Université de Montréal

Brain decoding of the Human Connectome Project Tasks in a Dense Individual fMRI Dataset

par

Shima Rastegarnia

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Mémoire présenté en vue de l'obtention du grade de
Maître ès sciences (M.Sc.) en Informatique

Novembre 2022

Université de Montréal

Département d'informatique et de recherche opérationnelle
Faculté des arts et des sciences

Ce mémoire intitulé

Brain decoding of the Human Connectome Project Tasks in a Dense Individual fMRI Dataset

présenté par

Shima Rastegarnia

a été évalué par un jury composé des personnes suivantes :

Irina Rish

(président-rapporteur)

Pierre-Louis Bellec

(directeur de recherche)

Guillaume Lajoie

(membre du jury)

Résumé

Les études de décodage cérébral visent à entrainer un modèle d'activité cérébrale qui reflète l'état cognitif du participant. Des variations interindividuelles substantielles dans l'organisation fonctionnelle du cerveau représentent un défi pour un décodage cérébral précis. Dans cette thèse, nous évaluons si des modèles de décodage cérébral précis peuvent être entrainés avec succès entièrement au niveau individuel.

Nous avons utilisé un ensemble de données individuel dense d'imagerie par résonance magnétique fonctionnelle (IRMf) pour lequel six participants ont terminé l'ensemble de la batterie de tâches du “Human Connectome Project” > 13 fois sur dix sessions d'IRMf distinctes. Nous avons implémenté plusieurs méthodes de décodage, des simples machines à vecteurs de support aux réseaux complexes de neurones à convolution de graphes. Tous les décodeurs spécifiques à l'individu ont été entrainés pour classifier simultanément les volumes d'IRMf simples (TR = 1,49) entre 21 conditions expérimentales, en utilisant environ sept heures de données d'IRMf par participant.

Les meilleurs résultats de prédiction ont été obtenus avec notre modèle de machine à vecteurs de support avec une précision de test allant de 64 à 79 % (niveau de la chance environ 7%). Les perceptrons multiniveaux et les réseaux convolutionnels de graphes ont également obtenu de très bons résultats (63-78% et 63-77%, respectivement). Les cartes d'importance des caractéristiques dérivées du meilleur modèle (SVM) ont révélé que la classification utilise des régions pertinentes pour des domaines cognitifs particuliers, sur la base d’a priori neuro-anatomique. En appliquant un modèle individuel aux données d’un autre sujet (classification inter-sujets), on observe une précision nettement inférieure à celle des modèles spécifiques au sujet, ce qui indique que les décodeurs cérébraux individuels ont appris des caractéristiques spécifiques à chaque individu.

Nos résultats indiquent que des ensembles de données de neuroimagerie profonde peuvent être utilisés pour former des modèles de décodage cérébral précis au niveau individuel. Les données de cette étude sont partagées librement avec la communauté (https://cneuromod.ca), et pourront servir de benchmark de référence, pour l’entrainement de modèles de décodage cérébral individuel, ou bien des études de “transfert learning” à partir de l’échantillon collecté par le human connectome project.

Mots clés: Intelligence Artificielle, Apprentissage automatique, IRMf, Décodage cérébral, États cognitifs

Abstract

Brain decoding studies aim to train a pattern of brain activity that reflects the cognitive state of the participant. Substantial inter-individual variations in functional organization represent a challenge to accurate brain decoding. In this thesis, we assess whether accurate brain decoding models can be successfully trained entirely at the individual level.

We used a dense individual functional magnetic resonance imaging (fMRI) dataset for which six participants completed the entire Human Connectome Project (HCP) task battery>13 times across ten separate fMRI sessions. We assessed several decoding methods, from simple support vector machines to complex graph convolution neural networks. All individual-specific decoders were trained to classify single fMRI volumes (TR = 1.49) between 21 experimental conditions simultaneously, using around seven hours of fMRI data per participant.

The best prediction accuracy results were achieved with our support vector machine model with test accuracy ranging from 64 to 79% (chance level of about 7%). Multilevel perceptrons and graph convolutional networks also performed very well (63-78% and 63-77%, respectively). Best Model Derived Feature Importance Maps (SVM) revealed that the classification uses regions relevant to particular cognitive domains, based on neuroanatomical priors. Applying an individual model to another subject's data (across-subject classification) yields significantly lower accuracy than subject-specific models, indicating that individual brain decoders have learned characteristics specific to each individual.

Our results indicate that deep neuroimaging datasets can be used to train accurate brain decoding models at the individual level. The data from this study is shared freely with the community (https://cneuromod.ca) and can be used as a reference benchmark, for training individual brain decoding models, or for “transfer learning” studies from the sample collected by the human connectome project.

