a vector, a_i is the set of voxels for ith brain atlas patch.

After the image is partitioned into patches at each time-point, the resulting design matrix $\bar{X} \in \mathbb{R}^{(T,P)}$ is obtained, where T is the number of time-points and P is the

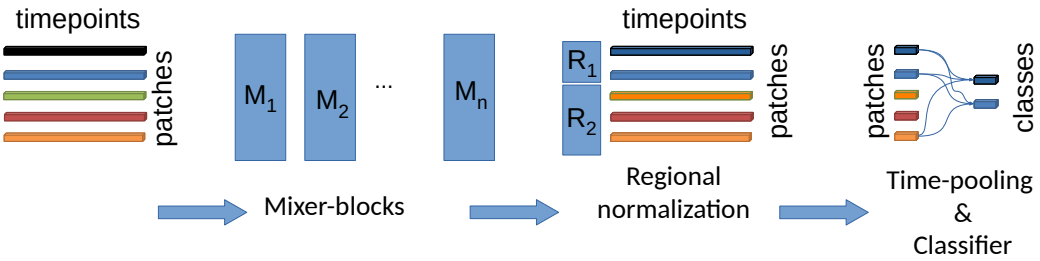


Figure 3.5: Ilustration of the proposed Structured MLP architecture for the brain decoding problem.

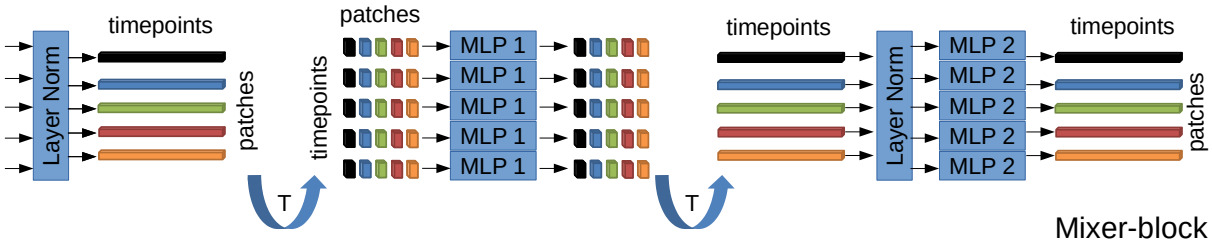


Figure 3.6: MLP mixer block is used to introduce non-linearity.

number of patches.

Once the design matrix is constructed, it is fed to a series of Mixer blocks. In the following, we firstly explain the mixer block, illustrated in 3.6. Secondly, we define the regional normalization method proposed in this work.

Mixer Block Mixer Block is proposed in Tolstikhin et al., 2021 and illustrated in 3.5. The MLP blocks apply layer normalization (Ba et al., 2016) of per-batch MLP features. Layer normalization is defined as follows,

$$y_i = \gamma \hat{x}_i + \beta, \quad \hat{x}_{ij} = \frac{x_{ij} - \mu_i}{\sqrt{\sigma_i^2 + \epsilon}}, \quad \mu_i = \frac{1}{m}\sum_{j=1}^{m} x_{ij}, \quad \sigma_i^2 = \frac{1}{m}\sum_{j=1}^{m}(x_{ij} - \mu_i)$$

where x_{ij} is the γ and β are the scale and bias parameters, which are learned from data.

In our model, the time dimension in a session is introduced to the neural network on the channel dimension. Let M_k be the kth mixer block and $Y = M_k(X)$ be the output of the block M_k. There are two steps in block, M_k.

Step 1: Time mixing step is defined as follows,

$$U_{*,i} = X_{*,i} + W_2\mathcal{V}(W_1 Layernorm(X)_{*,i}), \quad i = 1...T,$$

where $\mathcal{V}$ is the activation function, $X_{*,i}$ denotes column i of the design matrix X, W_1

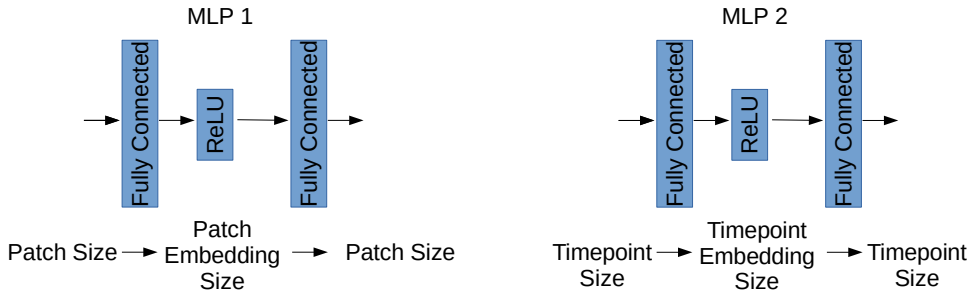


Figure 3.7: Patch and timepoint embedding in multi-layer perceptrons.

and W_2 are MLP layer weights, $\mathcal{V}$ is the activation function and $Layernorm$ is the normalization function. This step illustrated as "MLP 1" in figure 3.7.

Step 2: Patch (token) mixing step is defined as follows,

$$Y_{j,*} = X_{j,*} + W_4\mathcal{V}(W_3 Layernorm(X)_{j,*}), \quad j = 1...P,$$

where $\mathcal{V}$ is the activation function, $X_{*,i}$ denotes row i of the design matrix X, W_3 and W_4 are MLP layer weights, σ is the activation functions and layernorm is the normalization function. This step illustrated as "MLP 2" in figure 3.7.

At both steps, the MLP is applied as a residual operation, called a skip connection.

Regional normalization, proposed in this work, applies layer normalization separately to each brain region, as follows,

$$y_{R_i} = Layernorm(P_{R_i}), \quad x \in \mathbb{R}^{P\times T} \quad P \in R_1, R_2, ..., R_N.$$

where we define a separate layer normalization step for each set of patches that belong to the same brain region R_i.

The pooling block in the proposed model firstly marginalizes out the time dimension, followed by the pooling of the spatial dimension.

3.3 Experimental Results

In this section, first, we explain Human Connectome Project (Van Essen et al., 2013) dataset properties and the Automated Anatomical Labeling (Tzourio-Mazoyer et al., 2002) brain atlas. Then, we report the results of the reproduced method (Wang, 2020). Finally, Structured MLP experiment details and outcomes are shown. The models are compared in test set performance and convergence time in the experiments.

3.3.1 Human Connectome Project Dataset

Human connectome project (HCP) provides a publicly available and open dataset with over 300TB of static and dynamic data. We have used this data in two ways; the minimally preprocessed 4D task data and 3D contrast maps of task data. In this dataset, there are 1200 subjects, each with the resting-state and the task specific recordings.

The main motivation behind choosing the cognitive tasks are three-fold. Firstly, the selected tasks are repeatable and across subject variation is relatively low. Secondly, the selected tasks are complementary in the evoked brain activations. Thirdly, the activation patterns in the selected tasks cover a relatively wide range of voxels. Further details of the task paradigms and the data acquisition details are reported in Barch, 2013.

There are 7 tasks in the HCP task-fMRI experiments. The subtask at each time point in the experiment duration is listed in the HCP event documents (tsv files). In table 3.2, the tasks, the number of scans of each task, the duration of the experiment for each task and the related subtask of each task are listed.

In the following, we give a brief summary of each task.

In the **working memory** (WM) task, the subject is shown a series of pictures sequentially. There are four types of pictures; place, tool, face and body part types. For each picture, the subject answers one of the two predefined questions. The first question is whether the current picture is of same type as the two-previous picture (2-back). The second question is whether the current picture is of same type as the first picture in

the series (0-back).

In the **gambling** task, the subject plays a card game versus a computer program. In the card game, card numbers range in $[0-9]$. The subject bids an amount of money and guesses whether the computer picked a card that is more or less than 5. The program determines the card to evoke a reward or a punishment effect, depending on the bid and guess of the subject.

In the **motor** task, the subject is asked to do one of the following; tap left or right fingers, squeeze left or right toes or move their tongue.

In the **language** processing task, there are two subtasks; story and math. In the story subtask, the subject listens a short story and answers a two-choice question about the topic of the story. In the math subtask, the subject listens to questions that demand basic mathematical operations; addition and subtraction of two numbers. The subject answers the mathematical operation question by selecting one of the two given choices, with a button push.

In the **social** cognition task, the subject watches a video clip of moving objects of three shapes; square, circle and triangle. The subject answers whether there is a connection between the shapes and the object movements in the video clip. The positive answer, that a connection exists between the movement and the shape of the object, is referred as the theory of mind (TOM) subtask. The negative answer, that no obvious connection exists, is referred as random subtask.

In the **relational** task, there are two subtasks; relational and control. In the relational subtask, the subject is introduced objects on a screen. There are two objects on the top side of the screen and two objects on the bottom side of the screen. The objects differ in only one of the two attributes; the shape or the texture. The subject answers whether the difference (shape or texture) between the top side object attributes is the same between the bottom side object attributes. In the control subtask, two objects are on the top side of the screen, one object is on the bottom side of the screen, and an attribute name text is in the middle of the screen. The subject looks at the attribute name in the middle of the screen and decides whether one of the top side objects share that attribute with the bottom object.

Table 3.2: HCP dataset properties.

	Task	# scans	Duration	Subtasks
1.	Working Memory	405	5:01	0-back, 2-back
2.	Motor	284	3:34	Left foot, Right foot, Left hand, Right hand, Tongue
3.	Emotion	176	2:16	Fear, Neutral
4.	Gambling	253	3:12	Reward, Punish
5.	Language	316	3:57	Math, Story
6.	Social	274	3:27	Theory of mind, Random
7.	Relational	232	2:56	Relation, Control

The training procedure depends on long file-read operations, which requires a fast storage device. We limit the number of subjects to fit the 1 TB fast storage device. We chose the samples which have both left to right and right to left phase encoding runs for all 7 tasks in table 3.2. We break down each session into time intervals of subtasks to optimize file I/O during training.

3.3.2 Automated Anatomical Labeling Brain Atlas

Automated Anatomical Labeling (AAL) Tzourio-Mazoyer et al., 2002 brain atlas is a hand labeled single-subject brain fMRI volume in high resolution. AAL is used as a reference for brain fMRI images, such that the fMRI image is projected onto this volume and the coordinates of the projected image are labeled according to the readily annotated volume labels. There are 116 regions in AAL brain atlas.

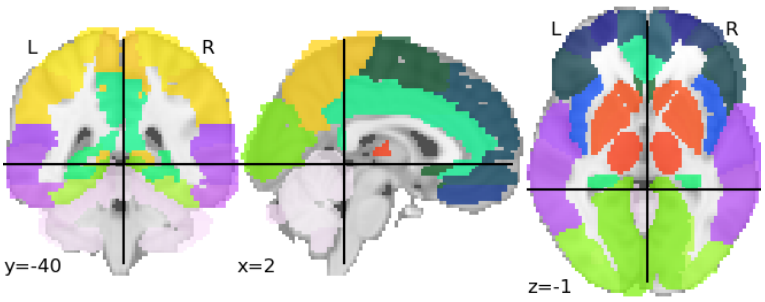


Figure 3.8: Automated anatomical labeling atlas. Each color shows a different anatomical region.

In the following subsection, we give the results of the reproduced 3D convolutional model, that is trained on a source dataset. The pre-trained model is fine-tuned in the transfer learning experiments. Then, we show the results for the two transfer learning experiments on two different target datasets, namely datasets of Motor and Working Memory subtasks. Finally, we show the superior convergence performance of the propose Structured MLP model.

3.3.3 Results of the Reproduced 3D Convolutional Model

In this subsection, we reproduce the results of the 3D convolutional model (Wang, 2020). The 3D convolutional model sequentially applies one dimensional temporal pooling and three dimensional spatial pooling. The temporal pooling is applied on the time dimension of the four dimensional fMRI data. The proposed 3D convolution model accepts a constant number of samples in the time dimension. In the training phase, 27 consecutive frames are sampled from the session interval as the input. In the testing phase, always the first 27 samples are used, so that the testing criterion is the same on all epochs.

The spatial pooling is applied on a three dimensional volume. We observed that 40% of the voxels in the 3D volume ($91 \times 109 \times 91$ voxels) is the empty/non-relevant space outside the brain volume. We discard the empty voxels at each border, and select the minimal cubic volume, resulting in $75 \times 93 \times 81$ voxels.

Our goal is to distinguish between the subtasks (conditions) of the target dataset.

Table 3.3: The table of subtasks that form the source and target datasets. model is trained on a dataset of 7 subtasks, one from each task, as listed on the source column. There are two transfer learning experiments. The first experiment aims to distinguish subtasks of the working memory task, listed on the target 1 column. The second experiment aims to distinguish subtasks of the motor task, listed on the target 2 column.

	Task	Source	Target 1	Target 2
1.	Working Memory	2-back places	0-back body, 2-back body	
2.	Motor	Right hand		Left foot, Right foot, Left hand, Tongue
3.	Emotion	Fear		
4.	Gambling	Loss		
5.	Language	Story		
6.	Social	Mental		
7.	Relational	Relation		

There are two transfer learning experiments; on working memory task, and on motor task. We list the subtasks of each task that are included in the source dataset and the two target datasets in table 3.3.

The main training routine distinguishes between the seven subtasks, shown in the source column of the table 3.3. There is a single subtask from each task of the HCP task-fMRI dataset in the source dataset. Human Connectome Project provides a secondary dataset, called test-retest, along with the main dataset, that repeats the same experiments in the HCP task-fMRI dataset on a new set of subjects. In the test-retest dataset, there are no intersecting subjects with the main dataset.

In the transfer learning experiments, we utilize a pre-trained model on the source dataset, given in the table 3.3. The pre-trained convolutional layer weights do not get

updated in the backpropagation algorithm. The remaining fully connected layers are re-trained on the target dataset, referred as fine-tuning (Yosinski et al., 2014). Note that target dataset has 60 subjects to demonstrate the transfer learning between a large source dataset and a small target dataset.

The number of samples for each of the subtasks that are listed in table 3.3 are as follows.

- For 60 subjects of the target sets,
  - 0-back body:120, 2-back body:119
  - left foot:239, left hand:239, right foot:239, tongue:240.
- For 1095 subjects of the source set,
  - 2-back places:2164, fear:5858, loss:4338, mental:5251, present-story:8372, relation:6241, right hand:4324.

Performance of a classifier is measured by the four cases, true positives, true negatives, false positives and false negatives. A true positive (tp) is a hit, positive prediction for positive sample, in classification context. A false negative (fn) is a miss, a negative prediction for a positive sample. A false positive is the type-1 error, a false alarm, a positive prediction for a negative sample. A false positive (fp) is a positive prediction for a negative sample. We report precision, recall and f1-score, which are defined based on these four cases; tp, fp, tn and fn. Precision is the ratio of true positives (tp) to true positive and false positives

$$precision = \frac{tp}{tp + fp}.$$

Recall is the ratio of true positives to the sum of true positives and false negatives,

$$recall \frac{tp}{tp + fn}.$$

F1-score is the harmonic mean of precision and recall;

$$F_1 = 2 \times \frac{2 \times tp}{2tp + fp + fn}.$$

A confusion matrix lists how many times a class $i \in (1..C)$ is classified as class $j \in (1..C)$, where C is the number of classes. In the confusion matrix results, the

Table 3.4: The performances of the source dataset training phase on the representative subtask of each seven tasks. Each result shows Mean(Std.) over repeated runs.

Task(Subtask)	Precision	Recall	F1-score
Gambling (loss)	0.89(0.03)	0.91(0.02)	0.90(0.01)
WM (2bk-places)	0.96(0.02)	0.93(0.02)	0.94(0.02)
Motor (right-hand)	0.99(0.00)	0.99(0.02)	0.99(0.00)
Social (mental)	0.98(0.01)	0.95(0.02)	0.97(0.01)
Relational (relational)	0.93(0.02)	0.96(0.02)	0.94(0.02)
Emotion (fear)	0.98(0.01)	0.97(0.02)	0.97(0.01)
Language (present-story)	0.98(0.02)	0.98(0.02)	0.98(0.01)

confusion matrix figures show the number of times the row header is classified as the column header. Hence, the diagonal cells show the correctly classified cases and non-diagonal cells show the incorrectly classified cases. We used two color maps for the correctly and incorrectly classified cases. The correctly classified samples lie on the diagonal of the confusion matrix and the diagonal cells follow a blue color map. The incorrectly classified samples on the non-diagonal cells are assigned a red color map.

The hardware environment includes an 8-core processor, a graphics processor with 8GB ram, 48GB system ram and solid state disk storage with 1TB disk space. The software environment is Nibabel for file reading operations, PyTorch for model training and Nilearn for visualization purposes.

We selected the subjects, who has samples for all of the seven tasks. One subtask-per-task data takes 1.2 TB storage space. We further limit the number of subjects to 823 that fits the 1TB fast access disk type. We split the task data file into smaller files of subtasks, for instance an emotion task fMRI file is split into fear and neutral subtask files.

We used ADAM optimizer with empirically selected learning rate $lr = 000.1$, $\beta = 0.9, 0.999$ and weight decay 10^{-5}. The learning rate is reduced on plateau with a factor of 0.1 after 3 consecutive epochs of non-decreasing validation loss. We spare

Table 3.5: All → WM subtasks transfer learning results.

Condition	Precision	Recall	F1-score
0bk-body	0.92(0.03)	0.88(0.05)	0.90(0.02)
2bk-body	0.89(0.04)	0.92(0.04)	0.91(0.02)

43 subjects to simulate the low sample size target dataset.

The training procedure is validated by shuffling the subjects randomly in each run of the experiment. Table 3.4 shows the results for the one-subtask-per-task training routine.

In our experiments, we observed that gambling and relational tasks are the least distinguishable among others. The reproduced results are 3% lower than the reported performance. In the original article, mean(std) accuracy is 93.7% (±1.9%). The missing 300 subjects may have a role in the performance of our implementation, due to our hardware limitations. We observed that gambling task is the worst performing task, which is also in accordance with the MVPA literature (Onal et al., 2017).

3.3.3.1 Working Memory Task Transfer Learning Experiment Results

In the first transfer learning experiment, 2-back and 0-back body subtasks of the Working Memory task, in the Target 1 column of the table 3.3, are classified. In the literature, the working memory target subtasks are known to evoke a distributed activity response Barch, 2013, which makes it hard to apply traditional MVPA methods.

The Motor and Working Memory experiments are run on a limited set of 60 subjects. We show the effect of fine-tuned models on the Working Memory target task performance. The 3D convolutional model is pretrained on the source dataset and fine-tuned to the Working Memory target dataset. We report the mean and standard deviation on 10 runs of the experiment. The 27 time-point session chunks of a subject is split into training, validation and test sets.

Precision, Recall and F1-score for each subtask are listed in 3.5. We observed the confusion matrix in figure 3.9. The model reached test set accuracy of 0.90%(±0.02%).

3.3.3.2 Motor Task Transfer Learning Experiment Results

Motor related subtasks include distinguishing between right hand, right foot, left hand, left foot and tongue movement induced brain signal recordings. The Motor target subtasks, in the Target 2 column of the table 3.3, have a local activity response inside the motor cortex brain region Barch, 2013.

The Motor target task experiments are run on the same number of subjects and fine-tuning method for transfer learning, as in the Working Memory transfer learning experiment. The 3D convolutional model is pretrained on the source dataset and fine-tuned to the Working Memory target dataset.

Precision, Recall and F1-score for each subtask are listed in table 3.6. We observed the confusion matrix in figure 3.10. The test set accuracy of the model is 0.85%(±0.02%).

The highest confusion is observed between the subtasks of the left and the right foot samples, which is also observed in the original work.

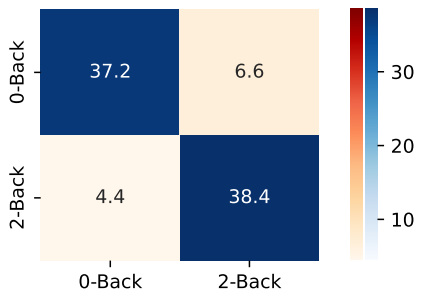


Figure 3.9: 3D convolutional model confusion matrix for the Working Memory subtask. Each cell shows the average number of samples.

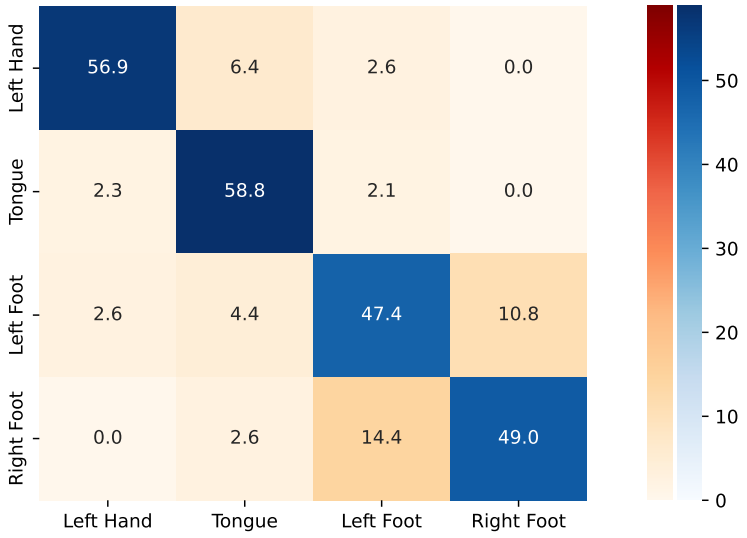


Figure 3.10: Wang, 2020 confusion matrix for the Motor subtasks.

3.3.4 Structured MLP Solution

In Structured MLP implementation, we used ADAM optimizer (Kingma and Ba, 2015) with 0.0001 learning rate. We multiply the learning rate with 0.1 every three epochs when the loss reaches a plateau. For implementation convenience, we discard the empty fMRI image borders to obtain the $90 \times 90 \times 90$ volume of voxels. We use batches of 3 samples, where each sample is a four dimensional matrix with 27 time-points. We split the subject samples in 0.7 training, 0.1 validation, 0.2 testing ratio, and randomize the samples as a stochastic validation. There are 60 subjects in the experiment.

Table 3.6: All $\rightarrow$ Motor subtasks transfer learning results

Condition	Precision	Recall	F1-score
Left Hand	0.95(0.01)	0.90(0.05)	0.93(0.03)
Tongue	0.87(0.03)	0.95(0.01)	0.91(0.02)
Right Foot	0.74(0.02)	0.76(0.03)	0.75(0.01)
Left Foot	0.82(0.01)	0.77(0.04)	0.79(0.03)

Table 3.7: The performances of the suggested Structured MLP on Working Memory subtasks.

Condition	Precision	Recall	F1-score
0-Back	0.85(0.03)	0.83(0.02)	0.84(0.00)
2-Back	0.84(0.01)	0.85(0.04)	0.84(0.01)

3.3.4.1 Working Memory Task Experiment Results

In the working memory task classification, the model distinguishes the conditions "0-Back" and "2-Back", listed in the table 3.3. Precision, recall and f1-score results are shown in the table 3.7. The confusion matrix 3.11 shows the mean values for each cell over random partitioning of the data.

The test set accuracy of the model is 0.84%(±0.01%).

3.3.4.2 Motor Task Experiment Results

In the Motor task classification, there are four subtasks, namely, "Left Hand", "Tongue", "Right Foot" and "Left Foot", shown in table 3.3. The accuracy score for the Motor task classification experiment is $83.00(2.16)\%$. Table 3.8 lists the precision, recall and f1-score results.

Figure 3.13 shows the "Right Foot" and "Left Foot" classes have a lower classification result, compared to other tuples of classes. This result is in line with the finding in

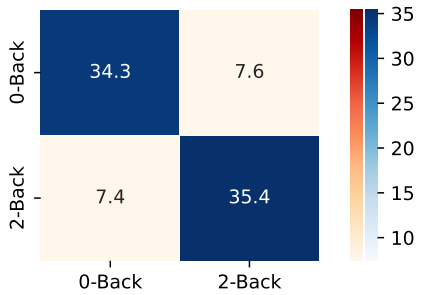


Figure 3.11: Structured MLP Confusion matrix for the Working Memory subtasks.

Table 3.8: The performances of the suggested Structured MLP on Motor subtasks

Condition	precision	recall	f1-score
Left Hand	0.92(0.01)	0.88(0.06)	0.90(0.04)
Tongue	0.87(0.05)	0.94(0.01)	0.90(0.03)
Right Foot	0.74(0.02)	0.68(0.09)	0.70(0.05)
Left Foot	0.76(0.04)	0.79(0.06)	0.77(0.03)

Wang, 2020.

3.3.5 Convergence Results

As expected, we observed that the representation learned from seven task data dramatically speeds up convergence on subtask learning via fine-tuning compared to training from scratch on the subtask, seen in figure 3.12.

This is consistent for both the working memory and motor subtasks. It takes 3-6 epochs for the fine-tuned model to converge. After convergence, fine-tuned model also outperformed the one trained from scratch on the subtask data.

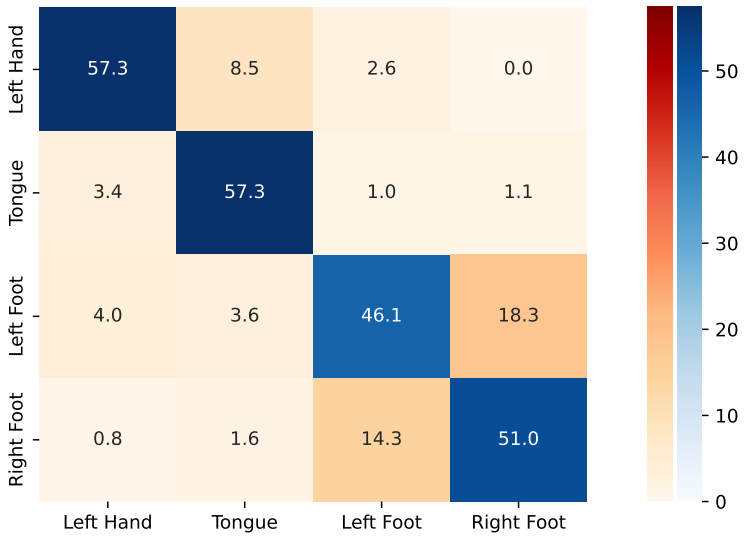


Figure 3.13: Structured MLP Confusion matrix for Motor subtask.

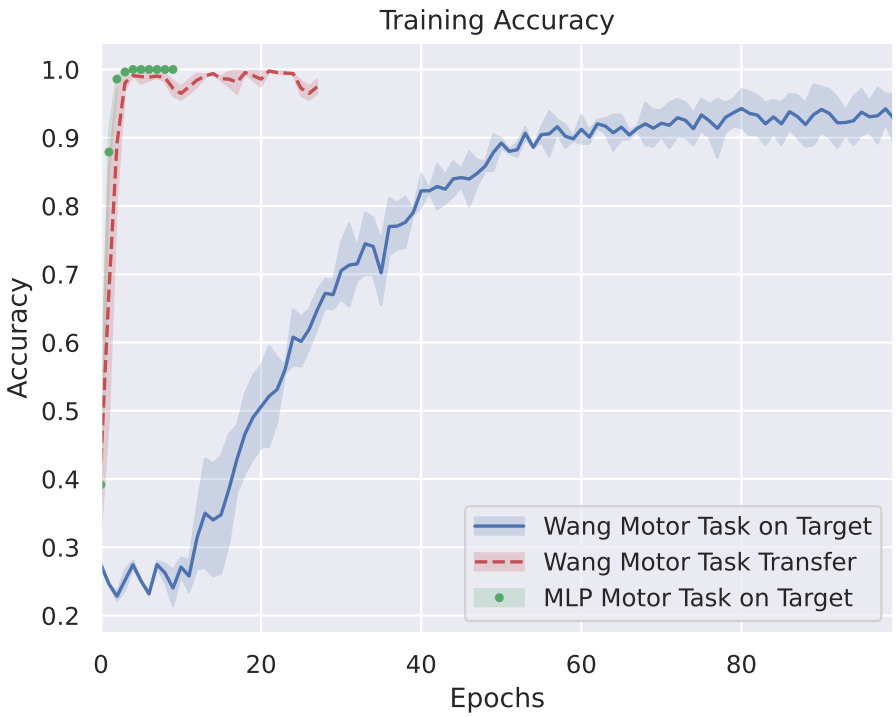


Figure 3.12: Motor task accuracy for 1- 3D convolutional model trained on Motor task target dataset, 2- 3D convolutional model trained on the source dataset with seven tasks and fine-tuned to the Motor task target dataset, 3- Structured MLP trained on Motor task target dataset.

Figures 3.12 and 3.14 shows the training set convergence curves for the three cases.

In the first case, 3D convolutional model is trained only on the target dataset. In the second case, 3d convolutional model is trained on the source dataset and finetuned to the target dataset. In the third case, our model structured MLP is trained only on the target dataset.

In figure 3.12, the green curve shows the convergence of the structured MLP model is very close to the curve of fine-tuned 3D convolutional model. Both the fine-tuned and structured MLP curves are significantly above the target-only trained 3D convolutional model. Note that fine-tuned model requires a large source dataset comprised of one subtask from each of the seven tasks in table 3.3.

In figure 3.14, the green curve shows the structured MLP model slightly slower than the curve of fine-tuned 3D convolutional model. The target-only trained 3D convolu-

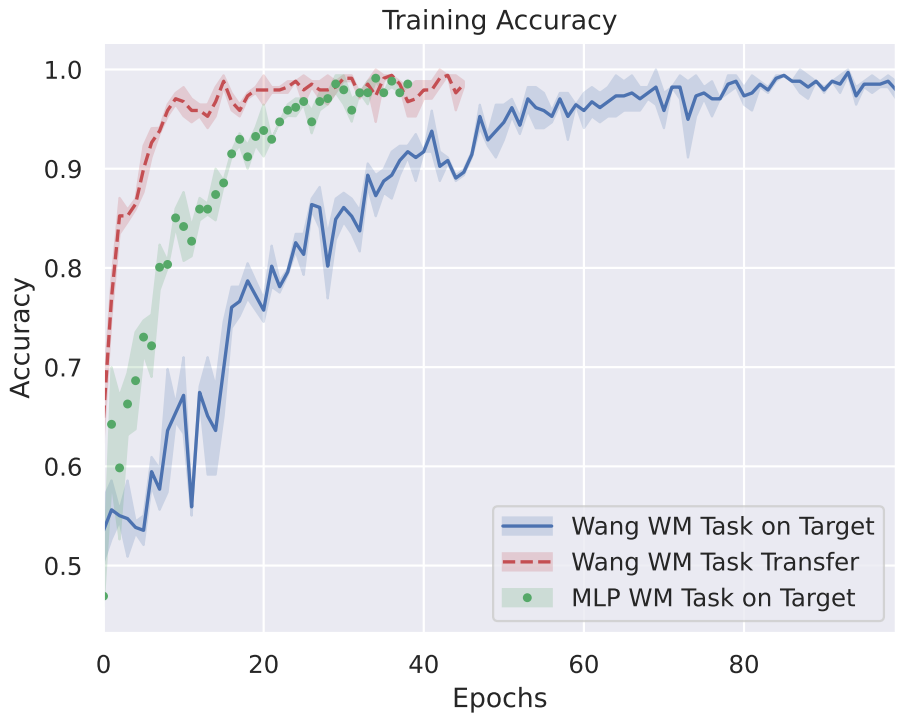


Figure 3.14: Working memory task accuracy for 1- 3D convolutional model trained on Working Memory task target dataset, 2- 3D convolutional model trained on the source dataset with seven tasks and fine-tuned to the Working Memory task target dataset, 3- Structured MLP trained on Motor task target dataset.

tional model convergence is significantly slower than the two other cases.

3.4 Chapter Conclusion

In this chapter, we show that MLP blocks with structured normalization can reduce the convergence time and improve transfer learning runtime performance for time-critical applications of MLP methods for brain decoding. Our work is part of an early research on fine-grained whole brain models, where the literature in brain decoding is applied on either local fine grained data or coarse whole brain data. Due to being an early research, we encountered problems in the implementation of the model and optimization of model parameters under hardware constraints. We observed that the performance of the Structured MLP model is close to the pretrained 3D convolutional model in two transfer learning experiments.

In the proposed MLP-Mixer (Tolstikhin et al., 2021) variant, the encoded feature resolution is equal on successive application of MLP blocks. We utilized the equal resolution over layers to apply normalization over brain regions. However, the equal resolution introduces a big overhead in space requirement of the model, which forced us to use a small batch size. As a future work, MLP-blocks can be applied to a larger neighborhood in the patch lattice graph at successive layers to imitate the decreasing resolution in the convolutional model, that solves the space requirement overhead.

An important contribution of Wang, 2020 is their saliency analysis, where they compare saliency maps of fMRI model with GLM results. The similarity of the learned saliency of samples with their GLM maps potentially has also ties with a recent theoretical work of Hu, 2019 that shows deep models learn increasingly complex features throughout training. We also saw that activation peaks at similar locations in the GLM maps and the gradient maps. However, the saliency extraction method, namely guided backpropagation is known to be highly data dependent (Adebayo et al., 2018).

Deep learning methods are vulnerable to class prediction changes due to small changes in the input, referred as adversarial vulnerability problem, which makes it hard to interpret the model. The inter-session and inter-subject changes in the fMRI samples is a big obstacle in the generalization ability of the suggested non-linear model. In the literature, adversarial attacks are such small perturbations designed to result in the largest change in its representation, that exploits the vulnerability of the deep models. In the brain decoding problem, this is especially important. The brain decoding model is the interpretation of the data for exploratory purposes. In neuroscience, black box models are traditionally avoided due to the interpretability problem. The solution in Wang, 2020 is also vulnerable to small perturbations. One proposed method to reduce adversarial vulnerability is via averaging out the gradients around an input, called SmoothGrad (Smilkov et al., 2017). McClure et al., 2020 works on improving the approach in Wang, 2020 by a method similar to SmoothGrad (Smilkov et al., 2017) that aims to have smoother loss surfaces by training the model further with samples that are injected adversarial noise. However, neither of these solutions are a panacea for black-box models. We investigate a more reliable approach in the second part of the thesis, that is inherently interpretable.

Recent MLP models are resolution sensitive (Liu et al., 2022), where the model does not generalize to a dataset of a different resolution. Resolution sensitivity of a model is problematic in transfer learning tasks where source and target dataset resolutions may differ. This is an open problem in recent MLP-Mixer variants, and a weakness in our model.

Furthermore, we also observed that our MLP variant is hard to stabilize compared to the 3D convolutional model Wang, 2020 and the test results are on par with the reproduced results of the 3D convolutional model Wang, 2020. The major problem in the MLP variant is the inter-subject and inter-session alignment problem. The reported results are affected by inter-session and inter-subject changes. As a future work, subject alignment is an important improvement to our MLP-Mixer variant, as the unsupervised domain adaptation goal in Gao, 2019.

We used Automated Anatomical Labeling map (Tzourio-Mazoyer et al., 2002) as a structural prior information. We propose a flexible model that can accommodate a subject-specific or sample-specific atlas on its pipeline, which further widens applicable prior information types. As in the work of Aydöre et al., 2019, a bank of structural maps, learned from data, is a possible future direction. Our work can also be improved by adopting more recent brain atlases, i.e. Schaefer (Schaefer et al., 2018) brain atlas or MMP (Glasser et al., 2016) brain atlas.

Linear models are preferred for explainability, interpretability, stability and smoothness concerns in neuroscience literature, as proposed in Saxe et al., 2020. The literature on linear models also enable many potential directions in a computationally tractable way. In the next chapter, we propose a transferable feature generation method to improve transfer learning performance in an unsupervised manner, without using target dataset class labels in the calibration process, as opposed to fine-tuning method that requires target dataset class labels.

CHAPTER 4

FEATURE ALIGNMENT FOR SINGLE-TASK TRANSFER LEARNING

Nowadays, there are numerous publications whose fMRI experiment data is hosted on publicly available online repositories, such as OpenNeuro (Markiewicz et al., 2021). Unfortunately, in most of the datasets, the number of subjects is less than 30 (Turner et al., 2018), due to the costly fMRI data acquisition process. Turner et al., 2018 show that the experiment replicability increases with the number of available subjects in an experiment (sample size of an experiment). Hence, the degree of replicability of the limited sample size experiments is relatively low.

Some studies investigate a common cognitive task to prove a hypothesis in an experiment. The articles by Aron et al., 2007 and Xue et al., 2008 follow the common "stop-go" cognitive task, where a subject is asked to repeat an action, until a stop signal is given. Likewise, there is a wide range of publications on naturalistic experiments, such as the audio clip listening or the movie watching experiments in Haxby et al., 2011.

As explained in section 2.3, fMRI data is collected from a subject in multiple sessions. The dataset acquired in an experiment is comprised of multiple subjects and sessions.

Each fMRI sample, recorded in a session of a cognitive experiment is represented by a vector of voxel activations in the activation space, where the dimension is the number of voxels in a brain volume at each time instant. The session vectors of a single subject is assumed to be closer in the activation space compared to vectors from different subjects. Furthermore, the session vectors are assumed to be further apart for two subjects from that of two different datasets, compared to two subjects from a single dataset. This observation allows us to form a three level hierarchy of

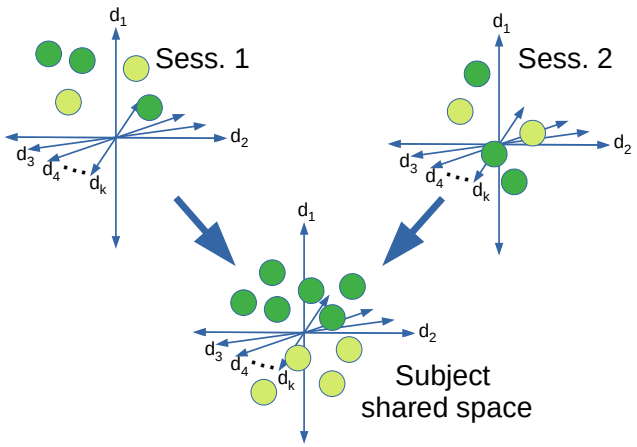


Figure 4.1: Feature alignment on samples from two sessions of the same subject. Each point is a sample recorded in the session, colored with the class label. The $d_{1..k}$ are the spatial dimensions. The feature alignment increases the classification performance compared to directly aggregating the session samples.

variability among samples of fMRI datasets. At the first level, data variations among the sessions of the same subject are the lowest. At the second level, the data variations are relatively higher among the subjects which come from the same dataset. At the third level, the data variations are the highest among the different datasets.

Each cognitive task evokes a different pattern of voxel activations. Thus, datasets with a common cognitive task evoke a common pattern of voxel activations. However, the common pattern of voxel activations drift between sessions, subjects and datasets in an increasing level, as defined in the hierarchy of data variability. In the context of the this chapter, we refer reducing the drift between session samples as **feature alignment**. Figure 4.1 illustrates the alignment of samples from two sessions of the same subject, where the aligned session samples of the same subject form the subject specific shared space. The color of the vectors indicate the different class of the cognitive states in the figure. The aggregation of samples with feature alignment allows discarding the subject specific activations and allows learning a task-specific model, compared to the aggregation of samples without a feature alignment step.

In this chapter, we apply transfer learning between small sample size datasets with a common cognitive task, namely the "stop-go" task. Considering the fact that there

is a hierarchical variability among the samples of inter-sessions, inter-subject and inter-dataset, we suggest a new feature alignment method, called hierarchical feature alignment. Then, we apply transfer learning methods to the aligned features using linear models. This approach increases the generalization problem of artificial neural network models, applied on fMRI dataset, with limited sample sizes.

Transfer learning on limited sample size fMRI data pose a difficult problem stemming from the hierarchy of variability. We shift our efforts from artificial neural network model normalization to reducing experiment induced noise with linear transformation of different sessions, subjects and datasets to a common feature space. We take into consideration the hierarchy of variability and align the data features at each level sequentially, called hierarchical feature alignment, which significantly improves transfer learning performance on the brain decoding problem.

In this chapter, we incorporate a brain template (AAL by Tzourio-Mazoyer et al., 2002) as the common ground to align the session data. Recall that, naturalistic experiments investigate brain signals of subjects while watching a movie clip or listening to an audio-story. Since the clip or audio sample is common for all subjects, at any time-point, the label of the cognitive state is also the same for all subjects. The data with a common label sequence is referred as homogeneously labeled data in the literature. If there is no common label sequence for each data acquisition session, the data is referred as heterogeneously labeled data. Our model does not require each session to have the same label sequence on the time dimension, hence it can be used in a wide range of heterogeneously labeled datasets. We also propose two hierarchical feature alignment methods; namely hierarchical group principal component analysis (H-PCA) and its variant that integrates the label information, supervised hierarchical group principal component analysis (SH-PCA). We follow the common benchmark proposed in Zhou et al., 2018. Our suggested methods, H-PCA and SH-PCA outperforms the state of the art approach in Yousefnezhad et al., 2020.

This chapter is organized in 5 sections. In the literature survey section, we explain the related work in both neuroscience literature and, in general, pattern analysis/machine learning literature. In section 4.2, we give the derivation of generalized canonical correlation analysis (GCCA) method, which is the basis of this chapter. In section 4.3,

we propose a GCCA based transfer learning approach with a brain atlas as the common ground between session samples. In section 4.3, we define a modified projection function that improves the alignment of session data. In section 4.4, we explain a preliminary analysis and show our experimental results. In section 4.5, we summarize our work and give future research directions.

4.1 Literature Survey on Feature Alignment for Brain Decoding

The background of transfer learning methods used in neuroscience literature can be traced back to principal component analysis (PCA, Pearson, 1901) and canonical correlation analysis (CCA, Hotelling, 1936).

Krzanowski, 1979 proposes a PCA-based method to find a common representation among multiple datasets in reduced dimensions, where PCA is applied on combined covariance matrices of data groups. Multiview learning involves alignment of different views of data to find a common variation. Canonical correlation analysis (CCA, Hotelling, 1936) maximizes the correlation between two data groups by applying a different linear transformation on each data group. Generalized canonical correlation analysis (GCCA, Kettenring, 1971) extend CCA to more than two sets. In the literature, there are different formulations for GCCA, such as sum of correlations (sumcor) and maximum variation (maxvar). Akaho, 2007 apply kernel method to CCA. Multiview learning is also a popular approach in transfer learning, where multi-modal or multi-source data can be used as views of the same target concept, as suggested in P.-h. Chen, 2017.

In Neuroscience literature, hyperalignment (Haxby et al., 2011) aligns subject pairs with Procrustean transformation (Schönemann and Carroll, 1970) to reach a common coordinate space over all pairs.

4.1.1 Hyperalignment and Transfer Learning in Neuroscience

Hyperalignment is a feature alignment method to reduce inter-subject variability across the subjects of an fMRI dataset.

A popular hyperalignment method, called Shared Response Model (SRM, P. Chen et al., 2015), decomposes a session matrix of size $T \times N$ into two matrices of size $T \times k$ and $k \times N$, where T is the number of timepoints, N is the number of topological points and k ($k < T$, $k < N$) is the reduced dimension size. SRM is proposed on naturalistic paradigm datasets, specifically movie watching, where the $T \times k$ matrix is assumed to be common among all session samples. The common $T \times k$ matrix is used to aggregate multi-subject fMRI samples, as well as subjects from different datasets with the same movie watching task. There is a wide range of variants for SRM. Examples include the local SRM based on searchlight (H. Zhang et al., 2016), autoencoder SRM (P. Chen et al., 2016) and SRM for heterogeneous data (Nastase et al., 2020).