Keywords: Artificial Intelligence, Machine Learning, fMRI, Brain Decoding, Cognitive States

Contents

List of Tables

List of Figures

List of Acronyms and Abbreviations

IRMf	*Imagerie par Résonance Magnétique fonctionnelle*
fMRI	*functional Magnetic Resonance Imaging*
TR	*Repetition Time*
AI	*Artificial Intelligence*
HCP	*Human Connectome Project*
hcptrt	*HCP test-retest dataset*
CNeuroMod	*Courtois project on Neural Modelling*
EEG	*Electroencephalography*
MEG	*Magnetoencephalography*
CT	*computed tomography*
MRI	*Magnetic Resonance Imaging*
fNIRS	*Functional Near-Infrared Spectroscopy*
BOLD	*Blood-Oxygen-Level-Dependent*
HRF	*Hemodynamic Response Function*
MNI	*Montreal Neurological Institute*
ROI	*Region of Interest*
MVPA	*Multi-Voxel Pattern Classification*
GCN	*Graph Convolutional Network model*
MSC	*Midnight Scan Club*
NSD	*Natural Scenes Dataset*
SVM	*Support Vector Machine*
MLP	*Multilayer Perceptron*
RBF	*Radial Basis Function*
CMRR	*Center for Magnetic Resonance Research*
C2P	*Concept to Production*
MIST	*Multiresolution Intrinsic Segmentation Template*
DiFuMo	*Dictionaries of Functional Modes*
gwMRF	*gradient-weighted Markov Random Field*
RFC	*Random Forest Classifier*
k-NN	*k-Nearest-Neighbor*
GLM	*first-level analysis of the HRF*
WM	*Working Memory*

Gratitude

I would like to acknowledge a number of people who have shaped my research trajectory over the last few years:

I would like to express my deep gratitude to my supervisor, Dr. Pierre Bellec, for giving me the chance to work as a member of the CNeuroMod project and initiate exploration in the field of Neuro-AI. I appreciate all his invaluable mentorship and beyond for supporting me at the time that I needed it.

I would like to thank my family, in particular, my husband and mother for their unconditional love and support in every aspect of my life.

I would also like to thank all the professors who mentored me during my undergraduate and helped me broaden my research perspective.

Chapter 1

Introduction

1.1 Background

The human brain has a complex computing structure built of billions of neurons as computing segments. These neurons form several organized components that are responsible for particular processes which conduct our thoughts, actions, and emotions. The question of how brain functionality leads to human intelligent behavior has attracted attention over the centuries, and despite several challenges, we are getting closer to answering this question. Advances in brain imaging technology have enabled scientists to study the brain in much greater detail. In parallel, statistical methods have been systematically developed for analyzing functional brain imaging data. Taken together, these advances have made it possible to achieve a better understanding of the link between brain functional architecture and behavior.

A common approach to studying the neural substrates of behavior is known as “brain decoding”. Brain decoding consists of predicting stimulus features with a measure of brain activity. Brain decoding is a promising and active field of research, which has benefited greatly from the development of new Artificial Intelligence (AI) paradigms (Figure 1.1). Modern brain imaging technologies make it possible to measure and record data of human brain activity non-invasively under various conditions. These techniques can be beneficial not only for a better understanding of the brain's functional architecture but also for diagnosing certain brain-related medical conditions. The relations between different brain systems and various computational approaches open a wide and inspiring research avenue for computer and neuroinformatics scientists.

So far the majority of brain decoding studies have been focused on group-level brain decoding approaches which aim to generalize brain decoding methods to data from new, previously unseen subjects. Generalization to new subjects is very challenging, because of the well-known variations of brain structure, function as well as behavior across

individuals. For this reason, a recent trend in research is to train brain decoding models entirely at the level of individual subjects, provided sufficiently large data is available. This Master's project aims at assessing the performance of such individual-level training, using a new large benchmark of decoding tasks that have been comprehensively studied at the individual level.

Brain decoding techniques infer cognitive states from snapshots of brain activity captured with neuroimaging tools such as functional magnetic resonance imaging (fMRI), magneto-, or electroencephalography (M/EEG). Significant recent milestones for group-level decoding include the decoding of natural language semantic features from non-invasive brain recordings data. Such decoding results could lead to scalable technologies with wide adoption and represent a promising path for developing safe and reliable devices with real-world usages, such as helping people with communication problems (de Projecto n.d.; Shen et al. 2022). In a recent study (Défossez et al. 2022), a convolutional neural network method was trained simultaneously across a large cohort of individuals to predict self-supervised representations of natural speech from a time series of high-dimensional brain signals. The model was assessed using four publicly available M/EEG datasets totaling data recorded from 169 healthy participants who passively listened to the speech. In another research (Tang et al. 2022), a decoder was trained to reconstruct hearing or imagining continuous natural language stimuli using fMRI scans. The dataset consisted of three subjects, each listening to sixteen hours of naturally spoken narrative stories. Decoding continuous natural language using fMRI scans is a challenging task considering the nature of the human brain's blood-oxygen-level-dependent (BOLD) signal (section 1.3.1), which is intrinsically slow (less than one brain image per second), and the speed of naturally spoken English (over two words per second). Natural language decoding, therefore, is a hard task because a single recorded brain volume may mix the response to several word stimuli. The authors overcome this obstacle by retaining a limited set of plausible candidate word sequences, based on a language generative model. The most likely word sequence is then selected by assessing their respective performance as stimuli with a brain encoding model. This procedure is used to score the likelihood that the subject is hearing or imagining each sequence candidate, and select the best one. The results show that using the word sequences for decoding not only captures the meaning of the words but often recovers the exact stimuli presented to the subject, including full correct sentences.

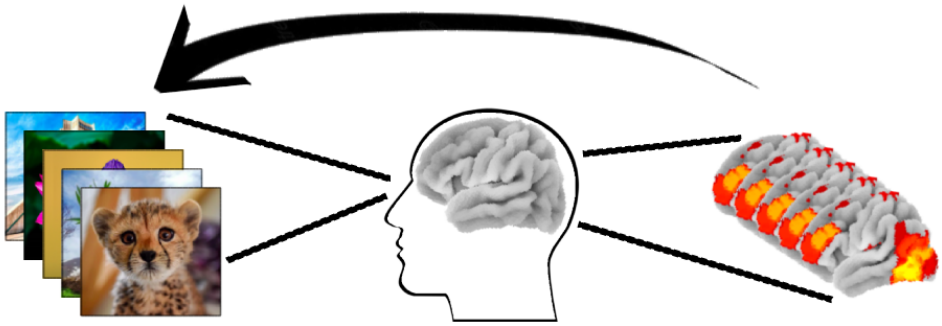


Figure 1.1. Brain decoding schematic view.

Despite impressive advances in deep learning methods for brain decoding at the group level, group-level brain decoding is still challenging and typically requires very large datasets to successfully train models. These challenges likely originate from large inter-subject variability in both brain structure and function while performing specific cognitive tasks (Gordon et al. 2017; Gratton et al. 2018). In group-level brain decoding studies of human functional MRI, the dominant approach is based on analyzing data averaged across groups of subjects, which causes a loss of individual-specific brain features. Therefore, a domain-general decoder that can classify several different cognitive states simultaneously suffers from the limitation of fMRI data size and very large-scale brain imaging datasets are essential (Y. Zhang et al. 2021). These limitations make between-subject decoding likely more complex than individual-level decoding (Porter et al. 2022).

An alternative approach to training classifiers on datasets derived from multiple individuals is to use deep phenotyping datasets, which rely on a limited number of individuals who get scanned intensively with a wide range of stimuli. Training decoders directly on individual participants' brain activity patterns, allows us to assess the relative success of decoding fine-scaled functional features. Importantly, training such a decoder critically depends on the availability of sufficient data for individual participants. An extensive sampling of neural activity is only feasible for a limited number of participants that can be scanned extensively and reliably for a long period of time. We are aware of only one published fMRI study so far attempting to train brain decoders at the individual level.

In (Porter et al. 2022) a ridge regression classifier was used for brain decoding at the individual level using Midnight Scan Club (MSC) fMRI dataset (Gordon et al. 2017). The goal of this study was to investigate to what extent individual differences affect functional state prediction and how brain networks vary during different task states. The MSC dataset

contains over 10 h of fMRI data per participant (N=10) across five different task states (Resting state, Motor task, Semantic task, Coherence task, and Memory task). In this research, machine learning models were trained on a subset of the dataset and then were tested on either held-out data from the same person or a new participant. The results show that the classification performance was significantly higher in the within-subject approach than in the similar between-subject experiment. These results suggest that the within-subject approach can produce reliable network maps based on individual-specific characteristics which reveal robust features of individual brain states and confirm that individual brain decoding studies have the potential to expand our knowledge of brain connectivities.

An important limitation of the Porter et al. (2022) study is that it is difficult to contrast performance directly between individually-trained and group-trained brain decoders due to the rarity of corresponding datasets. For example, we are unaware of any fMRI dataset featuring a large group of participants presented with the same set of tasks or stimuli as MSC (N=10). The group-level decoder was trained on N=10 subjects only, which is likely insufficient to successfully generalize across subjects even in the presence of strong signals. Deep phenotyping datasets are becoming increasingly available for specific domains such as vision (e.g., the Natural Scenes Dataset (NSD; N=8; Allen et al. 2022), movie watching (Doctor Who; N=1; Seeliger et al. 2019, 2021), and for a variety of tasks and conditions in collections such as the Individual Brain Charting dataset (IBC; N=10; Pinho et al. 2020), and the Courtois Project on Neuronal Modelling (CNeuroMod; N=6 (“Datasets — Courtois NeuroMod 2020-Beta Documentation” n.d.). These deep datasets for the most part suffer from the same limitation as MSC, in that there is no existing large-scale group data to compare the performance of individual vs group brain decoding models and the datasets feature too few subjects to train good group brain decoders (N <= 10). While the IBC sample (N=10) includes the task battery used for the HCP dataset (N=1200), it includes too few repetitions of those tasks (two per subject) to train individual models, which prevents direct comparisons between decoding models trained within and across subjects. For the current project, we trained individual brain decoders using the HCP test-retest (hcptrt) CNeuroMod dataset (“Datasets — Courtois NeuroMod 2020-Beta Documentation” n.d.) for which six participants completed the full HCP task battery (Van Essen et al. 2013) up to 14 times each. The hcptrt is large enough to attempt training purely individual brain decoding models. At the same time, The HCP task battery has been extensively studied as a benchmark for group-level decoding with the HCP sample, which can be used as a comparison point for the performance of individual brain decoding models trained with hcptrt.

1.2. Objectives of the thesis

The principal goal of the current study was to build individual brain decoding models that could accurately decode many different types of cognitive tasks. We aimed to investigate whether high decoding accuracy can be achieved with fewer hours of fMRI training data acquired within-, rather than between-, individual subjects. We trained different classifiers from basic machine learning models such as the Support Vector Machine (SVM; Cortes et al., 1995) to the state-of-the-art deep learning model, Graph Convolutional Neural Networks (GCN; Zhang et al., 2021). All the classifiers were trained on the HCP test-retest (hcptrt) dataset acquired by the Courtois project on Neural Modelling ("Datasets — Courtois NeuroMod 2020-Beta Documentation" n.d.).