SRM variants are searchlight SRM (H. Zhang et al., 2016), autoencoder SRM (P. Chen et al., 2016) and SRM for heterogeneous data (Nastase et al., 2020).

H. Zhang et al., 2018 apply SRM as a transfer learning method for the brain decoding problem. They enforce one subject to be part of both a source dataset and a target dataset, where both datasets are acquired under the same naturalistic paradigm cognitive tasks. The shared $k \times N$ subject specific topological matrix is treated as a link between the source and target datasets. Then, a linear transformation is estimated to align the common subject topological matrices between the two datasets. Learned linear transformation is applied to session data of all other subjects in the target dataset, as a means to align the source and target datasets.

Recently, Shared Space Transfer Learning method (SSTL), suggested by Yousefnezhad et al., 2020, applied an hierarchical feature alignment layer that jointly decorrelates session, subject and dataset views of the data via the solution of two sequential eigenproblems, borrowing the approach in Rastogi et al., 2015. In SSTL, the joint feature alignment applies GCCA followed by PCA. In another recent work by Karakasis et al., 2022, a two level GCCA method is proposed for the joint decorrelation of subject and dataset views of the data.

Representation of session data via a network of brain regions is a common task in neuroscience. Brain region networks are studied in the following publications in the literature on transfer learning for brain decoding. Nastase et al., 2020 increases inter-

subject correlation via a brain template. A hyperalignment technique is also suggested by Rustamov and Guibas, 2016 and Yousefnezhad and Zhang, 2017 utilizing a brain template.

Brain template based hyperalignment is realized in Rustamov and Guibas, 2016 and Yousefnezhad and Zhang, 2017.

Proposed models in P. Chen et al., 2015 and Yousefnezhad et al., 2020 are limited to homogeneous data. Additionally, SRM originally utilizes a subject that exists in both the source and target datasets as an additional anchor point.

The dimension reduction is an important operation to discard irrelevant variation embedded in fMRI data. In the next subsection, we survey the literature on utilizing class label information in projection of the input data to a relatively lower dimensional subspace.

4.1.2 Supervision in Dimension Reduced Representation

The dimension reduced representation methods, PCA and CCA, optimize an unsupervised objective function that disregards class labels.

In the brain decoding domain, Yousefnezhad et al., 2021 propose supervised hyperalignment for multiple subjects in a dataset using fisher discriminant analysis method.

The Supervised PCA method proposed in Barshan et al., 2011 employs an empirical measure of the Hilbert-Schmidt independence criterion (Gretton et al., 2005) to impose dependence between the kernel matrix of samples and the kernel matrix of labels.

4.1.3 Variational Autoencoder (VAE)

In the experimentation section of this chapter, we compare our proposed methods to the variational autoencoder (VAE, Kingma and Welling, 2014), which are briefly explained below.

Table 4.1: Chapter 4 - Hierarchical Feature Alignment notation

Variable	Explanation
$N = (W \times H \times D)$	Number of voxels
T	Number of timepoints
$X_N \in \mathbb{N}^{T \times N}$	Spatio-temporal data matrix
$X_R \in \mathbb{R}^{T \times N}$	Region averaged time-series data matrix
$X^{ROI} \in \mathbb{R}^{T \times Z}$	The set of Z voxel time-series in a region of interest
$A \in \mathbb{Z}^{W,H,D}$	Brain atlas A is a matrix of integer labels.
$U, S, V = SVD(.)$	Singular left and right vectors, U, V and singular values S
SVD_k	Truncated singular value decomposition of k dimensions
X_k	Rank-k approximation of matrix X via SVD
$cov(.)$	Covariance matrix
$proj(.)$	Projection matrix
$\bar{X}_r \in \mathbb{R}^{T_r \times k}$	k dimensional representation of data from session r
C_r	Covariance matrix of data from session r
$G_{s,d}$	Transformation matrix of subject s in dataset d
$tr(.)$	Trace operator
$\|\|.\|\|_F$	Frobenius Norm
I	Identity matrix
W_d	Transformation matrix for dataset d
D_{final}	Final set of transformed sessions
I	Identity matrix
$KL(P\|\|Q)$	Kullback-Leibler divergence of dist. P from dist. Q
$\mathcal{L}$	Loss value in VAE formulation
β	Hyperparameter in VAE formulation

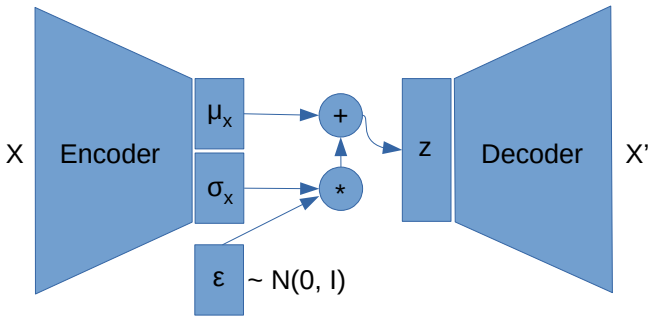


Figure 4.2: Illustration of the Variational Autoencoder (Kingma and Welling, 2014). The encoder estimates a Gaussian distribution with $\mathcal{N}(\mu_x, \sigma_x)$ iteratively. The ranom variable ϵ is sampled from a standard Gaussian distribution. The latent representation $z = \epsilon \times \sigma_x + \mu_x$ is a sample from the encoder Gaussian distribution. Decoder reconstructs X as X' on input vector z.

VAE is illustrated in figure 4.2, where input sample X is reconstructed as X', ϵ is sampled from $\mathbb{N}(0, I)$, z is the latent representation of the sample X,

$$z = \epsilon \times \sigma_x + \mu_x.$$

The VAE loss function is defined as follows,

$$\mathcal{L} = ||X - X'|| + KL(q(z|x)||\mathcal{N}(0, I)), \tag{4.1}$$

where KL denotes the Kullback-Leibler divergence, X is the input sample, X' is the reconstructed sample, q is the encoder distribution. Encoder network estimates μ_x and σ_x from data. VAE is trained via backpropagation algorithm, which requires each block in the network to be differentiable. The non-differentiable sampling operation is stochastically estimated by sampling ϵ from a standard Gaussian and using (μ_x,σ_x) to scale and offset ϵ, which is called the reparametrization trick.

The β-VAE (Higgins et al., 2017) loss function is defined as follows,

$$\mathcal{L} = ||X - X'|| + \beta KL(q(z|x)||\mathcal{N}(0, I)), \tag{4.2}$$

where the hyperparameter β on the KL term favors orthogonal features.

The σ-VAE (Rybkin et al., 2021) loss function is defined as follows,

$$\mathcal{L} = ||X - X'|| + KL(q(z|x)||\mathcal{N}(0, D)), \tag{4.3}$$

where D is a diagonal covariance matrix that replaces the identity covariance matrix. D is estimated from the empirical covariance matrix of the decoder block of VAE. The diagonal covariance matrix

$$D = (\sigma^*)^2 I,$$

where

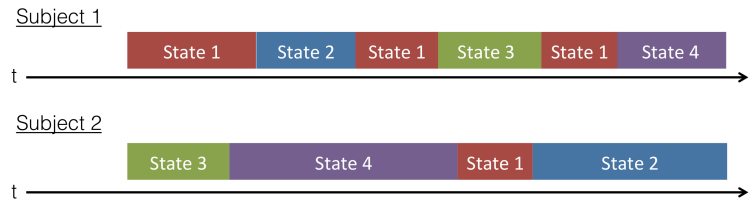
$$(\sigma^*)^2 = \sum_i (x_i - \mu_i)^2.$$

Conditional VAE (Sohn et al., 2015) proposed a method to include side information into the unsupervised VAE training procedure. The side information is simply added to both the input in one-hot form and the latent vector z.

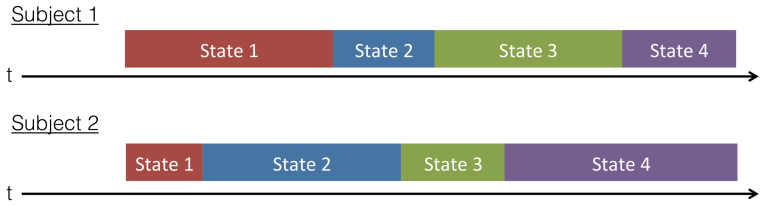
4.1.4 A Critique for Transfer Learning Methods for Brain Decoding

The methods, briefly overviewed in the previous section poses some important problems in brain decoding, as criticized below. These problems inspired us to suggest new feature alignment methods by revising the GCCA method, given in the next section.

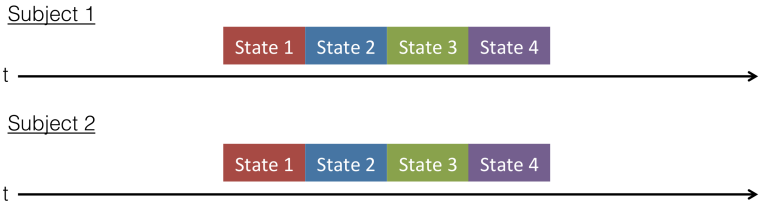
In a heterogeneously labeled dataset, the samples can be arranged to obtain a homogeneously labeled dataset, referred as temporal synchronization. Figure 4.3 illustrates the temporal synchronization steps, proposed by P. Chen et al., 2015. The given time-series data, in figure 4.3-a, do not have a common class label sequence. The following steps show how some of the samples are re-indexed and discarded to obtain a common class label sequence among the two time-series data. Figure 4.3-b illustrates the arrangement such that the cognitive states with the same class label are grouped together by reordering the samples in the session time duration. An equal order of cognitive states is maintained among the samples of each session. In the second step, illustrated in figure 4.3-c, the number of samples for each cognitive state is truncated



(a) Initially, the two time series data have arbitrary class labels (not temporally synchronized).



(b) Each cognitive state is grouped together by re-ordering of samples.



(c) Among the two time-series data, the number of samples are truncated to that cognitive state with the smallest number of samples.

Figure 4.3: Temporal synchronization strategy maintains homogeneity of class labels among the two heterogeneously labeled data groups, proposed in "Shared response model" P. Chen et al., 2015. Illustration from the author's presentation slides.

to match the lowest number of samples in a cognitive state across all subjects. The application by Yousefnezhad et al., 2020 follow this strategy as well, to apply transfer learning on datasets, given by Aron et al., 2007 and Xue et al., 2008.

A common class label order in both the source and target datasets is a vulnerability that needs to be addressed. Westfall et al., 2017 propose time-point randomization tests. Otherwise, the common time dimension can cause inflated or misleading success rates of transfer learning among datasets. Westfall et al., 2017 report that in single site neuroscience experiments, 60% of recent work have vulnerabilities on reports of between subject or between session comparisons.

Temporal synchronization strategy, illustrated in 4.3, has been criticized in reviews of Yousefnezhad et al., 2020 as it may cause to leak information that results in learning the specific ordering of data, which may result in shortcut learning Geirhos et al., 2020.

On the other hand, data with arbitrarily ordered class labels do not require time-point randomization tests.

High dimensional (N=10^6) feature vectors of fMRI samples require a large memory throughout the data processing and decoding steps. Constraining the spatial dimension with a region of interest (ROI), as in Yousefnezhad et al., 2020, still necessitates a large memory requirement (number of dimensions N~10^4). Although the online methods are proposed to manage the memory requirements, these methods are prone to accumulation of errors. Reducing the memory requirement in an fMRI transfer learning task on the high-dimensional fMRI data is usually a neglected problem in the fMRI literature.

Another issue observed in transfer learning methods is the non-linearity that does not guarantee preserving certain properties of the data, such as the covariance of data dimensions. Linear models, like PCA, center and rotate the data around the origin, hence the linear models preserve the covariance of dimensions in the transformed data.

In Neuroscience, it is important to utilize a method that mathematically guarantees to preserve the common property among the data groups. In the feature alignment problem, we define an anchor property of data. Hence, if the common property is inferred from the data, such as the covariance of data dimensions, linear methods become a powerful tool in aligning the data groups for the transfer learning methods for brain decoding.

4.2 Generalized Canonical Correlation Analysis

Generalized canonical correlation analysis (GCCA) is an essential method in the state of the art transfer learning models for brain decoding.

In canonical correlation analysis (CCA), given two groups of data, we aim to maximize the correlation of the two data groups by finding a separate transformation for each group. GCCA generalizes CCA to multiple data groups. In this section, we explain the maximum variance (max-var) formulation of GCCA. Max-var formulation estimates a common space G in objective function with a closed form solution. We briefly explain generalized canonical correlation analysis of Kettenring, 1971. The details of optimization is available in Ghojogh and Crowley, 2019.

Formally speaking, given the session data is represented by X. The projection matrix of X is defined as follows,

$$P = Proj(X) = X(X^T X)^{-1} X^T.$$

Then, minimizing the objective function of generalized canonical correlation is defined as,

$$\min_{G,\{K_j\}_{j=1}^J} \quad \sum_{j=1}^{J} \left\| G - X_j^T K_j \right\|_F^2, \quad s.t.\ G^T G = I, \tag{4.4}$$

where X_j are the observed session data, K_j are a mapping for X_j, G is an orthonormal matrix.

Choosing $K = (X^T X)^{-1} X^T G$, such that $XK = Proj(X)G = PG$, leads to a closed form solution of max-var GCCA.

Our goal is to minimize the following cost function with respect to G, as follows,

$$\begin{aligned}
&\sum_{j=1}^{J} \|G - P_j G\|_F^2 && = \sum_{j=1}^{J} \|(I - P_j)G\|_F^2 \\
&= \sum_{j=1}^{J} tr((G^T (I - P_j)^T (I - P_j) G)) && = \sum_{j=1}^{J} tr((G^T (I^2 + P_j^2 - 2IP_j) G)) \\
&= \sum_{j=1}^{J} tr((G^T (I^2 + P_j^2 - 2IP_j) G)) && = \sum_{j=1}^{J} tr((G^T (I + P_j - 2P_j) G)) \\
&= \sum_{j=1}^{J} tr((G^T (I - P_j) G)).
\end{aligned} \tag{4.5}$$

The above minimization equation is equivalent to the following maximization problem,

$$\max_{G} \sum_{j=1}^{J} tr((G^T(P_j)G)). \tag{4.6}$$

Thus, minimizing the objective function in equation 4.4 is equivalent to maximizing the following objective function,

$$\max_{G} tr(G^T(\sum_{j=1}^{J} P_j)G) \quad s.t. \quad G^TG = I. \tag{4.7}$$

The Lagrangian for equation 4.7 is solved as follows,

$$\begin{aligned} &\mathcal{L} = tr(G^TMG) - tr(\lambda^T(G^TG - I)), \\ &\frac{\partial \mathcal{L}}{\partial G} = 2MG - 2G\lambda, \\ &MG = G\lambda, \end{aligned} \tag{4.8}$$

where $M = \sum_{j=1}^{J} P_j$ and λ are the Lagrange multipliers.

Note that the rows of the orthogonal matrix G are the eigenvectors of matrix $M = \sum_{j=1}^{J} P_j$ corresponding to the eigenvalue λ, where j is the index of each group of data.

4.3 Hierarchical Feature Alignment

When employed to heterogeneously recorded sparse data, the available techniques mostly provide a poor decoding performances in a transfer learning schema.

In this section, we create a shared feature space by extracting the common information, embedded across all sessions and subjects of multiple source datasets to improve the decoding performance of a small size heterogeneous target dataset, based on session covariance matrices. We extended the method suggested by Yousefnezhad et

al., 2020 to heterogeneously labeled datasets for the transfer learning task, where the common dimension among datasets is defined by an anatomical brain atlas. Barshan et al., 2011 inspired our work to minimize a kernelized statistical independence criterion between data and class labels as an intermediate step in our label-guided solution.

4.3.1 Our Contributions

In this work, we use the automated anatomical labeling (AAL) brain atlas to construct the brain region time-series data, which is much lower dimensional ($N \approx 10^2$) than the voxel representation ($N \approx 10^9$) in the state of the art method. This approach eliminates the need for large memory requirements and complicated, error-prone online methods. Furthermore, brain region time-series data is more interpretable, owing to the enforced anatomical coherence by the brain atlas (P. Chen et al., 2016). Additionally, the brain atlas representation removes the requirement of having equal time label sequence and equal session time duration.

We detect low-dimensional common structures in fMRI data of subject groups who perform the same cognitive task via Generalized Canonical Correlation Analysis. The common structure is then used as the anchor point for aligning data.

In this setting, the high-variation directions in the feature space are assumed to be task related. On the contrary, low-variation directions are assumed to be relatively less task related and carry relatively more private and non-transferable information. We propose a method that assigns the optimal importance weight to each direction, that is not covered in the recent literature.

We propose two Hierarchical Feature Alignment methods, that outperform the state-of-the-art method in transfer learning for brain decoding.

Our first method generates aligned features, based on the relation of session covariance matrices, inspired by max-var formulation of generalized canonical correlation analysis.

Our second methods is the class label guided variant of the supervised transfer learn-

ing (H-PCA) method using the Hilbert-Schmidt independence criterion, referred as supervised H-PCA (SH-PCA).We augment generalized canonical correlation analysis with label supervision and utilize this in a transfer learning setting. We apply label supervision such that the data representation is further transformed to reduce the correlation distance between data features and class label distribution. We use this learned transformation as the transferred component of data in the testing phase.

4.3.2 Problem Definition

In the transfer learning for brain decoding problem, there are multiple small scale datasets available for a common task. Each dataset $d \in D$ has S subjects. A neural image in the dataset d is a discrete Spatio-temporal signal, represented by the matrix $X_N \in \mathbb{R}^{NxT}$, where T is the number of brain volumes, each of which is obtained at time instant t_i, for $i = 1, ..., T$, and there is a total of N voxels on each brain volume. The entries of the fMRI data matrix, $X_N = [x_{ij}]$, shows the intensity of voxel v_j at time instance t_i.

A subject, s, is given a stimulus, which forces a change in the cognitive state $c_i \in \{0, 1\}$ at time t_i. The number of time samples, T, and the number of subjects, S, vary across the sessions and datasets.

Instead of using all of the N voxels of a brain volume, we obtain the average time series for each anatomical region and represent our features in the data matrix, $X_R \in \mathbb{R}^{RxT}$, where R is the number of anatomical regions. Hence, the entries of X_R show the average intensity value of all voxels, which reside in the anatomical region, $r = 1, ..., R$, measured at time t_i.

Mathematically speaking, suppose that the time series of voxel, v_j, is represented by a vector, $x_j = [x_{1j}, ..., x_{Tj}]$, as the j^{th} row of the design matrix, X_N. Then, each anatomical region, r, is represented by average voxel intensity values which reside in that anatomical region by the following vector,

$$\mathbf{x_r} = \frac{1}{N_r} \sum_{\forall x_j \in r} x_j, \tag{4.9}$$

where the region averaged fMRI data matrix is represented by $X_R = [\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_R]^T \in \mathbb{R}^{R \times T}$, where R is the number of anatomical regions and T is the number of time samples in a session of a subject. $\hat{\mathbf{x}}_\mathbf{r} \in \mathbb{R}^T$ is the time series which represents the anatomic region r, N_r is the total number of voxels in region r 4.9. For simplicity, we set $X = X_R$.

The length T of the time series can vary among sessions, subjects, and datasets. The brain regions are determined by the Automated Anatomical Labeling (AAL) brain template, which segments N voxels into R anatomical regions.

We train our transfer learning model on source datasets D_S, where $|D_S| \geq 1$, and evaluate it on a target dataset D_T, where $D_S \cap D_T = \emptyset$, $|D_T| = 1$. Our main goal is to train a model M^S on source data and improve the performance of a model M^T on target dataset.

4.3.3 Brain Atlas Aligned GCCA

Recall that in naturalistic paradigm experiments, subjects watch the same video/audio clip and corresponding session data are analyzed for common features related to the clip. The session data are therefore "synchronized" in time by the same clip, introduced as the external stimuli. The synchronized time dimension is utilized as the common ground for transfer learning. However, the main line of work in neuroscience follow task-fMRI paradigm, where the time dimension class labels are arbitrary.

The state of the art method, SSTL (Yousefnezhad et al., 2020), apply generalized canonical correlation analysis on the common time dimension of each data group, that is only applicable to naturalistic paradigm datasets. The state of the art method works on a region of interest which is a set of J voxels, $X^{ROI} \in [x_{w_1,h_1,d_1}, x_{w_2,h_2,d_2}, ..., x_{w_J,h_J,d_J}]$, that are functionally relevant to the cognitive task. The time dimension label sequence $Y = \{y_1, y_2, ..., y_T\}$ is assumed to be equal for each session. Hence, in equation 4.4, the common space matrix is

$$G \in \mathbb{R}^{T \times T}.$$

The dependence on the temporal dimension T does not allow the model to be applied on heterogeneously labeled data. Authors used the temporal alignment to obtain the

common time dimension on all datasets. Refer to subsection 4.1.4 for details on temporal synchronization.

We represent the brain voxels $X \in \mathbb{R}^{W,H,D}$, that has $N = W \times H \times D = 91 \times 109 \times 91 \approx 10^7$ spatial dimensions, with the brain template AAL, that has 116 dimensions. The brain template partitions voxel coordinates spatially, in an anatomically and functionally coherent way. Hence it is a suitable common ground among subjects in a transfer learning setting, in other words, the same brain template applies to all of the subjects in the experiment.

Mathematically, a single fMRI voxel $x(w, h, d) \in \mathbb{R}$ is a BOLD intensity value at coordinate (w, h, d), where $w \in [1, W], h \in [1, H], d \in [1, D]$. A brain template $A \in \mathbb{Z}^{W,H,D}$ is a matrix of brain region labels, and each coordinate (w, h, d) is assigned a label l, such that $A(w, h, d) = l \in \mathbb{Z}$. There are R brain regions, $L = \{l_1, l_2, ..., l_R\}$ in the brain template A. We average the BOLD intensities of the voxels that have the same label l_i to represent the region R_i. We denote a session data as $X_{r,s,d} \in \mathbb{N}^{R \times T}$, where r,s,d are session, subject and dataset indices, R is the number of regions and T is the number of time-points in the session. In a session, each time-point is labeled with a cognitive state label $y \in \{0, 1\}$.