With the current project, we aimed to train individual domain-general brain decoding models that could decode many different cognitive tasks to determine whether high decoding accuracy can be achieved with fewer hours of fMRI training data acquired within rather than across individual subjects. Each participant (N=6, 3F | 3M) completed 23 different experimental conditions developed by the Human Connectome Project (HCP) (Van Essen et al. 2013), with 13 to 14 repetitions, totaling 7h 46min to 7h 51min of fMRI data per subject (Table 2.1).

Moreover, we assessed how well individual brain decoders generalize across subjects, and whether they can uncover general markers of cognitive states or are mainly sensitive to idiosyncratic features. To address this question we examined our best-performing model with cross-subject analyses that tested individual models on data acquired from other subjects.

In addition, we generated permutation importance maps from the trained SVM model using the Radial Basis Function (RBF) kernel to assess the homogeneity of decoding features learned across subjects and to evaluate whether decoding was inferred from cognitively relevant brain areas.

We continue this chapter by first covering the necessary background of neuroimaging techniques followed by an overview of the basics of fMRI in section 1.2. We explained the hcptrt data acquisition paradigm, preprocessing stages, and its experimental conditions in 1.3 followed by the brain parcellations approach in 1.4. Section 1.5 covered all the decoding models that were used in this research, and finally, the feature permutation importance is explained in 1.6.

1.3. Neuroimaging Techniques

There exist several brain imaging techniques, including functional and structural methods.

Structural neuroimaging, e.g. computed tomography (CT) and magnetic resonance imaging (MRI) techniques, study the anatomy and morphology of the brain. However, in functional methods brain function is evaluated to help us understand better the correlations between mental functions and the activity in specific areas of the brain (Kasliwal 2013). The most common functional neuroimaging methods are magnetic resonance imaging (fMRI), electroencephalography (EEG), Magnetoencephalography (MEG), and Functional Near-Infrared Spectroscopy (fNIRS) which provide an opportunity to understand the complex human brain cognitive function (Figure 1.2). In particular, fMRI has been developed to study the brain metabolic and vascular reactions to different stimuli.

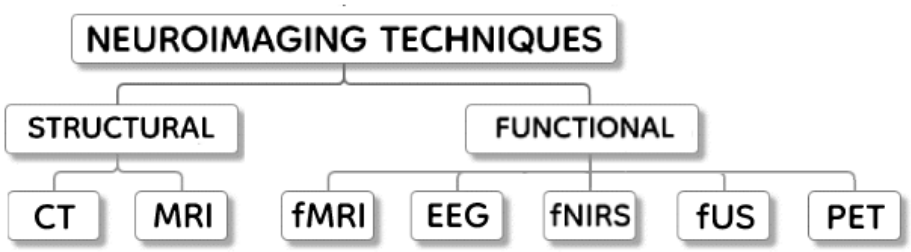


Figure 1.2. Commonly applied structural and functional neuroimaging techniques.

In this chapter, we introduce the preliminary ideas and terminology in functional magnetic resonance imaging (fMRI) to familiarize the reader with the required background knowledge of neuroimaging data in our research.

1.3.1. Functional Magnetic Resonance Imaging

During the last two decades, functional magnetic resonance imaging (fMRI) has become one of the most popular techniques for studying human brain activity. In this method, the brain's functionality is measured by estimating the changes in the local blood oxygenation level, which represent the amount of local brain activity. (Figure 1.3)

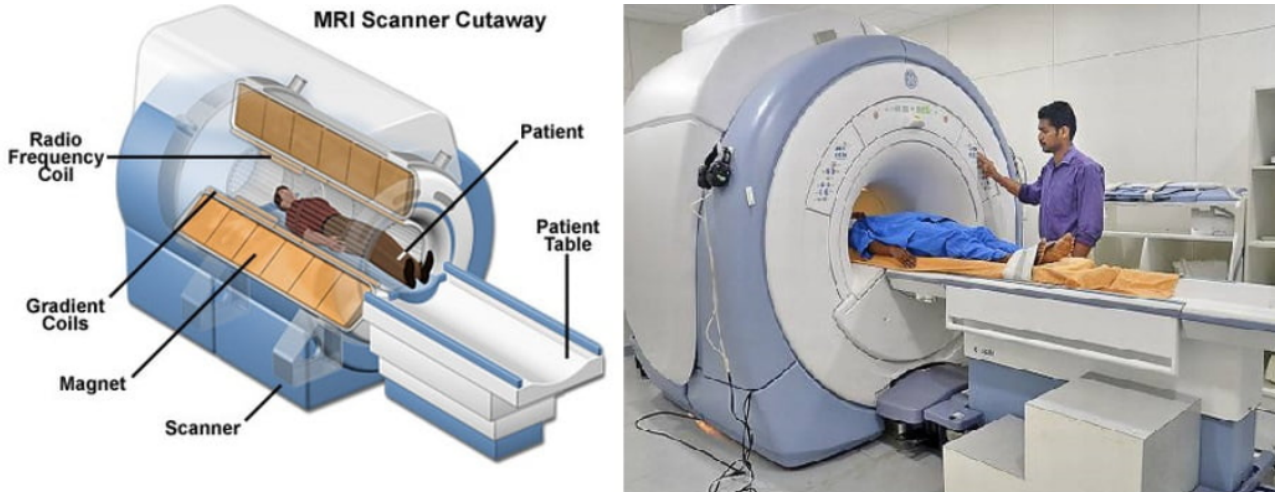
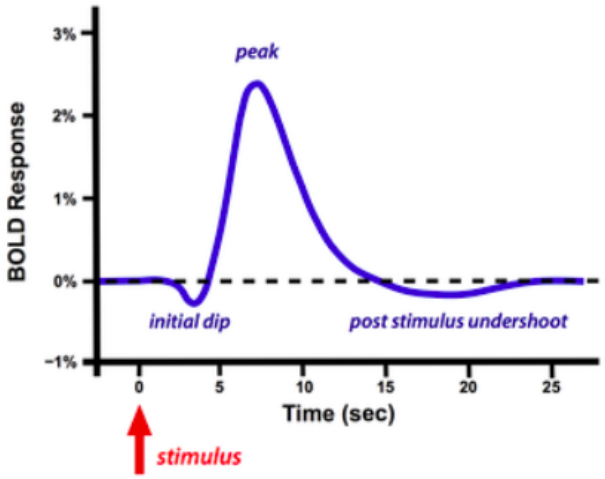