We mitigate the dependency on the time dimension with a brain region representation and using the brain template as the common dimension among session data. In the resulting GCCA formulation, the common space matrix is

$$G \in \mathbb{R}^{R \times R}.$$

Furthermore, each session data have the common spatial dimension R that enables the transfer learning goal across datasets. The dependence on the spatial dimension allow our model to utilize heterogeneously labeled data. We refer the brain atlas aligned SSTL variant as SSTL-V in the experiment section of this chapter.

4.3.4 Hierarchical Feature Alignment

Figure 4.4 shows the block diagram representation of the hierarchical feature alignment method, which consists of feature alignment and transfer learning modules in both training and transfer phases. In the training phase, the algorithm estimates three

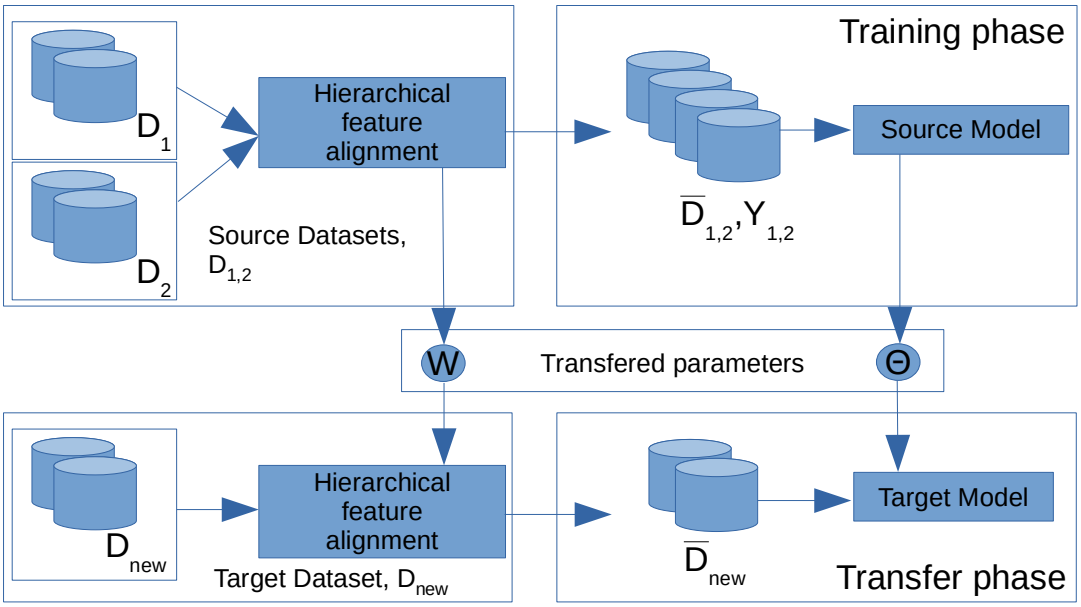


Figure 4.4: Overall view of "Hierarchical feature alignment" framework, modified to work on brain atlas alignment, where global alignment matrix W is transferred to the transfer phase and the classifier weights Θ are used as-is in the evaluation of transfer phase.

transformation matrices to align the data matrices hierarchically across the sessions, subjects and datasets. Then, a classifier is trained with source dataset(s) and tested on a target dataset, which is aligned with the global transformation matrix, obtained in the training phase. In this section, we propose two new methods for the hierarchical feature alignment block in the figure 4.4. We adopt the transfer learning routine in SSTL (Yousefnezhad et al., 2020). The hierarchical feature alignment block is revised by the brain atlas aligned GCCA.

Let us explain the two feature alignment modules, namely, hierarchical-group principal component analysis and class label guided low-dimensional representation in the following subsections.

4.3.4.1 Hierarchical Group Principal Component Analysis

The major assumption of the suggested method is that there exists a common structure of patterns across all sessions, subjects and datasets, induced by the common

cognitive task.

We aim to find a correspondence between lower dimensional representation of each data group in Hierarchical-group principal component analysis, such that transformed data group representations reflect the common pattern.

In order to estimate the common patterns embedded in the noisy fMRI readings, first we reduce the dimension of the feature space by principal component analysis (PCA) via singular value decomposition (SVD). Then, we estimate a transformation matrix at each level of session-subject-dataset hierarchy to generate a shared space, using generalized canonical correlation analysis.

We propose to scale projection method by eigenvalues to improve feature correspondence, matching the extreme points of the spectrum. Scaled projections come up in various fields, where the projection is weighted/scaled by the eigenvalues. Eigenvalue scaling emphasizes the eigenvectors that correspond to a larger portion of the variance and penalizes eigenvectors that correspond to a smaller portion of the variance. For instance, Seo and Kim, 2013 propose to scale subspace projections with eigenvalues for principal component analysis. We implement a similar idea on maxvar-GCCA to expand the eigenvalue distance, that potentially makes it easier to align multiple datasets in finding the subspace of the common variance. Further theoretical background can be found in Hanke and Neumann, 1993, regarding the geometry of scaled projections.

A projection matrix is a form of covariance matrix. Thus, the projection matrix also admits the covariance matrix properties. Our approach can be traced back to the analysis of composite covariance matrices, used in Fukunaga-Koontz transform (FKT), (Fukunaga and Koontz, 1970), where composite covariance matrices of positive and negative class labeled samples are aggregated for feature extraction in a binary classification problem. Koles et al., 1990 uses composite covariance matrices to find the common features among two groups in common spatial pattern analysis.

Singular value decomposition of a matrix X is defined as its decomposition into the orthonormal left and right matrices of singular vectors and a diagonal matrix of singular values.

Principal component analysis is the optimal low dimensional closed form representation, according to Eckart-Young-Mirsky theorem (Eckart and Young, 1936). The theorem states that the lower rank approximation, obtained with principal component analysis via singular value decomposition is the globally optimal estimation. In this respect, our method estimates the best possible feature aggregation in the least squares sense.

Mathematically, a low-rank representation of the data matrix X can be obtained by selecting the top-k rows of U, the top-k diagonals of S and the top-k columns of V as follows,

$$X_k = U_k S_k V_k^T. \tag{4.10}$$

For notational convenience, we define $U = U_k$, $S = S_k$ and $V = V_k$. Recall that principal component analysis decomposes the covariance matrix of a mean-centered data matrix into the scale (eigenvalue) and direction (eigenvector) components. The covariance matrix of X_k is defined as,

$$\begin{aligned} X_k^T X_k &= (VSU^T)(USV^T) \\ &= VS^2V^T \\ A = VS &\rightarrow AA^T = VS^2V^T, \\ X_k^T X_k &= AA^T. \end{aligned} \tag{4.11}$$

The above formulation enables us to represent the covariance matrix in terms of AA^T, where $A = VS$. In this representation, truncated right singular vectors, V and, singular value matrix, S are sufficient to find the covariance matrix.

At this point, we note that maxvar-GCCA (Kettenring, 1971) utilizes a linear projector, referred as proj,

$$
\begin{aligned}
proj(X_k) &= X_k(X_k^T X_k)^{-1} X_k^T, \\
&= USV^T(VSU^T USV^T)^{-1} VSU^T, \\
&= UU^T
\end{aligned} \tag{4.12}
$$

which is a special case of a covariance matrix used for aligning data. Note that "Shared Space Transfer Learning" (SSTL) and brain atlas aligned representation (SSTL-V), in subsection 4.3.3, utilize the standard projection function.

Recall that the maxvar-GCCA finds the eigenvectors of the sum of linear projectors UU^T, using equation 4.12. This task is simply achieved by maximizing equation 4.13 for J groups of data, $[X_k]_j$.

$$
\max_G tr[G^T(\sum_{j=1}^{J} U_j U_j^T)G] \quad s.t. \quad G^T G = I, \tag{4.13}
$$

Equation 4.13 applies principal component analysis on the sum of linear projectors.

Contrary to alignment on temporal dimension with $G \in \mathbb{R}^{T \times k}$ in equation 4.11, we align on the spatial domain where $G \in \mathbb{R}^{R \times k}$.

The scaled projection function is defined as,

$$
proj_{scaled}(X_k) = (US)(US)^T = cov(X_k^T), \tag{4.14}
$$

where the scaled projection leads to the covariance of principal components, US, which is equal to the covariance of brain regions.

In our objective function, we assign a high credit to high variance directions in the feature alignment process with the scaling profile that is based on eigenvalues of the reduced rank data matrix. Starting from the above definitions, we modify the objective function of maxvar-GCCA, which operates on the sum of linear projectors in equation 4.12, such that it operates on the sum of scaled projectors, in equation 4.14. We scale the linear projector by corresponding eigenvalues, which ultimately

corresponds to the covariance of brain regions, defined by principal component US, which is the multiplication of left singular vectors and diagonal matrix of singular values. We find the eigenvectors of the sum of covariance matrices US^2U^T, rather than the sum of linear projectors UU^T, in equation 4.13.

Note that, the sum of scaled projection matrices, $M = \sum_i^N U_i S_i$, is written in the matrix form, as follows,

$$M = [(U_0S_0)(U_1S_1)...(U_NS_N)][(U_0S_0)(U_1S_1)...(U_NS_N)]^T, \tag{4.15}$$

form the matrix in equation 4.15, similar to the idea in Savostyanov, 2014.

Next, we estimate the transformation matrices in the reduced space at the levels of the session, subject and dataset hierarchy with scaled linear projectors.

Mathematically, let the covariance of each session's principal components be represented by

$$C_r = U_{r,s,d} S^2_{r,s,d} U^T_{r,s,d},$$

where the subscripts, r, s, and d indicate session, subject, and dataset indices, respectively, and $X_{r,s,d} = U_{r,s,d} S_{r,s,d} V^T_{r,s,d}$. Then, we can estimate the subject-specific transformation matrix, $G_{s,d}$ that maximizes the variation of the sum of covariance matrices C_r for a single subject s, as follows,

$$\max_{G_{s,d}} tr[G^T_{s,d}(\sum_r C_r)G_{s,d}] \quad s.t. \quad G_{s,d}G^T_{s,d} = I. \tag{4.16}$$

Estimation of the subject-specific transformation matrix, $G_{s,d}$, in equation 4.16 enables us to prioritize the higher variance direction by multiplying the unit direction U and scale S, rather than using only the unit direction U in equation 4.12.

Once the subject-specific transformation matrix $G_{s,d}$ is estimated, the dataset-specific W_d transformation matrix is estimated via linear Karhunen-Loeve (KL) transform, as in Yousefnezhad et al., 2020, as follows,

$$\max_{W_d} tr[(W^T_d(\sum_s G_{s,d})W_d)] \quad s.t. \quad W_dW^T_d = I. \tag{4.17}$$

Finally, we repeat the KL transform of equation 4.17 to estimate the global, inter-dataset matrix W, as follows,

$$\max_{W} tr[(W^T(\sum_{d} W_d)W)] \quad s.t. \quad WW^T = I. \tag{4.18}$$

The estimated matrices, $G_{s,d}$, W_d and W, can be used to align the fMRI data matrix in each dataset, for each subject and session. Hence, the covariance of each sample is represented in the transformed space by D_{final} in 4.19, whose rows are the samples used in the classification step,

$$D_{final} = C_r G_{s,d} W_d W, \tag{4.19}$$

where C_r is the covariance of principal components and $G_{s,d}, W_d, W$ are "session-subject", "subject-dataset" and "dataset-global" mappings, respectively. The final data representation is of size $N \times k$, where N is the number of samples and k is the lower rank number used in SVD.

In the transfer phase, we recalculate the subject-specific transform matrix, $G_{s,d}$ and the dataset transform matrix W_d. However, we use the same global transformation matrix, W in the transfer phase.

4.3.4.2 Label Guided Low-Dimensional Representation

Finally, we apply supervision on the hierarchical group principal component analysis in our experimentation.

In this subsection, we explain the suggested class label guidance method for estimating the common variation structure among data groups. We follow the supervised principal component analysis method, suggested in Barshan et al., 2011 in our hierarchical-group formulation. We apply label guidance using Hilbert-Schmidt independence criterion (HSIC, Gretton et al., 2005) which measures independence between two distributions.

Recall that subject level transformation matrices G_s are found by principal component

analysis, in equation 4.16. We find the label-guided alternative $\bar{G}_s$ via kernel version of supervised principal component analysis. We first explain the direct supervised PCA and then define the kernelized version.

We seek the subspace $\bar{G}_s$ that maximizes the statistical dependence between rotated session principle components, $\bar{G}_s^T C_r$ and label matrix Y, where each row of Y is a one-hot encoded label vector. For this purpose, we use the Hilbert-Schmidt independence criterion, which measures the degree of statistical independence.

An empirical estimate of the Hilbert-Schmidt independence criterion between projection sum, $C_R = \sum_r^R C_r$, and label variable Y is given as follows,

$$HSIC(C_R, Y) = \frac{1}{(n-1)^2} tr(CL), \tag{4.20}$$

where $C = C_R^T C_R$ and $L = Y^T Y$ are mean-centered gram matrices and n is the number of samples.

We find B that maximizes the dependence between C and L, as follows,

$$\max_B tr(B^T C_R L C_R^T B) \quad s.t. \quad B^T B = I, \tag{4.21}$$

where $L = Y^T Y$ and $C = C_R^T C_R$ are mean centered gram matrices.

Solution of equation 4.21 is the set of eigenvectors u for eigendecomposition of $C_R L C_R^T$.

We obtain the kernelized formulation of supervised PCA with variable substitutions in equation 4.21, as follows,

$$\max_{\bar{G}_s} tr(\bar{G}_s^T K_C K_L K_C \bar{G}_s) \quad s.t. \quad \bar{G}_s^T K_C \bar{G}_s = I, \tag{4.22}$$

where we substitute C_R with a kernel $\phi(C_R)$, we set $K_C = \phi(C_R)^T \phi(C_R)$, the optimization parameter B is substituted with $\phi(C_R)\bar{G}_s$ and we optimize for $\bar{G}_s$.

Equation 4.22 is a generalized eigenvalue problem, where $\bar{G}_s$ are generalized eigenvectors of $K_C K_L K_C$. We obtain the label guided variant of our method, referred as supervised hierarchical group PCA (SH-PCA), by using 4.22 in place of equation 4.16.

Algorithm 1 Label guided generalized canonical correlation analysis

Definitions	
r_s, s_d, d	*r:session, s:subject, d:dataset*
X_{r_s,s_d}	*Data from each session*
$P_{r_s,s_d} = X(X^T_{r_s,s_d}X_{r_s,s_d})^{\breve{}1}X^T_{r_s,s_d}$	*Projection matrix*
	implemented via SVD
G_{s_d}	*Subject specific*
$SH-PCA$	*Supervised Hierarchical Group PCA*
SVM	*Support vector machine*
$\hat{X}_{d_s}, \hat{X}_{d_t}$	*Source, target aligned data*
Aligned features	
for each dataset d	* multi-source case
for each subject s_d	
$G_{s_d} = SH-PCA(\sum_{r_s} P_{r_s,s_d})$	G, common proj. subspace over r_s
for each dataset d	
$G = \bigcup G_{s_d}$	Union of projection subspace matrices
$W = eigvec(cov(G-\mu_G))$	W, principal components of G
$\{\hat{X}_{r_s,d} = P_{r_s,d}G_sW\}_{r,s}$	$\hat{X}$, new features
Transfer learning evaluation	
$M_s = SVM(\hat{X}_{d=source})$	Train model M_s on source data
$M_s(\hat{X}_{d=target})$	Evaluate model on unseen target data

4.4 Experimental Results

In this section, we firstly give details about the stop-go paradigm for testing response inhibition, studies carried out under this paradigm, and details of the datasets acquired in related response inhibition experiments. Then, we give the results of our change point analysis, based on ROI data. Finally, on a brain template representation, we show the results of the two proposed methods of hierarchical feature alignment; hierarchical-group principal component analysis (H-PCA) and its supervised variant (SH-PCA).

4.4.1 The Cognitive Paradigm in the Datasets

In our transfer learning experiments, we work on data that is acquired on **stop-signal paradigm** that tests response inhibition. In the Stop-Signal paradigm, a subject repeats an action (initiation) and stops the action immediately on a given signal (inhibition). The common properties of brain signals on response inhibition is tested in this approach.

The neuroscience studies assume that language and motor systems have common mechanisms for response inhibition, where the same regions are involved in non-language motor functions and speech production.

Depending on the cognitive task, different stimuli are designed in Aron et al., 2007 and Xue et al., 2008 and applied for transfer learning in Zhou et al., 2018. We worked on 4 datasets, where the subjects are asked to inhibit the following actions on stimuli;

1. Word: Vocally reading non-necessarily meaningful words,
2. Manual: Pressing a button depending on the letter shown on a screen,
3. Vocal: Vocally reading one letter, (Xue et al., 2008)
4. Signal: pressing a button depending on the arrow direction on screen (Aron et al., 2007).

In these datasets, data acquisition from each subject is in two sessions. The number

Table 4.2: Dataset naming and details. We distinguish successful-unsuccessful stop states.

Dataset	Experiment details			Label distribution	
	Subject #	Session #	Time points	Successful stop	Unsuccessful stop
Word	20	2	[45,51]	705	1046
Manual	20	2	[45,51]	783	993
Vocal	20	2	[45,51]	665	1030
Signal	13	2	[43,50]	821	896

of time-points vary between sessions. The samples belong to one of the two classes; **successful stop** and **unsuccessful stop**, depending on the response time of the subject to the stop signal. Further details can be found on table 4.2.

We experiment on transfer learning between multiple small scale source datasets to a small scale target dataset. We initially used the hosted preprocessed datasets which are based on region-of-interest (ROI) voxels. We improved the data preprocessing by excluding the irrelevant time points, labeled as "junk" or "go". We used a whole-brain parcellation, called Automated Anatomical Labeling (AAL, Tzourio-Mazoyer et al., 2002).

The datasets are publicly available on OpenNeuro. The repositories for Aron et al., 2007 and Xue et al., 2008 are DS000007 [1] and DS000008 [2]. We preprocessed the data using fMRIprep [3] neuroimaging preprocessing tool Esteban et al., 2020. We extract region mean time series data with automated anatomical labeling (AAL) template.

[1] https://openneuro.org/datasets/ds000007/versions/00001
[2] https://openneuro.org/datasets/ds000008/versions/00001
[3] fMRIPrep-21.0.0

4.4.2 Temporal Change-point Analysis

In this subsection, we perform a preliminary analysis on response inhibition datasets with a simple problem, where we showcase the performance of a basic heuristic on temporally synchronized datasets. The simple problem is defined as follows. We assume that cognitive state toggles only once during a session, such that the first k samples are from the first cognitive state and the remaining T-k samples are from the second cognitive state in a session of T samples. We estimate the state change time-point k (temporal change-point) and the cognitive states before and after the time-point k, as part of this initial analysis.

We analyze the Word, Manual and Signal datasets, defined in the subsection 4.4.1. There are 4 labels in each of the datasets; "junk", "failed", "go" and "successful". We classify the fMRI images, labeled "successful" and "failed". The originally heterogeneous datasets are temporally synchronized, such that the samples of each cognitive state are grouped together. The temporally synchronized datasets are hosted as part of the study by Yousefnezhad et al., 2020. We refer the reader to section 4.1.4 for details of temporal synchronization.

We start with an **oracle method** that is aware of the initial state of the subject and we only estimate the state-change time-point with a basic heuristic method. The **non-oracle method** assumes transition direction is also unknown. There two cognitive states in the dataset, hence, if the oracle method detects the change point correctly, it can classify all of the time points correctly. As the basic heuristic to find the state change time-point, we use the time-difference function $t - t_f$ on time series data, where t_f is the f time-points delayed time series data and the maximum on this function is taken to be the change-point.

In mathematical notation, $X \in \mathbb{R}^{T \times N}$, where T is the number of time-points in the session and N is the number of voxels in the region of interest. Mean time-series data is $X_{mean}(t) = \frac{1}{N} \sum_{i=1}^{N} X(t, i)$, where $t \in (1..T)$. The time-difference signal is $X_{timediff}(t) = X_{mean}(t) - X_{mean}(t-f)$, where f is a given time delay and $t \in [f..T]$. The basic heuristic finds the time-point t,

$$\arg\max_{t} X_{timediff}(t).$$

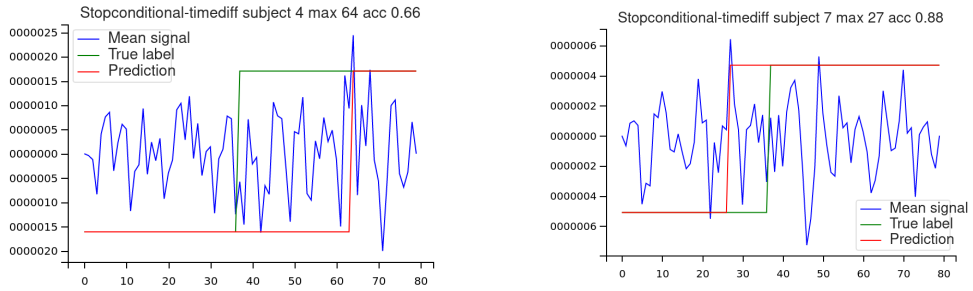


Figure 4.5: Heuristic prediction and class label superimposed on time-difference signal

We split the time-series data into chunks of c consecutive samples, such that there are $\frac{T-f}{c}$ chunks of data. Let k denote the estimated change time-point, we take equal number of chunks from the two intervals, (f..k) and (k..T), such that there are min($\frac{k-f}{c}$,$\frac{T-k}{c}$) chunks in both intervals.

As the classifier function, we tested with both a support vector machine (SVM) and a 2-layer multi-layer perceptron (MLP) and observed no significant difference. We list the MLP results below. We use the MLP implementation of Scikit-Learn library (Pedregosa et al., 2011). There are 100 parameters in the hidden layer. We use the rectified linear unit activation and the ADAM (Kingma and Ba, 2015) optimizer with $10^{-}4$ learning rate. We establish leave one subject out cross-validation, where we keep one subject's data out in each fold. In each fMRI image, the spatial dimension is constrained to a region of interest (ROI) that is common among all datasets. The ROI is composed of 19174 voxels. One hyperparameter of our analysis method is the size of each chunk in number of time-points. We repeat the experiment for varying chunk sizes in the plots of this subsection.

In the following experiments, we train a classifier on a source dataset and evaluate it on a target dataset. We use the basic heuristic change point based classifier, illustrated in 4.5.

In the following, we list the classification results for varying data chunk size c for three cases, in-dataset, between single datasets, and between multiple source datasets to single target dataset.

In figures 4.6 and 4.7, we list results for single and multiple source transfer learning.

In the figure captions, datasets are defined with a single letter as given below. Blue line is the mean accuracy of the model trained only on the target dataset, that is on the right side of the arrow.

Chunk size takes values [2, 5, 10, 12, 15, 18, 20, 24, 28, 30]. Each point in the plot shows the result of 10-fold cross validation mean, and the error bar around the point shows the standard deviation of cross-validation. The letter definitions for each dataset are as follows. **A**, **B** and **D** stand for "Word", "Manual" and "Signal" datasets. Figures show the source and target datasets in the following form, $Source \rightarrow Target$.

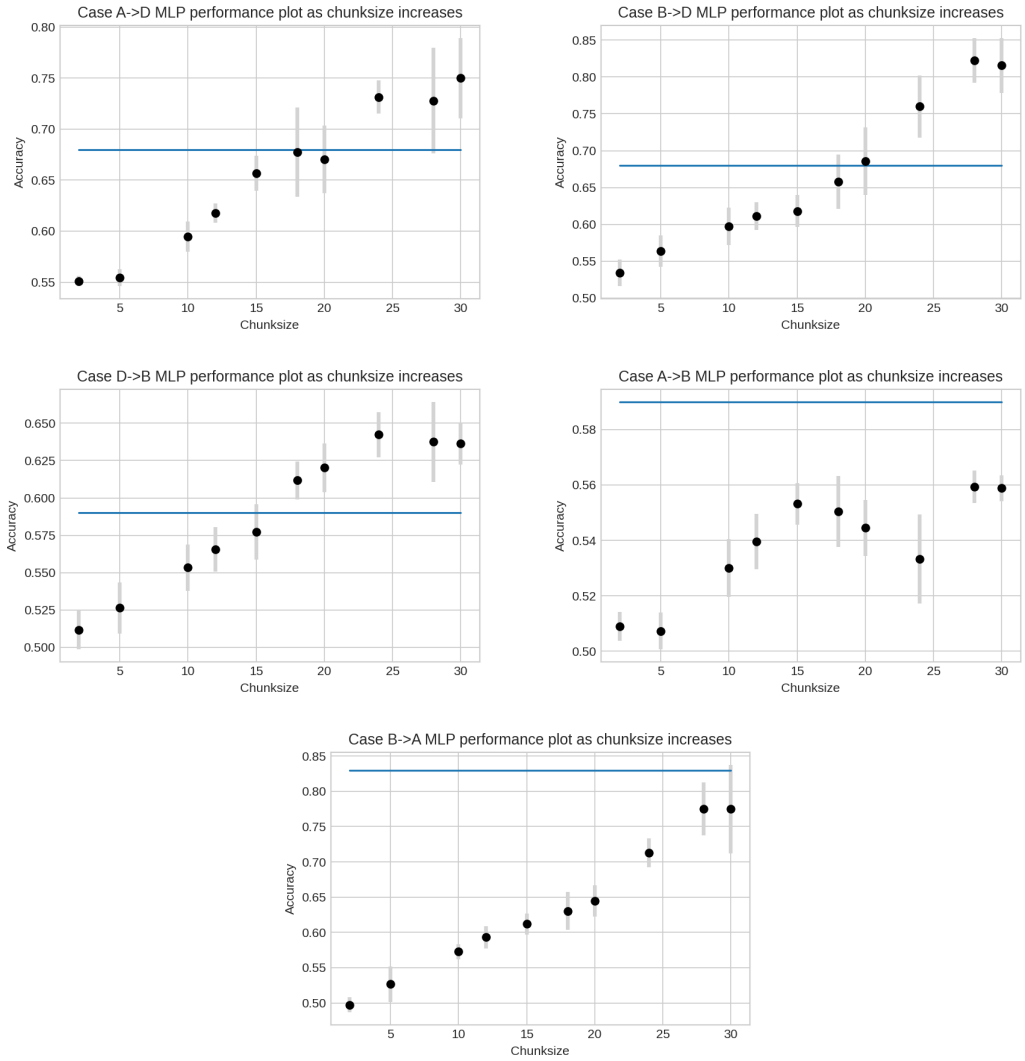


Figure 4.6: 10-fold stratified cross-validation accuracy for single source dataset to single target dataset transfer learning

Due to the sampling method around the change point, increasing the chunk size reduces the number of samples that can be obtained. This happens when the detected change point is close to the start or end of the time-series, which becomes a limiting factor for number of samples that can be taken. For dataset D (stopconditional dataset), this case was prevalent and models trained on other datasets have performed even better than the model trained on D.

In most of the cases, there is a steady increasing trend of accuracy as chunk size increases. The plots in both single and multiple source datasets validate this observation.

Multiple source datasets improve transfer learning performance. We observe in plots [A→D, B→D] and [AB→D], that, when D is the target dataset, performance surpasses %85 while average single dataset best performance is %78.5. We also observed that when there are multiple source datasets [AB→D, AD→B], target-only training (blue line) is surpassed at a lower chunk size ($|c|$=15, 12), compared to single source cases [A→D, B→D] ($|c|$=[18,20]).

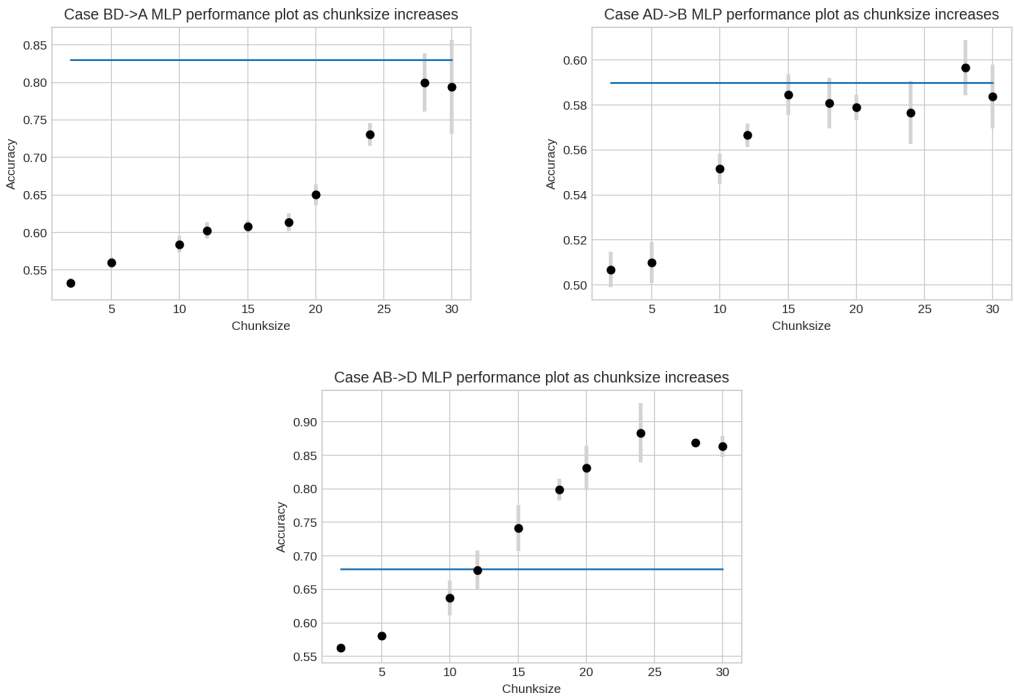


Figure 4.7: 10-fold stratified cross-validation accuracy for multiple source datasets to single target dataset transfer learning

In this preliminary experiment, we locate a large-variance time-point and take samples around that point. When we do not consider this variant of sampling around a critical point, we observe that a multi-layer perceptron or a support vector machine can not distinguish between classes.

Our analysis leads to the following outcomes. Multiple source datasets improve transfer learning performance. Multiple source datasets allow a lower chunk size compared to single source transfer learning. The results support the utility of multiple datasets in small scale transfer learning tasks, from the perspective of required chunk size and overall performance.

One drawback of our method is that it requires the whole time interval of a subject. Furthermore, the temporal synchronization allows the simple heuristic that perform well, which groups the time-points of each cognitive together. The most important drawback is due to the unrealistic and simple problem setting, since, in reality, there are multiple changes of cognitive states in a session duration and the order of the cognitive states are arbitrary.

In the following subsections, we replace temporal alignment with spatial alignment. Furthermore, instead of using the data hosted in the recent study, we preprocess the raw data from scratch for reproducibility. We use recent preprocessing a standardized preprocessing tool-chain for a reproducible result.

4.4.3 Template-aligned GCCA

This section is a revised version of Eryol and Vural, 2022b presented at IEEE SIU conference. Instead of using the hosted dataset in Yousefnezhad et al., 2020, we preprocess the raw data hosted on OpenNeuro repository with fMRIprep tool by Esteban et al., 2018.

We classify the aligned data samples with Support vector machine (Cortes and Vapnik, 1995) with radial basis function kernel.

We employ leave-one-subject-out (LOSO) cross-validation (CV) for all TL cases and methods. The figures also include the standard deviation for each point on the curve

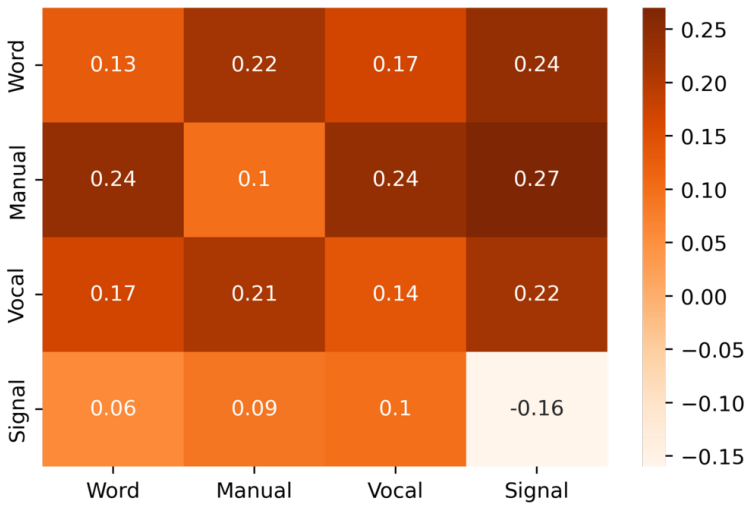


Figure 4.8: Single dataset in source set. Positive value shows the additional increase in performance of SSTL-V (Eryol and Vural, 2022b), with respect to the baseline method.

with bars. Each result table is also available in the appendix. We observed that the model has a low standard deviation on LOSO CV.

In our experimentation setting, we evaluate transfer learning from source set composed of a single dataset in figure 4.8 and two-dataset combinations in figure 4.9. In both figures, each cell of the matrix shows the difference of classification performance between template-aligned samples and standard principal component analysis.

The row headers in both figures show the source datasets and column headers shows the target datasets, used in the transfer learning experiment. Each cell shows the performance improvement over the baseline method, when SSTL-V is used for feature alignment.

In the figures, the cells which have a common dataset in both source and target set are ignored in the overall comparison tables in the next section, as these are not valid transfer learning cases. We observe a large performance improvement with respect to the baseline in figure 4.9 for the cases, where "Signal" is the target dataset (the last column). This result is important since the "Signal" dataset belongs to the separate study, and the datasets "Word", "Manual" and "Vocal" belong to a common study.

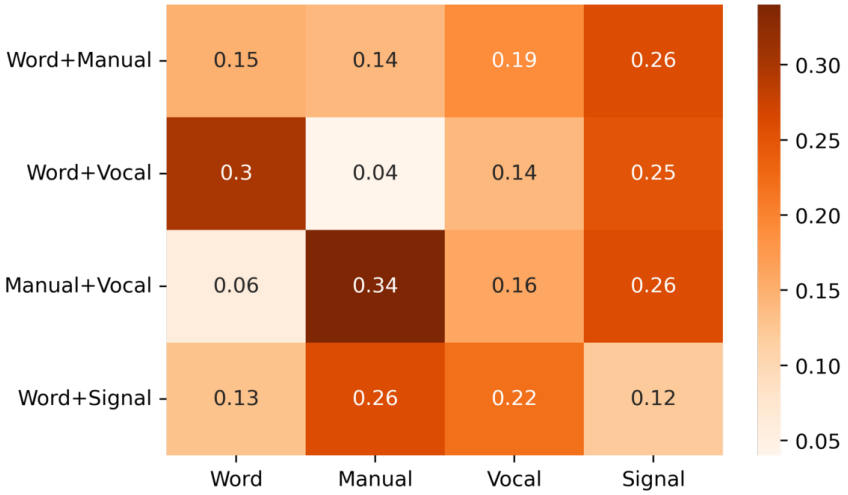


Figure 4.9: Two datasets in source set. Positive value shows the additional increase in performance of SSTL-V (Eryol and Vural, 2022b), with respect to the baseline method.

4.4.4 Hierarchical Feature Alignment with Brain Region Covariance and Supervised GCCA

This section is a revised version of Eryol and Vural, 2022a presented at IEEE BIBE conference, organized in 3 parts. We firstly explain the transfer learning setting. Secondly, we show the results versus recent state of the art method. Thirdly, we list figures of transformation matrices, visualized on the brain template/atlas.

4.4.4.1 Transfer learning setting

The transfer learning setting in Yousefnezhad et al., 2020 works on non-intersecting source and target datasets. During model training, the source dataset/s are aligned on session-subject-dataset hierarchy. The support vector machine (SVM) model is trained on generated features in the feature alignment step.

The same feature alignment process is repeated for the target dataset with one exception, the global parameter W is transferred from the training phase, hence G and W parameters are recalculated. It is important to note that the classifier is not retrained on target data nor it is partially adapted to the target dataset.

4.4.4.2 Hierarchical Feature Alignment Results

In the experiments, SVM uses radial basis function (RBF) kernel. The lower dimension number takes values $k = [10, 20, 30, 40]$, for reduced rank SVD. As suggested in Barshan et al., 2011, for SH-PCA, a linear kernel is used. The incremental PCA method proposed in Brand, 2002 is followed in Yousefnezhad et al., 2020 to manage the space complexity of their model. Our suggested data representation with AAL template reduces the number of spatial dimensions dramatically that does not require using an incremental PCA method.

In the following experiment H-PCA stands for the hierarchical feature alignment method and SH-PCA stand for the label-guided H-PCA. SSTL-V stands for the template aligned GCCA method.

We group the methods used in the experimentation based on label-guided, drift-aware and multi-dataset properties. The baselines with label-guided property learn a representation of input data taking the label information into consideration. Drift-awareness refers to handling of the data distribution discrepancy between groups of data that are known in advance, i.e. between sessions of the same subject. Hierarchy property refers to methods that accommodate the subsumption relation between groups of data.

The following baselines learn a low dimensional representation; SSTL-V, PCA, conditional VAE (Sohn et al., 2015), β-VAE (Higgins et al., 2017) and σ-VAE (Rybkin et al., 2021). In the RAW case, we simply use the data with no feature generation and evaluate the trained SVM model on the target dataset.

We adopt the leave-one-subject-out technique to rule out subject specific results. An experiment is repeated for each subject where in each repetition, we remove one subject's data, corresponding to $\sim$16 single source repetitions and $\sim$32 times multiple source dataset repetitions. on figures 4.10 and inside parentheses on table 4.4.

Three of the datasets (Word, Manual, Vocal) used in our results are obtained from the study Xue et al., 2008, while the last one (Stop) is from a separate study Aron et al., 2007.

Table 4.3: Baseline method properties. (+: template-aligned version of the state-of-the-art method (Yousefnezhad et al., 2020), *:Conditional-VAE Sohn et al., 2015 adapted to β-VAE Higgins et al., 2017)

Methods	**Label-guided**	**Drift-aware**	**Hierarchy**
RAW	X	X	X
PCA	X	X	X
β-VAE in Higgins et al., 2017	X	X	X
σ-VAE in Rybkin et al., 2021	X	X	X
Conditional β-VAE *	X	✓	X
SSTL-V in Eryol and Vural, 2022b +	X	✓	✓
SH-PCA in Eryol and Vural, 2022a	✓	✓	✓
H-PCA in Eryol and Vural, 2022a	X	✓	✓

In the figure 4.10, we show the results where the source and target datasets are from separate studies, which introduces an additional challenge.The relatively high performances of the proposed methods (around 85%) for the independent dataset "Stop", shows the robustness of the suggested feature alignment model for transfer learning settings for brain decoding.

Figure 4.10 shows the comparative accuracy of the suggested methods.Since RAW and PCA methods employ data sets with no feature alignment, the dimension of the feature space is constant for label-guided, drift-aware and multi-dataset experiments, as shown in table 4.4. In table 4.4, a valid case has no intersection between source and target datasets. The single source dataset and multi-source dataset results are the two parts of the table. We report the mean and standard deviation for each cell on the table. The RAW method has no lower dimensional representation and the same result applies for all k-values. Figure 4.10 indicates that the SSTL-V method significantly improves the performance of the PCA method. Furthermore, our H-PCA and its supervised version, SH-PCA have superior performances compared to SSTL-V method.

In figure 4.10, we observe that the unsupervised transfer learning method, H-PCA,

outperforms other methods in single source cases. On the other hand, for the multi-source cases, the supervised method, SH-PCA, performs better than the unsupervised variant. We saw that, SSTL-V, has the highest standard deviation among the top three methods. Table 4.4 also supports the results in figure 4.10. In single-source case, our H-PCA improves the state-of-the-art method of SSTL-V by 5.6% in accuracy on average. In the multi-source cases, our supervised variant SH-PCA performs slightly better than H-PCA and outperforms the state-of-the-art method by 4.5% on average. Simply performing PCA on data and classifying the lower dimensional representation is on par with using the raw data, and is very close to chance level.

The VAE variants depend on the beta parameter. The results for the beta parameter sweep for each across-study transfer learning case are in the figure 4.11. We also list the convergence plots up to 300k epochs in figure 4.12.

Table 4.4: Mean transfer learning over all valid cases. k: number of dimensions in the lower dimensional representation. (*: BIBE, **: Eryol and Vural, 2022b modified work of Yousefnezhad et al., 2020). β-VAE in Higgins et al., 2017, σ-VAE in Rybkin et al., 2021, conditional VAE in Sohn et al., 2015

Methods	**k=10**	**k=20**	**k=30**	**k=40**
Single source				
RAW	0.54(0.03)	0.54(0.03)	0.54(0.03)	0.54(0.03)
PCA	0.55(0.02)	0.54(0.02)	0.54(0.02)	0.54(0.02)
β-VAE	0.55(0.02)	0.54(0.02)	0.54(0.02)	0.54(0.02)
σ-VAE	0.55(0.03)	0.56(0.02)	0.56(0.00)	0.55(0.03)
Conditional β-VAE	0.55(0.02)	0.54(0.02)	0.54(0.02)	0.54(0.02)
SSTL-V **	0.77(0.01)	0.78(0.01)	0.77(0.01)	0.77(0.01)
H-PCA*	0.82(0.01)	0.83(0.02)	0.83(0.02)	0.83(0.02)
SH-PCA*	0.81(0.01)	0.82(0.01)	0.82(0.01)	0.82(0.01)
Multi source				
RAW	0.55(0.04)	0.55(0.04)	0.55(0.04)	0.55(0.04)
PCA	0.55(0.03)	0.54(0.03)	0.53(0.03)	0.53(0.02)
β-VAE	0.55(0.02)	0.54(0.02)	0.54(0.02)	0.54(0.02)
σ-VAE	0.55(0.03)	0.55(0.03)	0.55(0.03)	0.55(0.03)
Conditional β-VAE	0.55(0.02)	0.54(0.02)	0.54(0.02)	0.54(0.02)
SSTL-V **	0.78(0.02)	0.82(0.04)	0.82(0.05)	0.82(0.05)
H-PCA*	0.83(0.01)	0.84(0.01)	0.85(0.01)	0.85(0.01)
SH-PCA*	0.85(0.01)	0.85(0.00)	0.86(0.01)	0.86(0.01)

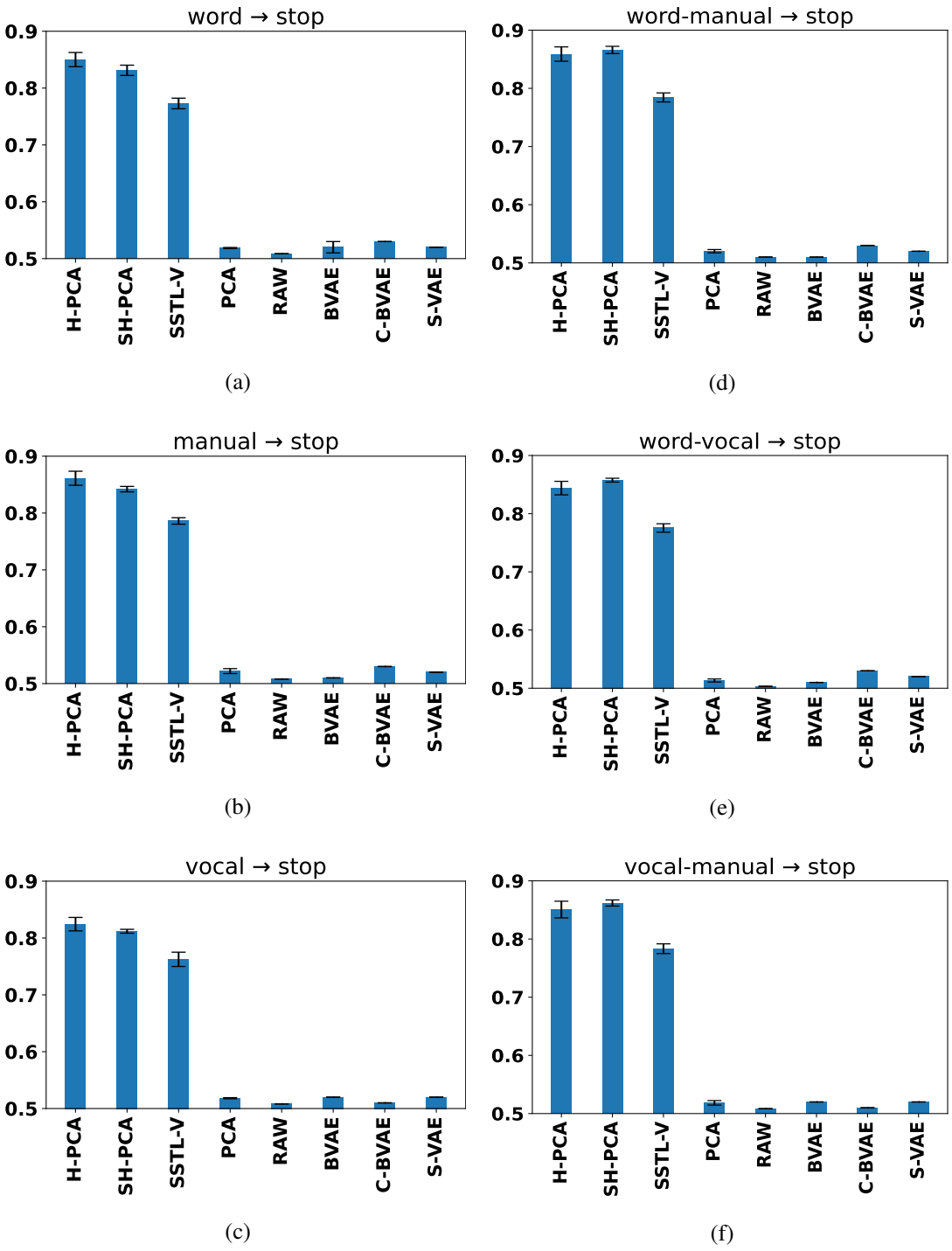


Figure 4.10: Transfer between independent studies; source dataset from Xue et al., 2008 → target dataset from Aron et al., 2007. Left hand side of the arrow shows the source dataset(s) and right hand side shows the target dataset. Each bar shows the accuracy averaged over low-dimensional representation feature size with standard deviation error bars.