Figure 1.3. A view of an MRI scanner. (https://engineeringinsider.org/mri-machine-uses-working)

When neurons in a certain brain region are activated, metabolic activity in these neuronal tissues will change due to the oxygen needed by these cells for producing energy. In the fMRI technique, the time-varying changes of the blood-oxygen-level-dependent (BOLD) signal at the different parts of the brain is measured. When the neuronal tissue is activated, in the first phase the BOLD signal level reduces, and the concentration of oxygenated blood increases in these activated regions. After around 5 seconds, the BOLD signal level reaches its maximum level. When there is no more activation in the tissue the tissues become deoxygenated, and after about 15 seconds the BOLD signal level reaches its initial state [6, 7]. (Figure 1.4)



BOLD Hemodynamic Response Function (HRF)
following a single brief stimulus

Figure 1.4. The shape of a BOLD Response from a brief peripheral stimulus is known as the Hemodynamic Response Function (HRF). After the stimulus onset, there is an initial dip, followed by a positive dominant peak and a post-stimulus undershoot. After a couple of seconds, the signal goes back to its usual level (demonstrated with the dashed line).
(https://mriquestions.com/does-boldbrain-activity.html)

Oxygenated and deoxygenated hemoglobin carry different magnetic properties and Magnetic Resonance Imaging (MRI) scanners can distinguish these magnetic properties by aligning the atomic nuclei to a magnetic field and can measure the emitted electromagnetic signal. Therefore, MRI scanners give us the possibility to benefit from this cerebral hemodynamic activity to produce images of the brain using these differing contrasts through non-invasive approaches.

Generally, in order to track cognitive processes, participants are asked to perform specific tasks inside the scanner that are precisely designed for this purpose. The data is measured in the voxels with a specific dimension determining the fMRI image's spatial resolution. The value of each voxel is the average of the BOLD signals in that brain area. To capture a three-dimensional image across the entire brain activity, a certain time is required, which is called repetition time (TR) and depends on the type of scanner. Brain parcels refer to groups of voxels that are segmented to represent some anatomical/functional similarities. In some parcellation strategies, such as Multiresolution Intrinsic Segmentation Template (MIST), parcels are not indicating spatially contiguous locations. Therefore, instead of parcel or parcellation, the terms 'region of interest' (ROI) or simply 'region' (for spatially contiguous parcel) might be used. The time series of each parcel can be estimated generally by averaging the time series of all the voxels within a parcel.

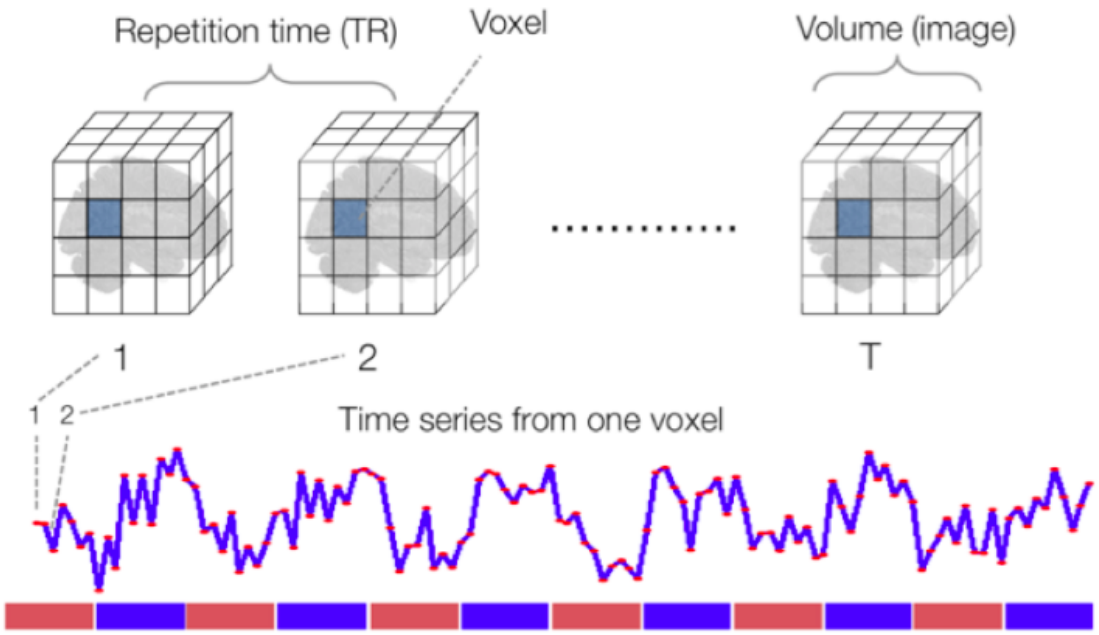


Figure 1.5. BOLD data time series, fMRI dataset consists of several time series of 3-D images (volumes, often covering the whole brain), measured at multiple time steps (TRs). (https://leanpub.com/principlesoffmri/read)

1.3.2. Overview of fMRI data Preprocessing

Recorded fMRI data from the scanner requires a sequence of appropriate preprocessing operations to assure the quality of recorded data and make it usable for later analysis.

Some of these steps help to detect and remove spurious artifacts in the data. These artifacts may be caused either by the MRI scanner machine or the participants. During this operation, certain statistical assumptions are applied in the spatial locations of brain regions. Some of these processes can be applied using existing fMRIprep pipelines, while some preprocessing operations need to be chosen based on the desired statistical analysis. All the preprocessing steps are not essential for all the experiments, and these series of operations need to be selected attentively to guarantee the best use of data and reliable results. Generally, after ensuring that the data are not corrupted by major artifacts (Quality control) a series of preprocessing steps will be applied to the fMRI dataset. The most important ones are slice-timing (timing issues) correction, motion (head movement)

correction, spatial normalization (transforming image onto a standard anatomical reference space), and spatial smoothing (noise correction).

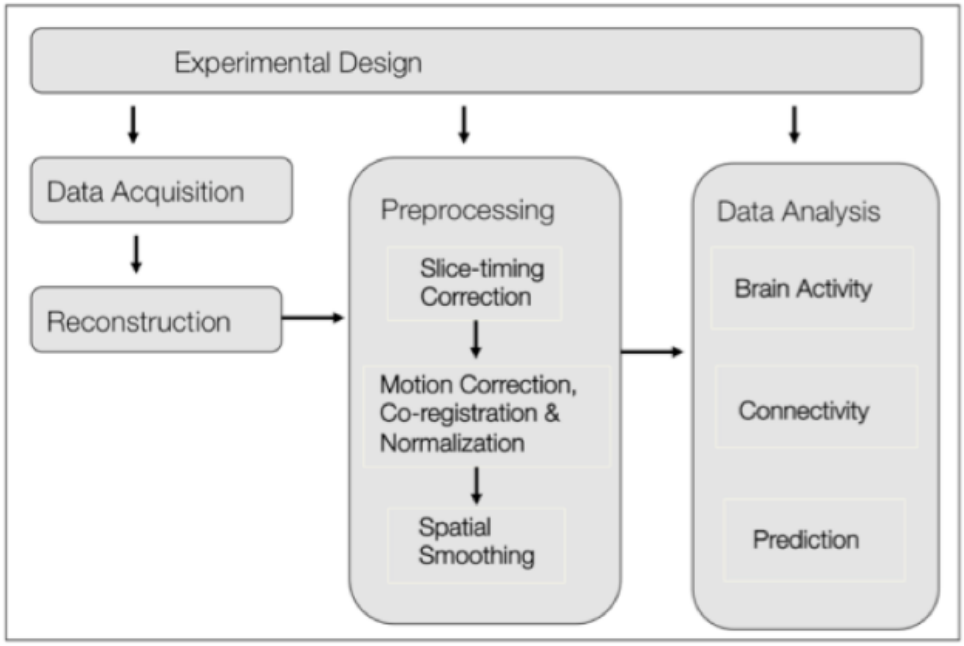


Figure 1.6. A diagram of the fMRI preprocessing pipeline. https://leanpub.com/principlesoffmri/

1.3.2.1 Slice-time Correction

Slice-time correction refers to the correction of differences in the acquisition time of different voxels in the image, which can become problematic for the analysis of fMRI data. By applying this correction, we ensure that all the voxels contained in a particular brain volume were acquired at exactly the same time. One of the most popular approaches for slice-timing correction is selecting a slice as a reference and interpolating the rest of the slices to match the timing of that reference slice. This data correction is not required when the data acquisition can be done rapidly by acquiring multiple slices simultaneously. (Poldrack, Mumford, and Nichols 2011; “Read Principles of fMRI” n.d.)

1.3.2.2 Motion Correction

Participants' head movement can have drastic effects on fMRI. The assumption is that time course points from a particular voxel contain signals from only its actual location,

while head motions during and between the scan session can cause the location mismatch of subsequent images in the time series (bulk motion). The motion correction processing ensures the consistency of the values represented by the brain's voxels. A standard motion correction technique can correct these artefactual effects by realigning the scans across time and applying a rigid body transformation (rotation and translation parameters) to each volume. Usually, the first volume is considered as a reference in the time course. (Poldrack, Mumford, and Nichols 2011)

1.3.2.3 Co-registration

A co-registration step deals with the anatomical alignment of the scans. In some cases, the recorded fMRI BOLD image volumes might not be well aligned with the structural features of the brains. The image co-registration process addresses this issue and fits scans into another space using the recorded anatomical images since they have a clear distinction of boundaries. For this purpose, the functional volumes are fitted to the anatomical image, and map the fMRI signal onto the surfaces generated in the anatomical images using a regression. (“Read Principles of fMRI” n.d.)