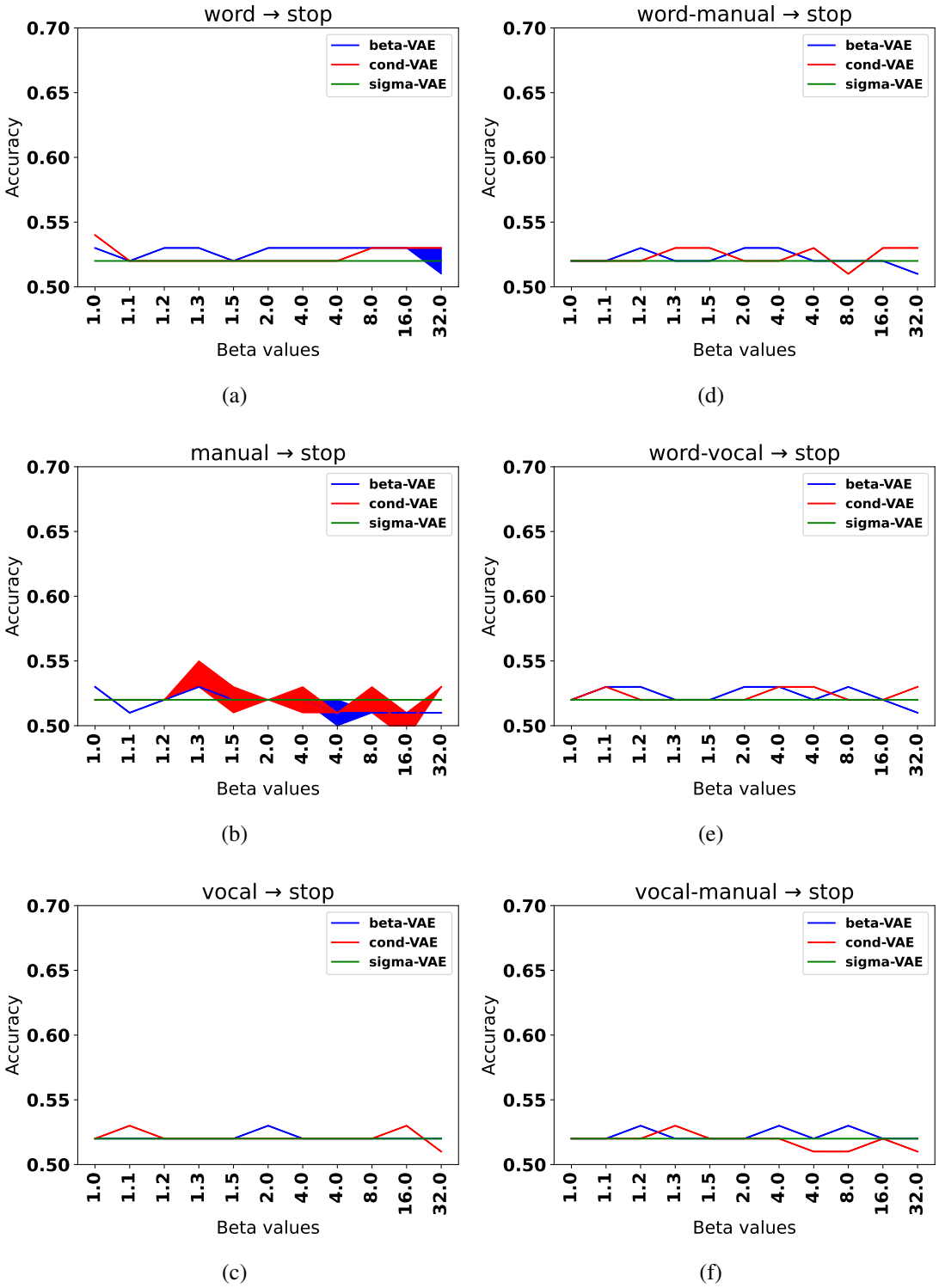


Figure 4.11: Performance of VAE variants over varying beta values. Transfer between independent studies; Xue et al., 2008 → Aron et al., 2007. Subfigures a-c) have single and subfigures d-f) have multi source datasets in TL task.

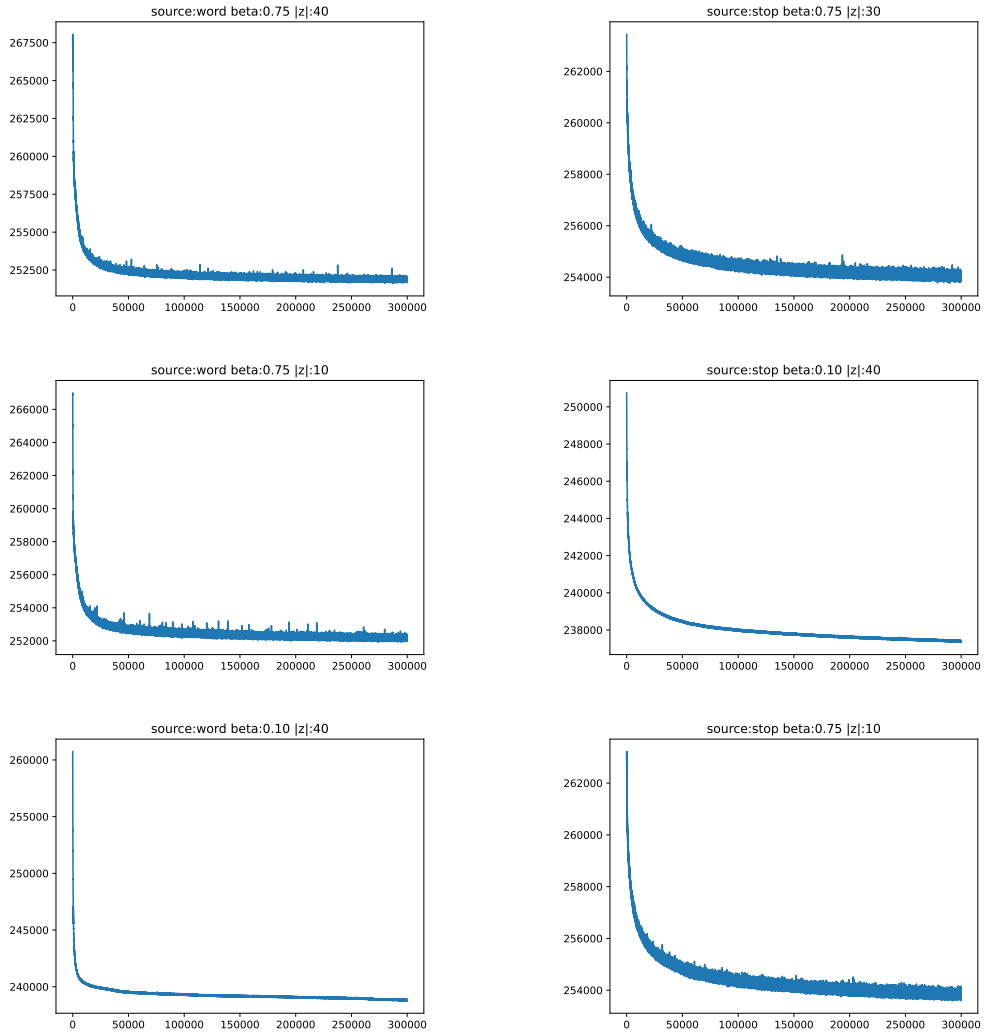


Figure 4.12: Sample convergence plots for β-VAE. Each plot shows the change of loss over epochs for a source dataset, latent dimension size and β value. Total of 300k epochs, latent dimensions set [10, 20, 30, 40], β values [0.1, 0.25, 0.50, 0.75, 1, 2, 4, 8, 16, 32, 64]. Note that σ-VAE Rybkin et al., 2021 adds a closed form parameter to estimate the β value.

4.4.5 Visualization of Region Specific Weights

In this section, we show the visualization of learned coefficients. The objective function of GCCA is of the form in equation 4.23.

$$\min_{G,R} \sum_{j=0}^{J} \|G - XR\|_F^2 \tag{4.23}$$

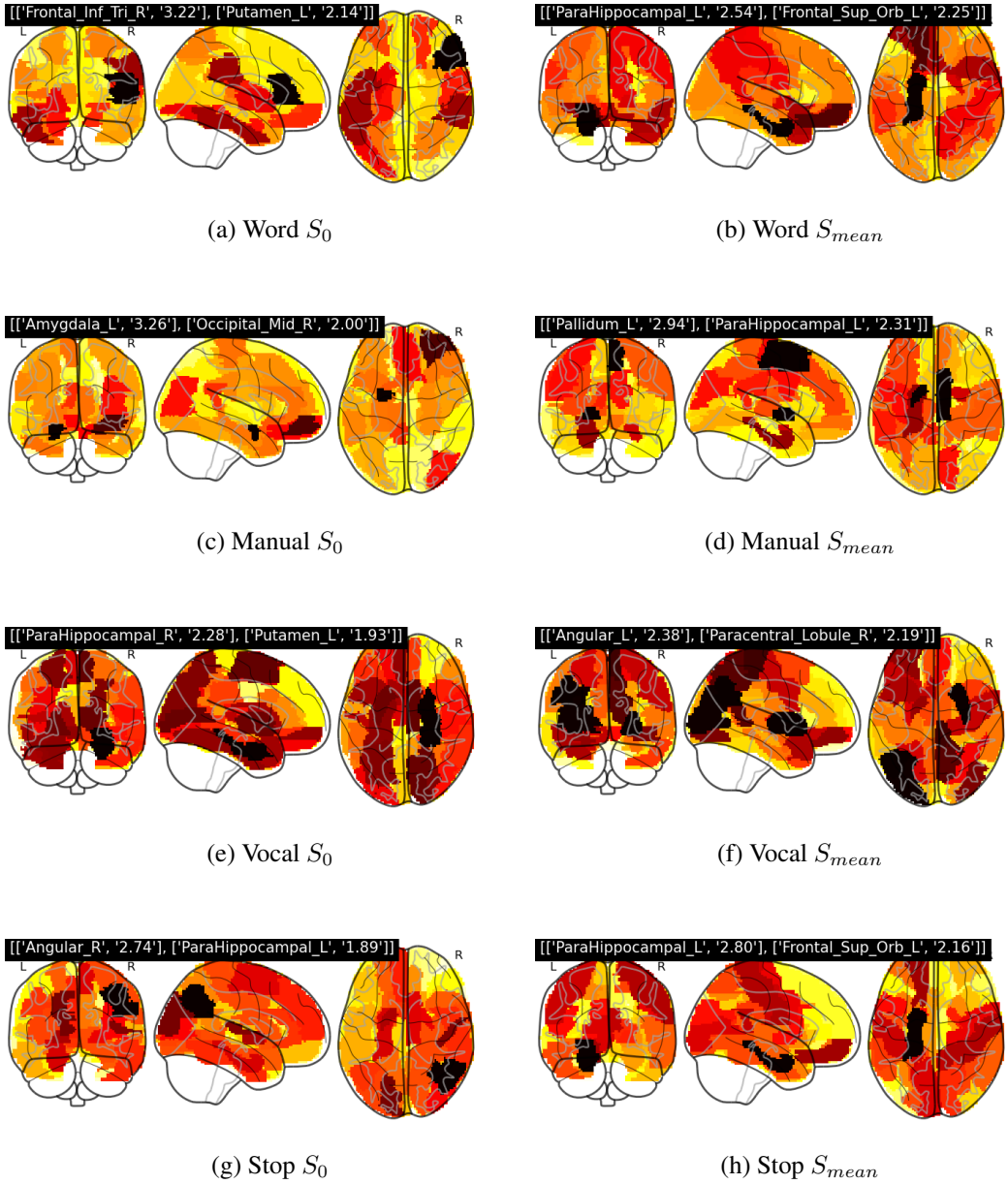
The max-var solution, used in our previous work Eryol and Vural, 2022a, has a closed form solution. In this formulation, view-coefficient R_j is substituted such that $X_j R_j$ is the projection of view X_j on the common subspace G, in equation 4.24, where $R_j = (X_j^T X_j)^{-1} X_j^T G$.

$$\min_{G} \sum_{j=0}^{J} \left\|G - X_j (X_j^T X_j)^{-1} X_j^T G\right\|_F^2 \tag{4.24}$$

Both the GCCA and successive PCA operations generate linear combination of feature dimensions. We find the top three weighted dimensions each corresponding to a brain region.

Recall that, in our feature alignment solution, we apply GCCA at subject level over sessions, such that G_s is the subject specific subspace. Below we show subject-specific z-score maps of $P_{r,s,d} G_{s,d} W_d$, which are transformed samples on AAL template on the left column and mean $G_{s,d}$ per dataset d.

In the following figures 4.14 and 4.15, we investigate the role of alignment weights on the original data dimensions that correspond to brain regions on AAL atlas. The dimensions with a higher weight have a bigger role in the alignment process. We show the sorted brain region occurrence counts as bar plots per dataset in figures 4.14 and 4.15. For word dataset, left rolandic operculum stands out, that is related to visceral sensation and stress in Sutoko et al., 2020. Right supramarginal gyrus is related to language perception and processing Wikipedia, 2022, that occurs for the manual dataset. For both vocal and stop datasets, temporal pole occurs as the common dimension, which is associated to semantic memory in Muzio, 2022.



(a) Word S_0

(b) Word S_{mean}

(c) Manual S_0

(d) Manual S_{mean}

(e) Vocal S_0

(f) Vocal S_{mean}

(g) Stop S_0

(h) Stop S_{mean}

Figure 4.13: Visualization of subject and dataset specific multipliers per region. The region names are the top-2 highest absolute magnitude dimensions.

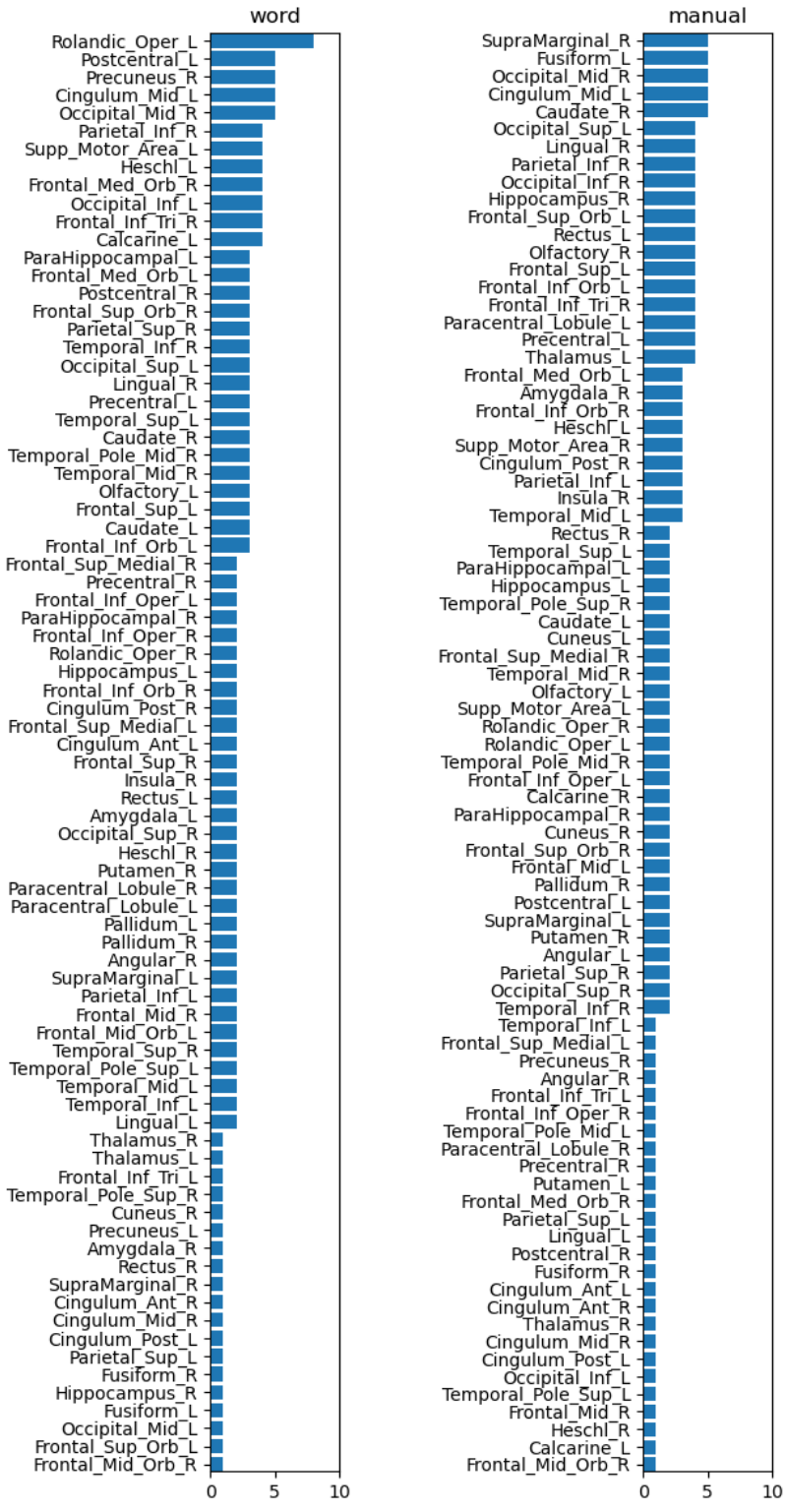


Figure 4.14: Word and manual dataset bar plots for the occurrence count of each region in the top 10 highest weighted regions per dataset, and weights $G_{s,d}W_dW$.

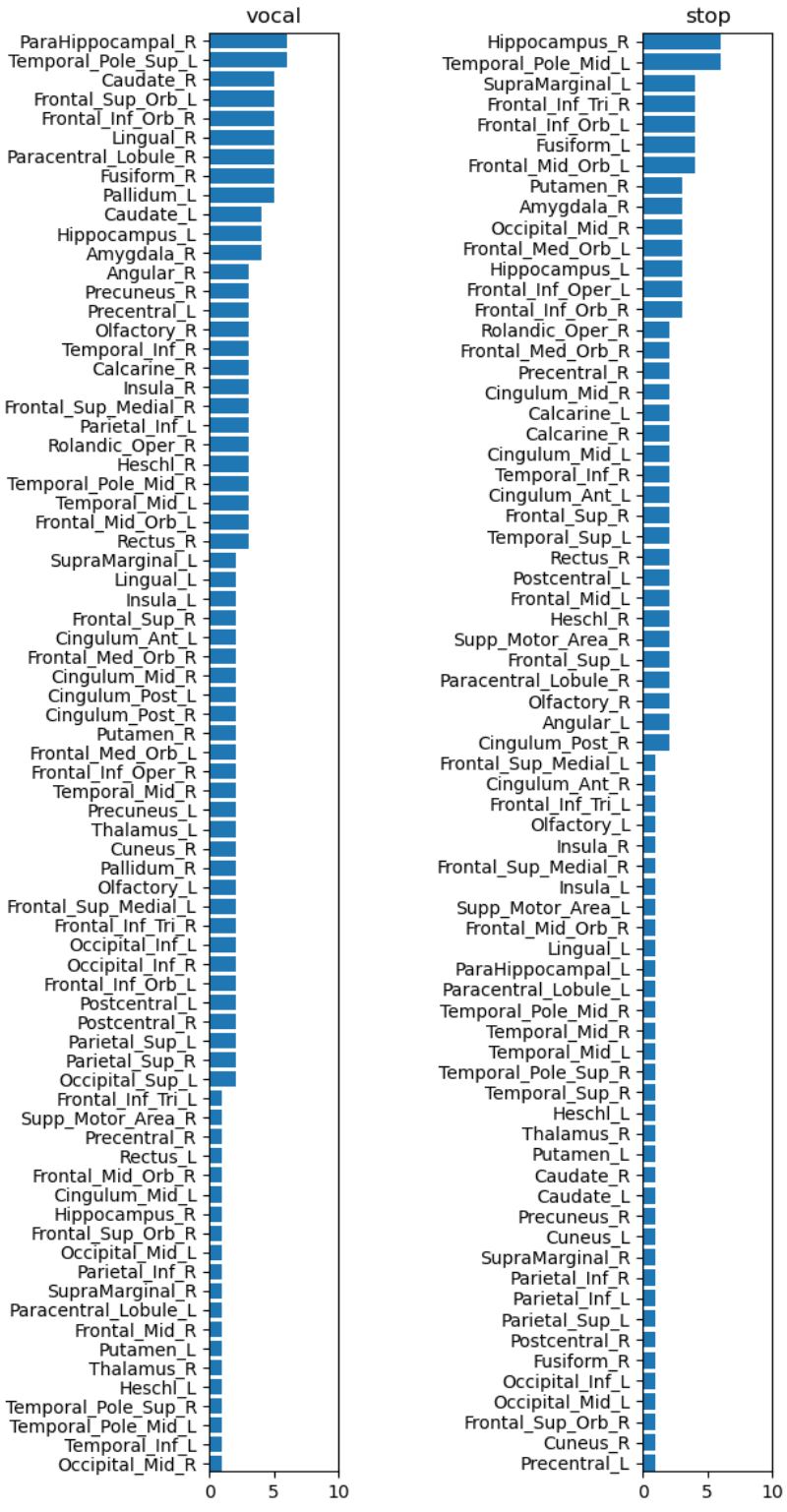


Figure 4.15: Vocal and stop dataset bar plots for the occurrence count of each region in the top 10 highest weighted regions per dataset, and weights $G_{s,d}W_dW$.

4.4.5.1 Ablation study

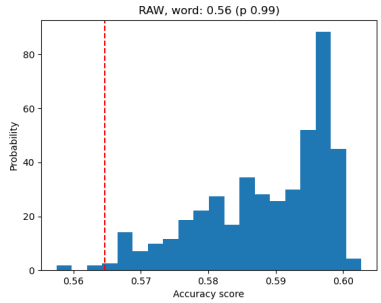
We adopt a permutation based significance test in Ojala and Garriga, 2009. This method generates an empirically random dataset from the original one to test the null hypothesis that samples and labels are independent. The random set is generated via permutation of labels on samples. Since this step is infeasible to compute, they generate a limited number of randomized datasets as a Monte Carlo approximation. We apply this method to test the significance in transfer learning, rather than supervised learning.

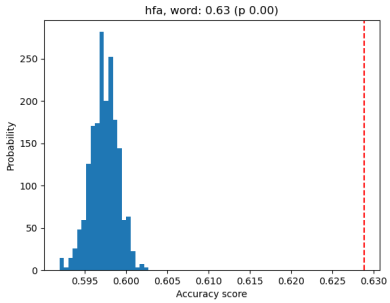
Given a dataset $D = \{X_i, y_i\}$, $i \in [1, N]$ and permutation function π; randomized dataset is calculated as $\hat{D} = \{X_i, \pi(y)_i\}$. The p-value 4.25 is the ratio of randomized datasets whose performance is higher than the non-random dataset.

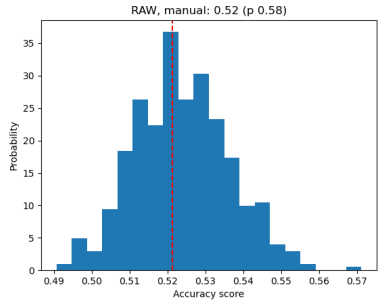
$$p = \frac{\mid D' \in \hat{D} : e(f, D') \leq e(f, D) \mid +1}{k + 1} \tag{4.25}$$

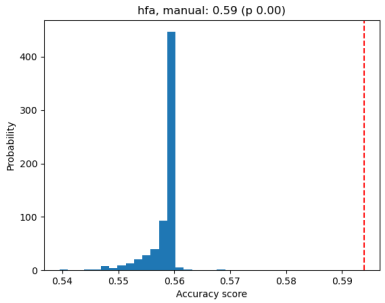
The figures are generated for 100 random permutations of labels (2 fold cross-validation due to time constraints) as the null hypothesis. The red line shows the score on original data and blue bars are the histogram of scores over randomized data. p-value for raw data, data aligned by SSTL-V and data aligned by SH-PCA are 0.01, 0.06, 0.01, in respective order.

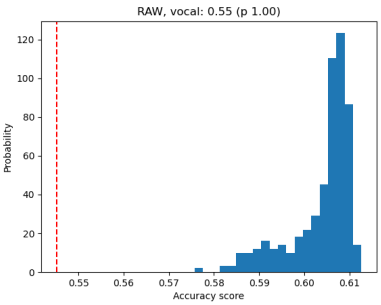
This is a preliminary result that shows gCCA based alignment method with label augmentation SH-PCA is more stable compared to SSTL-V. This ablation study is performed on single dataset.

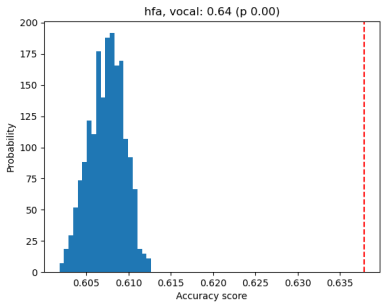


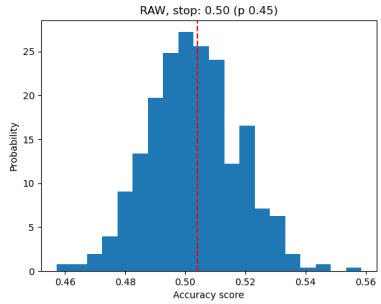


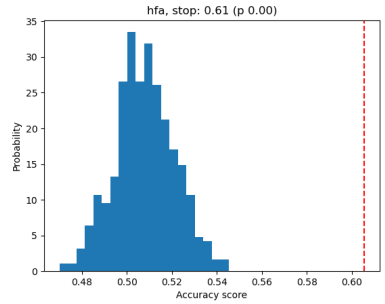












(a) Non-aligned raw data

(b) H-PCA aligned data

Figure 4.16: Randomization test shows the impact of feature alignment under repeated experiments with label randomization. Red line shows performance with real labels, bars shows the randomized label performance histogram.

4.5 Chapter Conclusion

In the temporal change analysis experiment, we start with the assumption that there is a change-point during the experiment timeline. We build a model to predict this change-point. The experiment is carried out on the readily-preprocessed datasets that carried a potential problem of temporal alignment. We show that the exceptional success is weakly related to the data itself.

In the following methods, firstly, we avoided the shortcomings of temporal alignment. Secondly, we preprocessed the data from scratch and removed out-of-interest labels from the data (i.e. samples labeled as "junk", "go"). The removed out-of-interest labels form the majority classes and form an imbalanced label distribution. Furthermore, in a transfer learning setting, the common aspect of data aggregation needs to be of equal size, such as an equal number of time-points or spatial regions/locations. In the benchmark datasets, the common aspect should be different than the temporal dimension, since data acquisition duration and presented stimulus order varies among sessions. We proposed variations of the previous work that aligns data on spatial aspect of the data. Another point in our improvements is the standardization of the spatial dimension among subjects with a brain atlas. The previous work follows a region of interest (ROI), formed by set of voxels, however the number of voxels in the ROI vary between subjects. We used a brain atlas to form a spatial standardization that enables transfer learning on the spatial domain, where each spatial dimension corresponds to the same brain region of the brain atlas among all subjects.

In this work, we proposed two new feature alignment methods for transfer learning on brain decoding data, using a modified maximum variance generalized canonical correlation analysis (maxvar-GCCA) method at its core. The first suggestion in this work is to use a maxvar-GCCA-like solution that suppresses low variance directions and emphasizes high variance directions in the feature space. The second method proposes to utilize the valuable label information in building the feature space.

Both methods hierarchically estimate transformation matrices to align multiple sessions, subjects, and datasets. This hierarchical alignment reduces the inter-session, inter-subject, inter-dataset variances and keeps the label-dependent variation at a low-

dimensional representation step. The suggested approach avoids losing valuable information related to the target stop-signal paradigm.

Both methods outperform the state-of-the-art alignment method (template based variant of Yousefnezhad et al., 2020 based on standard maxvar-GCCA), and steadily in all single-source and multi-source datasets and varying lower-dimensional representations. We observe that the highest impact in the superior performance of our method is the novel maxvar-GCCA-like high-variance-sensitive solution.

The datasets used in our transfer learning setting are part of two studies. An important simulation of the performance of our method in the wild is its performance on an unseen data from a new study. Our proposed method is, also, more successful than standard maxvar-GCCA based state-of-the-art method on the target dataset obtained from an independent study.

CHAPTER 5

SUMMARY AND CONCLUSION

Brain decoding studies generally follow one of the two practices in fMRI data analysis; they either consider a region of interest or a coarse whole-brain data. Fine-grained whole-brain models are a new direction of brain decoding research. Imposing coarse-grained structural prior information is an important part of fine-grained whole-brain models. Furthermore, the limited data problem and varying task-related patterns in each task-fMRI session data introduce difficulties in transfer learning for brain decoding. Due to computational constraints, fine-grained whole-brain models should locally adapt to changes of patterns among data acquisition sessions.

In our first study, we propose a new four dimensional multi-layer perceptron model, called the Structured MLP model, on minimally preprocessed whole-brain fMRI images from the Human Connectome Project task-fMRI dataset (Barch, 2013). In this study, we suggest a model on whole-brain fine-grained data, that is a new research direction. The structured MLP model decomposes each three dimensional fMRI image into non-overlapping volumetric patches. In the 3D convolutional baseline model, the convolution operation reduces the resolution of the encoded features at each successive layer. The decreasing resolution limits our ability to impose the same spatial constraint at each layer. Structured MLP model keeps the resolution of the input three dimensional image equivalent to the encoded features by the MLP block. Equivalent resolution in the encoded feature representation allows us to apply the brain atlas, as the prior for normalization with respect to brain regions, at any intermediate feature in the successive application of the MLP blocks. The most important problem in brain decoding on fMRI images is the change of patterns in the data across sessions, subjects and datasets. The voxels that have a role in the brain-behavior relationship

vary across subjects, however they generally reside in the same brain region defined by the brain atlas. Batch normalization is a well-studied method that reduces the covariate shift Ioffe and Szegedy, 2015 during training. Another application of batch normalization is for reducing the discrepancy between datasets. The across session change is addressed in the Structured MLP via a regional normalization method, a special form of batch normalization that is decomposed into regions. We follow a specific batch sampling procedure, where we initially learn intra-session differences among classes. Then, we learn inter-subject differences. The advantage of the Structured MLP is that it shows on-par performance in convergence time, compared to the pre-trained 3D convolutional model, where the pre-training necessitates a large scale source dataset. The drawback of the Structured MLP is that it is hard to manage the number of parameters, due to two reasons. Firstly, the intermediate features that have an equivalent resolution in the Structured MLP model introduces a large parameter overhead. Secondly, the number of parameters in our MLP-Mixer (Tolstikhin et al., 2021) variant depends on the patch size, where reducing the patch size in the three dimensional volume increases the number of patches exponentially. It is hard to interpret, stabilize and smoothen the model parameters, due to the the black-box nature of the model, opposed to the hierarchical feature alignment methods. Furthermore, our hardware constraints have been a limiting factor in exploring a larger set of hyperparameters and subject sets, due to the large number of parameters of Structured Multi-layer Perceptron.

In our second study, we generate transferable features from multiple source datasets that improves the brain decoding performance on the target dataset. We follow the transfer learning benchmark in Yousefnezhad et al., 2020. The proposed feature alignment models preserve the covariance of brain regions at successive linear transformations, applied at session, subject and dataset levels . Furthermore, we proposed a supervised variant of the transferable feature generation method, inspired by Barshan et al., 2011. The core method, called Hierarchical-Group Principal Component Analysis (H-PCA), suppresses low variance directions and emphasizes high variance directions in the feature space. The supervised variant, called Supervised Hierarchical-Group Principal Component Analysis, imposes dependency on labels in the core method, H-PCA. We experiment on transfer learning performance across independent studies,

Xue et al., 2008 and Aron et al., 2007, that follow the common stop-go paradigm. There are four datasets of stop-go paradigm, where three datasets are obtained from the first study and one dataset is obtained from the second study. In the experiments, we report the transfer learning performance for two cases; "single source dataset to target dataset" and "multiple source datasets to target dataset" transfer learning experiments. We see that H-PCA method outperforms the baselines steadily in both single source and multiple source transfer learning experiments. A limitation in our model is the brain region mean time-series representation of the whole-brain fMRI images, that discards local patterns in a region. Furthermore, we require the number of spatial dimensions to be equal for all subjects in an experiment, which avoids using the proposed method for a ROI representation, where number of voxels may vary across subjects. The linear methods become computationally infeasible on raw whole-brain images due to hardware constraints, as opposed to the MLP model in the third chapter.

A general comparison between the two approaches, Structured MLP and Hierarchical Feature Alignment, is given in the table 5.1.

The application areas of black-box methods are generally in biomarker design, that can require real-time monitoring, 4D whole brain processing for disease tracking and preventive medicine. But the reliability-wise, this approach is still an early-stage research. On the other hand, linear methods with well-known behavior are more reliable.

As a future work, for both studies, we plan to adapt recent brain atlases to improve our work, for instance Schaefer (Schaefer et al., 2018) brain atlas or Multi-model Parcellation (Glasser et al., 2016) brain atlas. We foresee that the Structured MLP model is suitable for disease biomarker design problems, namely early detection and progression monitoring of the Alzheimer's Disease or Attention-Deficit Hyperactivity Disease. Finally, we plan to utilize the Hierarchical Feature Alignment model on aligning neural network features to reduce distribution discrepancy in the neural network representations.

Table 5.1: Comparison of proposed solutions in this thesis

Method	Supervision in TL	Data Properties	Motivation	Outcome
Structured MLP	• Supervised TL • Brain atlas prior information	• 4D fMRI image • $N \approx 10^7$ voxels • Large scale dataset • $N_{subj} = 1200$	• Biomarker design • Real-time processing • Fine-grained whole brain image processing	• Faster convergence via fine-tuning • Controlling the effect of each region
Feature Alignment	• Unsupervised TL • No shared subjects • Subsumption hierarchy	• Covariance of brain regions • $N = 116$ brain regions • Mechanistically common tasks • Small scale datasets • $N_{subj.} \approx 18$	• Improving clinical study statistical power • Reliable • Preserves variation among brain regions	• Higher unsupervised TL performance • Successful TL between independent studies • Outperforms SOTA method

REFERENCES

Adebayo, J., Gilmer, J., Muelly, M., Goodfellow, I., Hardt, M., & Kim, B. (2018). Sanity Checks for Saliency Maps [Issue: Nips arXiv: arXiv:1810.03292v2], In *NeurIPS*. Issue: Nips arXiv: arXiv:1810.03292v2.

Akaho, S. (2007). *A kernel method for canonical correlation analysis* (tech. rep. arXiv:cs/0609071). arXiv. https://doi.org/10.48550/arXiv.cs/0609071

Aron, A. R., Behrens, T. E., Smith, S., Frank, M. J., & Poldrack, R. A. (2007). Triangulating a Cognitive Control Network Using Diffusion-Weighted Magnetic Resonance Imaging (MRI) and Functional MRI. *Journal of Neuroscience*, *27*(14), 3743–3752. https://doi.org/10.1523/JNEUROSCI.0519-07.2007

Aydöre, S., Thirion, B., & Varoquaux, G. (2019). Feature Grouping as a Stochastic Regularizer for High-Dimensional Structured Data, In *ICML*.

Ba, J. L., Kiros, J. R., & Hinton, G. E. (2016). Layer Normalization [arXiv:1607.06450 [cs, stat]]. arXiv. https://doi.org/10.48550/arXiv.1607.06450

Bach, F., Jenatton, R., & Mairal, J. (2012). Structured sparsity through convex optimization. *preprint*arXiv arXiv:1109.2397v2, 1–27.

Barch, D. M. (2013). Function in the human connectome: Task-fMRI and individual differences in behavior [Publisher: Elsevier B.V.]. *Neuroimage*, *80*, 169–189. https://doi.org/10.1016/j.neuroimage.2013.05.033

Barch, D. M., Burgess, G. C., Harms, M. P., Petersen, S. E., Schlaggar, B. L., Corbetta, M., Glasser, M. F., Curtiss, S., Dixit, S., Feldt, C., Nolan, D., Bryant, E., Hartley, T., Footer, O., Bjork, J. M., Poldrack, R., Smith, S., Johansen-Berg, H., Snyder, A. Z., & Van Essen, D. C. (2013). Function in the human connectome: Task-fMRI and individual differences in behavior. *Neuroimage*, *80*, 169–189. https://doi.org/10.1016/j.neuroimage.2013.05.033

Barshan, E., Ghodsi, A., Azimifar, Z., & Zolghadri Jahromi, M. (2011). Supervised principal component analysis: Visualization, classification and regression on

subspaces and submanifolds. *Pattern Recognition*, *44*(7), 1357–1371. https://doi.org/10.1016/j.patcog.2010.12.015

Beer, J. C., Aizenstein, H. J., Anderson, S. J., & Krafty, R. T. (2018). Incorporating Prior Information with Fused Sparse Group Lasso : Application to Prediction of Clinical Measures from Neuroimages [arXiv: arXiv:1801.06594v3]. *Arxiv*, *15261*.

Brand, M. (2002). Incremental singular value decomposition of uncertain data with missing values, In *ECCV*.

Cai, M. B., Shvartsman, M., Wu, A., Zhang, H., & Zhu, X. (2020). Incorporating structured assumptions with probabilistic graphical models in fMRI data analysis. *Neuropsychologia*, *144*. https://doi.org/10.1016/j.neuropsychologia.2020.107500

Chen, L., Gautier, P., & Aydore, S. (2020). DropCluster : A structured dropout for convolutional networks [arXiv: arXiv:2002.02997v1].

Chen, P.-h. (2017). Multi-view Representation Learning with Applications to Functional Neuroimaging Data, (September).

Chen, P., Chen, J., Yeshurun, Y., Hasson, U., Haxby, J. V., & Ramadge, P. J. (2015). A Reduced-Dimension fMRI Shared Response Model, In *NeurIPS*.

Chen, P., Zhu, X., Zhang, H., Turek, J. S., Chen, J., Willke, T. L., Hasson, U., & Ramadge, P. J. (2016). A Convolutional Autoencoder for Multi-Subject fMRI Data Aggregation. *arXiv:1608.04846 [cs, stat]*. Retrieved November 1, 2021, from http://arxiv.org/abs/1608.04846

Cortes, C., & Vapnik, V. (1995). Support-vector networks. *Mach Learn*, *20*(3), 273–297. https://doi.org/10.1007/BF00994018

Duda, R. O., Hart, P. E., & Stork, D. G. (2001). *Pattern Classification* (2nd ed.). New York, Wiley.

Eckart, C., & Young, G. (1936). The approximation of one matrix by another of lower rank. *Psychometrika*, *1*(3), 211–218. https://doi.org/10.1007/BF02288367

Eryol, E., & Vural, F. T. Y. (2022a). Hierarchical Feature Alignment for Transfer Learning on Neural Decoding Tasks, In *BIBE*, IEEE.

Eryol, E., & Vural, F. T. Y. (2022b). Template-aligned Transfer Learning on Brain Decoding Problem, In *IEEE SIU*.

Esteban, O., Blair, R., Markiewicz, C. J., Berleant, S. L., Moodie, C., Ma, F., Isik, A. I., Erramuzpe, A., Kent, M., James D. andGoncalves, DuPre, E., Sitek, K. R., Gomez, D. E. P., Lurie, D. J., Ye, Z., Poldrack, R. A., & Gorgolewski, K. J. (2018). fMRIPrep [Publisher: Zenodo]. *Software*. https://doi.org/10.5281/zenodo.852659

Esteban, O., Ciric, R., Finc, K., Blair, R. W., Markiewicz, C. J., Moodie, C. A., Kent, J. D., Goncalves, M., DuPre, E., Gomez, D. E. P., Ye, Z., Salo, T., Valabregue, R., Amlien, I. K., Liem, F., Jacoby, N., Stojić, H., Cieslak, M., Urchs, S., ... Gorgolewski, K. J. (2020). Analysis of task-based functional MRI data preprocessed with fMRIPrep. *Nat Protoc*, *15*(7), 2186–2202. https://doi.org/10.1038/s41596-020-0327-3

Fayyaz, M., & Gall, J. (2020). SCT: Set Constrained Temporal Transformer for Set Supervised Action Segmentation, In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, Seattle, WA, USA, IEEE. https://doi.org/10.1109/CVPR42600.2020.00058

Frégnac, Y. (2017). Big data and the industrialization of neuroscience: A safe roadmap for understanding the brain? *Science (80-.).*, *358*(6362), 470–477. https://doi.org/10.1126/science.aan8866

Fukunaga, K., & Koontz, W. (1970). Application of the Karhunen-Loève Expansion to Feature Selection and Ordering [Conference Name: IEEE Transactions on Computers]. *IEEE Transactions on Computers*, *C-19*(4), 311–318. https://doi.org/10.1109/T-C.1970.222918

Gao, Y. (2019). Decoding Brain States from fMRI Signals by using Unsupervised Domain Adaptation [Publisher: IEEE]. *IEEE J. Biomed. Heal. Informatics*, *PP*(100), 1. https://doi.org/10.1109/JBHI.2019.2940695

Geirhos, R., Michaelis, C., Zemel, R., Brendel, W., Bethge, M., & Wichmann, F. A. (2020). Shortcut Learning in Deep Neural Networks. *Nat. Mach. Intell.*, 1–29.