1.3.2.4 Spatial Normalization

Spatial Normalization refers to the alignment of data from different individuals into a common spatial framework. Since the size and shape and sizes of the brain are different for individuals a common space of fixed dimensions can be very useful in analyses. It helps combine data for a group analysis as they map to the same anatomical or functional structures. In this application the data of each subject is warped into a standard template space, e.g Montreal Neurological Institute (MNI) template.

1.3.2.5 Spatial Smoothing

The spatial smoothing process includes blurring the scans intentionally to reduce data noise. This process involves applying a filter to the fMRI images for removing high-frequency information. The filter should not be larger than the activation signals that we want to investigate. The main reasons behind this process are: helping to increase the signal-to-noise ratio for signals with larger spatial scales and also overcoming the increased noises when we are dealing with small voxels. Besides, it can help reduce the mismatches across individuals when data are combined in inter-individual studies. Moreover, it should be considered that for some analyzing approaches a specific degree of

spatial smoothness is required, such as convolution of the three-dimensional image with a three-dimensional Gaussian filter (or kernel). (Poldrack, Mumford, and Nichols 2011)

1.4. The Courtois NeuroMod hcptrt dataset

In this study, we used the HCP test-retest (hcptrt) dataset acquired from the CNeuroMod databank ("Datasets — Courtois NeuroMod 2020-Beta Documentation" n.d.). The main objective of the CNeuroMod project is to create a rich individual neuroimaging data resource that can accelerate the study of brain activity using integrative AI models and help investigate the data scaling properties of such models. CNeuroMod project accumulates hundreds of hours of neuroimaging scans per subject that include structural and functional MRI scans with a wide variety of cognitive domains and several biophysiological measures simultaneously. CNeuroMod project is already one of the largest available individual fMRI datasets which cover both naturalistic and controlled stimuli. It contains numerous naturalistic tasks from different cognitive domains, such as movie watching, video game playing, and functional tasks designed for the Human Connectome Project. The CNeuroMod project goal is to present a unique neuroimaging resource in terms of the amount of data at the level of an individual also the variety of cognitive domains.

Hcptrt, which is one of the CNeuroMod datasets, is a dataset of block-design fMRI tasks and includes over 8h of BOLD fMRI data per participant. The principal purpose of hcptrt is to offer a benchmark for individual-level brain decoding using naturalistic and easy-to-decode tasks. The tasks were designed for the HCP consortium, and have been acquired on N=1200 subjects, with 1h of BOLD fMRI data per participant. The hcptrt dataset is thus an individual version of one of the largest and most popular benchmarks for brain decoding currently available.

1.4.1. Task design/paradigm

The fMRI paradigm was based on the Human Connectome Project (HCP)'s task battery, which assesses seven cognitive domains and 23 different task conditions. Detailed descriptions of the HCP tasks and hcptrt protocol can be found in Barch et al., 2013 and CNeuroMod documentation ("Datasets — Courtois NeuroMod 2020-Beta Documentation" n.d.), respectively. Seven different tasks were included in the hcptrt dataset: emotion processing, gambling, language processing, motor, relational processing, social cognition, and working memory. In a previous study (Y. Zhang et al. 2021), our group excluded the

gambling task due to its short event design (1.5s for button press, 1s for feedback, and 1s for ITI). In the current study, we tested our benchmark decoding models with and without the gambling task conditions, and also opted to exclude the gambling task because of its low precision and recall scores compared to the remaining six tasks (listed in Table 2.1). Participants were given detailed instructions and the possibility to practice each task using an example session before the first session. All participants completed between 13 and 18 runs of each task over 10 to 13 separate fMRI sessions. While some participants completed more than 14 runs for certain tasks, we included only 14 runs per task to match the number of repetitions across tasks and participants, which in total included around 7h 51min of fMRI data per subject (except sub-03, for which only 13 runs of the working memory task were available at that time). Each run was dedicated to a single task (block design). Each session either included two runs for each of the seven tasks, or one run per task plus one 15 min resting-state run. Task order was counterbalanced across sessions.

Emotion processing task

During the emotion processing task (92 volumes or 2m 17s per run), participants were shown a triad of images of either faces or shapes and were asked to select from the two images at the bottom which matched the target image (face or shape) located at the top of the screen (adapted from Smith et al. 2007). Face images either had an angry or a fearful expression. Faces and shapes were presented in three separate blocks each (blocks of faces or shapes) followed by one fixation block at the end of each run.

Language task

The language task includes two conditions; story and mathematics (159 volumes or 3m57s per run). In the story condition, participants listened to a brief auditory story (~20s), followed by a two-alternative forced-choice question about the story. In the mathematics condition, participants listened to a series of arithmetic questions (additions and subtractions), followed by two choices of answers to the operations. For the math task, the level of difficulty increased with every correct answer to maintaining a similar level of difficulty across participants. Each task repetition included two runs, each with four stories and four mathematical blocks, interleaved.

Motor task

During the motor task (144 volumes, 3m34s per run), participants were presented with a

visual cue instructing them to either tap their left or right fingers, squeeze their right or left toes, or move their tongue (adapted from Buckner et al. 2011; Yeo et al. 2011). Each of the task's two runs included 13 blocks: two tongue movements, four hand movements (two right and two left), four foot movements (two right and two left), and three additional 15s blocks of fixation.

Relational task

The relational task included two conditions: matching and relational processing (119 volumes or 2m57s per run). During the relational processing condition, participants answered whether or not a pair of objects shown at the top of the screen matched along the same dimension (shape or texture) as a pair of objects shown at the bottom of the screen. In the control matching condition, participants answered whether or not an object shown at the bottom of the screen matched either one of two objects shown at the top of the screen along a dimension (shape or texture) specified by a cue word in the middle of the screen (adapted from Smith et al. 2007). The task included two runs, each with three relational blocks, three matching blocks, and three fixation blocks.