Ghojogh, B., & Crowley, M. (2019). Eigenvalue and Generalized Eigenvalue Problems : Tutorial [arXiv: arXiv:1903.11240v1]. *Arxiv*, (2), 1–8.

Glasser, M. F., Coalson, T. S., Robinson, E. C., Hacker, C. D., Harwell, J., Yacoub, E., Ugurbil, K., Andersson, J., Beckmann, C. F., Jenkinson, M., Smith, S. M., & Van Essen, D. C. (2016). A multi-modal parcellation of human cerebral cortex. *Nature*, *536*(7615), 171–178. https://doi.org/10.1038/nature18933

Gorgolewski, K. J., Varoquaux, G., Rivera, G., Schwarz, Y., Ghosh, S. S., Maumet, C., Sochat, V. V., Nichols, T. E., Poldrack, R. A., Poline, J.-B., Yarkoni, T., & Margulies, D. S. (2015). NeuroVault.Org: A web-based repository for collecting and sharing unthresholded statistical maps of the human brain. *Front. Neuroinform.*, *9*(APR), 1–9. https://doi.org/10.3389/fninf.2015.00008

Gretton, A., Bousquet, O., Smola, A., & Sch, B. (2005). Measuring Statistical Dependence with Hilbert-Schmidt Norms, In *Algorithmic Learning Theory*.

Habeeb, H., & Koyejo, O. (2020). Towards a deep network architecture for structured smoothness, In *ICLR*.

Hanke, M., & Neumann, M. (1993). The geometry of the set of scaled projections. *Linear Algebra and its Applications*, *190*, 137–148. https://doi.org/10.1016/0024-3795(93)90223-B

Haxby, J. V., Guntupalli, J. S., Connolly, A. C., Halchenko, Y. O., Conroy, B. R., Gobbini, M. I., Hanke, M., & Ramadge, P. J. (2011). A Common, High-Dimensional Model of the Representational Space in Human Ventral Temporal Cortex. *Neuron*, *72*(2), 404–416. https://doi.org/10.1016/j.neuron.2011.08.026

Hernández-garcía, A., & König, P. (2016). Data augmentation instead of explicit regularization. *preprint*arXiv arXiv:1806.03852v4.

Higgins, I., Matthey, L., Pal, A., Burgess, C., Glorot, X., Botvinick, M., Mohamed, S., & Lerchner, A. (2017). Beta-VAE: Learning Basic Visual Concepts with a Constrained Variational Framework, 22.

Hinton, G. E., Srivastava, N., Krizhevsky, A., Sutskever, I., & Salakhutdinov, R. R. (2012). Improving neural networks by preventing co-adaptation of feature detectors [arXiv:1207.0580 [cs]]. arXiv. https://doi.org/10.48550/arXiv.1207.0580

Hotelling, H. (1936). Relations Between Two Sets of Variates. *Biometrika*, *28*(3/4), 321–377. https://doi.org/10.2307/2333955

Hu, W. (2019). The Surprising Simplicity of the Early-Time Learning Dynamics of Neural Networks [arXiv: arXiv:2006.14599v1], In *ArXiv Prepr.* arXiv: arXiv:2006.14599v1.

Ioffe, S., & Szegedy, C. (2015). Batch Normalization : Accelerating Deep Network Training by Reducing Internal Covariate Shift [arXiv: arXiv:1502.03167v3]. *ICML*.

Karakasis, P. A., Liavas, A. P., Sidiropoulos, N. D., Simos, P. G., & Papadaki, E. (2022). Multisubject Task-Related fMRI Data Processing via a Two-Stage Generalized Canonical Correlation Analysis [Conference Name: IEEE Transactions on Image Processing]. *IEEE Transactions on Image Processing*, *31*, 4011–4022. https://doi.org/10.1109/TIP.2022.3159125

Kettenring, J. R. (1971). Canonical analysis of several sets of variables. *Biometrika*.

Kim, J. (2018). Regional Attention Based Deep Feature for Image Retrieval, In *Bmvc*.

Kingma, D. P., & Ba, J. L. (2015). Adam: A method for stochastic optimization [ISSN: 13087711 arXiv: 1412.6980], In *3rd Int. Conf. Learn. Represent. ICLR 2015 - Conf. Track Proc.* ISSN: 13087711 arXiv: 1412.6980. https://doi.org/10.1063/1.4902458

Kingma, D. P., & Welling, M. (2014). Auto-encoding variational bayes [_eprint: 1312.6114]. *2nd Int. Conf. Learn. Represent. ICLR 2014 - Conf. Track Proc.*, (1050), 1–14.

Koles, Z. J., Lazar, M. S., & Zhou, S. Z. (1990). Spatial patterns underlying population differences in the background EEG. *Brain Topogr*, *2*(4), 275–284. https://doi.org/10.1007/BF01129656

Kong, D., Liu, J., Liu, B., & Bao, X. (2016). Uncorrelated Group LASSO, In *Aaai*.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). ImageNet Classification with Deep Convolutional Neural Networks [event-place: Red Hook, NY, USA], In *Proc. 25th Int. Conf. Neural Inf. Process. Syst. - Vol. 1*, Curran Associates Inc. event-place: Red Hook, NY, USA.

Krzanowski, W. J. (1979). Between-Groups Comparison of Principal Components. *Journal of the American Statistical Association*, *74*(367), 703–707. https://doi.org/10.2307/2286995

Li, B., Wu, F., Weinberger, K. Q., & Belongie, S. (2019). Positional Normalization, (NeurIPS), 1–13.

Lin, G., Shen, C., Hengel, A. V. D., & Reid, I. (2016). Efficient Piecewise Training of Deep Structured Models for Semantic Segmentation, In *CVPR*.

Liu, R., Lehman, J., Molino, P., Such, F. P., Frank, E., Sergeev, A., & Yosinski, J. (2018). An intriguing failing of convolutional neural networks and the CoordConv solution [arXiv: 1807.03247]. *Adv. Neural Inf. Process. Syst.*, *2018-December*, 9605–9616.

Liu, R., Li, Y., Tao, L., Liang, D., & Zheng, H.-T. (2022). Are we ready for a new paradigm shift? A survey on visual deep MLP. *Patterns*, *3*(7), 100520. https://doi.org/10.1016/j.patter.2022.100520

Marek, S., Tervo-Clemmens, B., Calabro, F. J., Montez, D. F., Kay, B. P., Hatoum, A. S., Donohue, M. R., Foran, W., Miller, R. L., Hendrickson, T. J., Malone, S. M., Kandala, S., Feczko, E., Miranda-Dominguez, O., Graham, A. M., Earl, E. A., Perrone, A. J., Cordova, M., Doyle, O., … Dosenbach, N. U. F. (2022). Reproducible brain-wide association studies require thousands of individuals. *Nature*. https://doi.org/10.1038/s41586-022-04492-9

Markiewicz, C. J., Gorgolewski, K. J., Feingold, F., Blair, R., Halchenko, Y. O., Miller, E., Hardcastle, N., Wexler, J., Esteban, O., Goncavles, M., Jwa, A., & Poldrack, R. (2021). The OpenNeuro resource for sharing of neuroscience data. *eLife*, *10*, e71774. https://doi.org/10.7554/eLife.71774

McClure, P., Moraczewski, D., Lam, K. C., Thomas, A., & Pereira, F. (2020). Improving the Interpretability of fMRI Decoding using Deep Neural Networks and Adversarial Robustness [arXiv:2004.11114 [cs, q-bio, stat] version: 3]. arXiv. https://doi.org/10.48550/arXiv.2004.11114

Mitchell, T. M. (1997). *Machine Learning*. Mcgraw-Hill. Retrieved April 24, 2023, from http://www.cs.cmu.edu/~tom/mlbook.html

Muzio, B. D. (2022). Temporal pole | Radiology Reference Article | Radiopaedia.org. https://doi.org/10.53347/rID-34748

Nastase, S. A., Liu, Y.-F., Hillman, H., Norman, K. A., & Hasson, U. (2020). Leveraging shared connectivity to aggregate heterogeneous datasets into a common response space. *NeuroImage*, *217*, 116865. https://doi.org/10.1016/j.neuroimage.2020.116865

Norouzi, S. (2022). Structured DropConnect for Convolutional Neural Networks, In *ICIP*.

Ojala, M., & Garriga, G. C. (2009). Permutation Tests for Studying Classifier Performance [event-place: Miami Beach, FL, USA], In *2009 Ninth IEEE Inter-*

national Conference on Data Mining, IEEE. event-place: Miami Beach, FL, USA. https://doi.org/10.1109/ICDM.2009.108

Onal, I., Ozay, M., Mizrak, E., Oztekin, I., Vural, F. T., Member, S., Ozay, M., Mizrak, E., & Oztekin, I. (2017). A new representation of fMRI signal by a set of local meshes for brain decoding. *IEEE Trans. Signal Inf. Process. over Networks*, *3*(4), 683–694. https://doi.org/10.1109/TSIPN.2017.2679491

Pearson, K. (1901). LIII. On lines and planes of closest fit to systems of points in space. *The London, Edinburgh, and Dublin Philosophical Magazine and Journal of Science*, *2*(11), 559–572. https://doi.org/10.1080/14786440109462720

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., & Cournapeau, D. (2011). Scikit-learn: Machine Learning in Python. *JMLR*, 6.

Poldrack, R. A., & Gorgolewski, K. J. (2017). OpenfMRI: Open sharing of task fMRI data [ISBN: 1053-8119]. *Neuroimage*, *144*, 259–261. https://doi.org/10.1016/j.neuroimage.2015.05.073

Poldrack, R. A., Nichols, T., & Mumford, J. (2011). *Handbook of Functional MRI Data Analysis*. Cambridge, Cambridge University Press. https://doi.org/10.1017/CBO9780511895029

Qiao, S., Wang, H., Liu, C., Shen, W., & Yuille, A. (2020). Micro-Batch Training with Batch-Channel Normalization and Weight Standardization [arXiv:1903.10520 [cs]]. arXiv. https://doi.org/10.48550/arXiv.1903.10520

Rastogi, P., Van Durme, B., & Arora, R. (2015). Multiview LSA: Representation Learning via Generalized CCA [event-place: Denver, Colorado], In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Association for Computational Linguistics. event-place: Denver, Colorado. https://doi.org/10.3115/v1/N15-1058

Rustamov, R. M., & Guibas, L. (2016). Hyperalignment of Multi-subject fMRI Data by Synchronized Projections (I. Rish, G. Langs, L. Wehbe, G. Cecchi, K.-m. K. Chang, & B. Murphy, Eds.). In I. Rish, G. Langs, L. Wehbe, G. Cecchi, K.-m. K. Chang, & B. Murphy (Eds.), *Machine Learning and Interpretation in Neuroimaging*, Cham, Springer International Publishing. https://doi.org/10.1007/978-3-319-45174-9_12

Rybkin, O., Daniilidis, K., & Levine, S. (2021). Simple and Effective VAE Training with Calibrated Decoders [arXiv:2006.13202 [cs, eess, stat]]. arXiv. Retrieved December 6, 2022, from http://arxiv.org/abs/2006.13202

Savostyanov, D. (2014). Linear algebra - Efficient way to find SVD of sum of projection matrices? [Publication Title: MathOverflow]. Retrieved March 10, 2022, from https://mathoverflow.net/questions/178562/efficient-way-to-find-svd-of-sum-of-projection-matrices

Saxe, A., Nelli, S., & Summerfield, C. (2020). If deep learning is the answer, then what is the question? [arXiv:2004.07580 [q-bio]]. arXiv. https://doi.org/10.48550/arXiv.2004.07580

Scardapane, S., Comminiello, D., Hussain, A., & Uncini, A. (2017). Group sparse regularization for deep neural networks. *Neurocomputing*, 1–9. https://doi.org/10.1016/j.neucom.2017.02.029

Schaefer, A., Kong, R., Gordon, E. M., Laumann, T. O., Zuo, X.-N., Holmes, A. J., Eickhoff, S. B., & Yeo, B. T. T. (2018). Local-Global Parcellation of the Human Cerebral Cortex from Intrinsic Functional Connectivity MRI. *Cereb Cortex*, *28*(9), 3095–3114. https://doi.org/10.1093/cercor/bhx179

Schönemann, P. H., & Carroll, R. M. (1970). Fitting one matrix to another under choice of a central dilation and a rigid motion. *Psychometrika*, *35*(2), 245–255. https://doi.org/10.1007/BF02291266

Seo, J.-W., & Kim, S. D. (2013). Novel PCA-Based Color-to-Gray Image Conversion. *ICIP*, 5.

Smilkov, D., Thorat, N., Kim, B., & Vi, F. (2017). SmoothGrad : removing noise by adding noise, In *ArXiv*.

Sohn, K., Lee, H., & Yan, X. (2015). Learning Structured Output Representation using Deep Conditional Generative Models. *NeurIPS*, 9.

Sudlow, C., Gallacher, J., Allen, N., Beral, V., Burton, P., Danesh, J., Downey, P., Elliott, P., Green, J., Landray, M., Liu, B., Matthews, P., Ong, G., Pell, J., Silman, A., Young, A., Sprosen, T., Peakman, T., & Collins, R. (2015). UK Biobank: An Open Access Resource for Identifying the Causes of a Wide Range of Complex Diseases of Middle and Old Age [Publisher: Public Library of Science]. *PLOS Medicine*, *12*(3), e1001779. https://doi.org/10.1371/journal.pmed.1001779

Sun, Z., Qiao, Y., Lelieveldt, B. P. F., & Staring, M. (2019). Integrating spatial-anatomical regularization and structure sparsity into SVM : Improving interpretation of Alzheimer ' s disease classi fi cation. *Neuroimage*, *178*(May 2018), 445–460. https://doi.org/10.1016/j.neuroimage.2018.05.051

Sutoko, S., Atsumori, H., Obata, A., Funane, T., Kandori, A., Shimonaga, K., Hama, S., Yamawaki, S., & Tsuji, T. (2020). Lesions in the right Rolandic operculum are associated with self-rating affective and apathetic depressive symptoms for post-stroke patients [Number: 1 Publisher: Nature Publishing Group]. *Sci Rep*, *10*(1), 20264. https://doi.org/10.1038/s41598-020-77136-5

Thomas, A. W., & Samek, W. (2019). Deep Transfer Learning For Whole-Brain fMRI Analyses [arXiv: arXiv:1907.01953v1]. *Arxiv*, *100*, 1–8.

Tolstikhin, I., Houlsby, N., Kolesnikov, A., Beyer, L., Zhai, X., Unterthiner, T., Yung, J., Steiner, A., Keysers, D., Uszkoreit, J., Lucic, M., & Dosovitskiy, A. (2021). MLP-Mixer: An all-MLP Architecture for Vision. *arXiv: 2105.01601 [cs]*. Retrieved November 1, 2021, from http://arxiv.org/abs/2105.01601

Turner, B. O., Paul, E. J., Miller, M. B., & Barbey, A. K. (2018). Small sample sizes reduce the replicability of task-based fMRI studies [Publisher: Springer US]. *Commun. Biol.*, *1*(1). https://doi.org/10.1038/s42003-018-0073-z

Tzourio-Mazoyer, N., Landeau, B., Papathanassiou, D., Crivello, F., Etard, O., Delcroix, N., Mazoyer, B., & Joliot, M. (2002). Automated anatomical labeling of activations in SPM using a macroscopic anatomical parcellation of the MNI MRI single-subject brain. *Neuroimage*, *15*(1), 273–289. https://doi.org/10.1006/nimg.2001.0978

Ulyanov, D., Vedaldi, A., & Lempitsky, V. (2017). Instance Normalization: The Missing Ingredient for Fast Stylization [arXiv:1607.08022 [cs]]. arXiv. https://doi.org/10.48550/arXiv.1607.08022

Van Essen, D. C., Smith, S. M., Barch, D. M., Behrens, T. E. J., Yacoub, E., Ugurbil, K., & WU-Minn HCP Consortium. (2013). The WU-Minn Human Connectome Project: an overview. *Neuroimage*, *80*, 62–79. https://doi.org/10.1016/j.neuroimage.2013.05.041

Wang, X., Girshick, R., Gupta, A., & He, K. (2018). Non-local Neural Networks. *CVPR*, 7794–7803. https://doi.org/10.1109/CVPR.2018.00813

Wang, X. (2020). Decoding and mapping task states of the human brain via deep learning. *Hum Brain Mapp*, *41*(6), 1505–1519. https://doi.org/10.1002/hbm.24891

Wang, X., Shan, S., & Chen, X. (2019). Fully Learnable Group Convolution for Acceleration of Deep Neural Networks [arXiv: arXiv:1904.00346v1], In *CVPR*. arXiv: arXiv:1904.00346v1.

Westfall, J., Nichols, T. E., & Yarkoni, T. (2017). Fixing the stimulus-as-fixed-effect fallacy in task fMRI. *Wellcome Open Res*, *1*, 23. https://doi.org/10.12688/wellcomeopenres.10298.2

Wikipedia. (2022). Visceral sensation | Encyclopedia.com. Retrieved December 12, 2022, from https://www.encyclopedia.com/medicine/encyclopedias-almanacs-transcripts-and-maps/visceral-sensation

Wu, Q., Hong, D., & Zou, J. (2016). Spatial Regularization for Neural Network and Application in Alzheimer's Disease Classification [Issue: December], In *Futur. Technol. Conf.* Issue: December.

Wu, R., & Kamata, S.-i. (2018). Sparse Graph Based Deep Learning Networks for Face Recognition. *IEICE TRANS. INF. SYST.*, (9), 2209–2219.

Wu, Y., & He, K. (2018). Group Normalization [arXiv:1803.08494 [cs]]. arXiv. https://doi.org/10.48550/arXiv.1803.08494

Xue, G., Aron, A. R., & Poldrack, R. A. (2008). Common Neural Substrates for Inhibition of Spoken and Manual Responses. *Cerebral Cortex*, *18*(8), 1923–1932. https://doi.org/10.1093/cercor/bhm220

Yosinski, J., Clune, J., Bengio, Y., & Lipson, H. (2014). How transferable are features in deep neural networks? [arXiv: 1411.1792]. *Adv. Neural Inf. Process. Syst.*, *4*(January), 3320–3328.

Yousefnezhad, M., Selvitella, A., Han, L., & Zhang, D. (2021). Supervised Hyperalignment for multi-subject fMRI data alignment. *IEEE Trans. Cogn. Dev. Syst.*, *13*(3), 475–490. https://doi.org/10.1109/TCDS.2020.2965981

Yousefnezhad, M., Selvitella, A., Zhang, D., Greenshaw, A. J., & Greiner, R. (2020). Shared Space Transfer Learning for analyzing multi-site fMRI data [Issue: NeurIPS], In *NeurIPS*. Issue: NeurIPS.

Yousefnezhad, M., & Zhang, D. (2017). Anatomical Pattern Analysis for decoding visual stimuli in human brains [arXiv: 1710.02113]. *arXiv:1710.02113 [cs, q-bio, stat]*. Retrieved December 4, 2021, from http://arxiv.org/abs/1710.02113

Zhang, H., Chen, P.-H., Chen, J., Zhu, X., Turek, J. S., Willke, T. L., Hasson, U., & Ramadge, P. J. (2016). A Searchlight Factor Model Approach for Locating Shared Information in Multi-Subject fMRI Analysis [arXiv:1609.09432 [cs, q-bio, stat]]. arXiv. https://doi.org/10.48550/arXiv.1609.09432

Zhang, H., Chen, P.-H., & Ramadge, P. J. (2018). Transfer Learning on fMRI Datasets, In *AISTATS*.

Zhang, Y., Tetrel, L., Thirion, B., Bellec, P., Annotation, F., States, H. C., & Convolution, D. G. (2020). Functional Annotation of Human Cognitive States using Deep Graph Convolution. *bioRxiv Prepr.*, 1–57.

Zhang, Z., Li, J., Shao, W., Peng, Z., Zhang, R., Wang, X., & Luo, P. (2019). Differentiable Learning-to-Group Channels via Groupable Convolutional Neural Networks [arXiv: arXiv:1908.05867v2], In *CVPR*. arXiv: arXiv:1908.05867v2.

Zhou, S., Li, W., Cox, C. R., & Lu, H. (2018). Domain Independent SVM for Transfer Learning in Brain Decoding [arXiv: arXiv:1903.11020v1], In *Arxiv*. arXiv: arXiv:1903.11020v1.

CURRICULUM VITAE

PERSONAL INFORMATION

Surname, Name: Eryol, Erkin
Nationality: Turkish (TC)
Marital Status: Single

EDUCATION

Degree	Institution	Year of Graduation
Ph.D.	METU - Computer Engineering	2023
M.S.	METU - Computer Engineering	2010
B.S.	Başkent University - Computer Engineering	2007
High School	Çankaya Atatürk Anatolian High School	2002

PROFESSIONAL EXPERIENCE

Year	Place	Enrollment
2018-2023	METU-Council of Higher Education	Scholarship/AI research grant
2012-2017	METU	Research assistant
2011-2012	METU HASAT Project	Project specialist
2010-2011	Bilkent University	Research assistant
2008-2010	Orbim-METU	Software programmer

PUBLICATIONS

National Conference Publications

E. Eryol, F. T. Y. Vural, “Template-aligned Transfer Learning on Brain Decoding Problem,” in IEEE SIU, 2022.

International Conference Publications

E. Eryol and F. T. Y. Vural, “Hierarchical Feature Alignment for Transfer Learning on Neural Decoding Tasks,” in IEEE BIBE, 2022.