Social cognition task

In the social cognition task (139 volumes or 3m27s per run), participants were presented with short (20s) video clips of objects (squares, circles, triangles) that either interacted in some way ("mental interaction condition", where shapes appeared to take each other's thoughts and feelings into consideration) or moved randomly on the screen ("random condition"). Following each clip, participants needed to identify the condition it belonged to "mental interaction", "no interaction" or "not sure". Each of the task's two runs included five videos of each condition and five 15s fixation blocks.

Working memory (WM) task

The working memory task (202 volumes or 5m1s per run) consisted of a category recognition (0-back) sub-task and a 2-back working memory sub-task. Each sub-task included separate blocks of places, tools, faces, and body parts stimuli. During the 2-back task, participants indicated whether they saw the same image 2 images back. During the 0-back task, participants indicated whether they saw an image that matched a target shown at the beginning of the trial. Each of the task's two runs contained 10 trials from each of the eight conditions (stimulus type * subtask), as well as four 15s fixation blocks.

For each HCP task except WM, task conditions (e.g., the “mental” and “random” conditions from the social cognition task) were used as task labels to identify cognitive states for brain decoding. To be consistent with our previous work, combinations of stimulus type (faces, tools, etc) and trial type (0-back and 2-back) were considered as eight different task labels for the WM task, for a total of 21 task labels across the hcptrt dataset (Table 2.2, “Labels” column).

1.5. Decoding methods

We evaluated nine different intra-subject brain decoding models using CNeuroMod’s hcptrt, a dense individual fMRI dataset for a challenging benchmark on single volume classification across 21 task conditions from six different cognitive domains using less than 8 hours of single-subject data. Models included eight conventional machine learning algorithms: Support Vector Machine (SVM), multi-layer perceptron (MLP), Logistic Regression, K-Nearest Neighbors (KNN), Ridge Regression, Bagging Ensemble, Random Forest Classifier (RFC), and Gaussian Naive Bayes models, all implemented using Scikit-learn (Pedregosa, Varoquaux, and Gramfort, n.d.). We also evaluated a state-of-the-art GCN whose implementation was adapted from our previous work (Y. Zhang et al. 2021). For each subject, the dataset was split into 80-20% for train and test sets respectively and the results are reported from the test set. Since our data structure consisted of several imbalanced functional tasks from different runs (volumes per task, Table 2), we used a stratified 10-fold cross-validation approach with three repetitions (RepeatedStratifiedKFold in Scikit-learn) for our GridSearchCV function, to ensure that each fold had an equal proportion of observations with a given class label. We ran three repetitions for all experiments. The learning algorithms and hyperparameters for the Scikit-learn models were tuned using a grid search approach performed on the train set, as detailed below for each model.

1.5.1. Support Vector Machine

In the SVM algorithm for classification, data items are plotted as a point in N-dimensional space where N depends on the number of features that distinctly classify the data points. The objective is to find a hyperplane with the maximum margin. Data points falling on either side of the hyperplane can be attributed to different classes.

Hyper-parameters (gamma, C, and kernel type) were tuned with a grid search optimization. The most accurate results were achieved with C=10, gamma=0.001, and the ‘RBF’ kernels.

1.5.2. Logistic Regression

Logistic regression is a fast and relatively uncomplicated linear classifier that uses the ‘sigmoid function’ (‘logistic function’) to return a probability value, (Eq. 1).

$$f(x) = \frac{1}{1+e^{-(x)}} \quad \text{(Eq. 1)}$$

It analyzes a set of data points with one or more independent input variables and finds the best-fitting model to describe the data points.

Grid search optimization was conducted and the best prediction results were obtained with Liblinear as the solver algorithm, L1 penalty, C=0.1, 20 as the maximum iterations, and all classes weighted as one (class weight = ‘None’).

1.5.3. Bagging Ensemble

We implemented a Bagged Decision Trees classifier with grid search. A Bagging classifier is an ensemble machine learning algorithm that fits base models on random subsets of the original dataset and then aggregates all the individual predictions to form a final prediction. It is mainly used with decision trees since they're a reliable way of achieving regularization. In ensemble methods, the main idea is that correctly combining several weak classifiers can lead to a more accurate and strong learner strategy. Therefore in ensemble methods, rather than having a single model for the best solution, multiple models solve a common problem independently, and their predictions are combined following some kind of deterministic averaging process that the most common is taking the majority vote. We used the Scikit-learn implementation of Bagging ensembles for machine learning.

We used grid-search to optimize a Bagged Decision Trees classifier implemented with Scikit-learn. This algorithm aggregates predictions from weak base models with a deterministic averaging process. The most accurate results were achieved with n_estimators=1000, max_features=100, bootstrap=False, and warm_start=True.

1.5.4. Gaussian Naive Bayes

Another supervised machine learning algorithm, which was tried in this study, is Gaussian Naive Bayes. Naive Bayes is a group of probabilistic supervised machine learning classification algorithms based on applying Bayes' theorem with the 'naive' assumption of conditional independence between the features and their equal contribution to the outcome. The Naive Bayes model is easy to build and particularly useful for very large data sets, but its biggest disadvantage is the requirement for predictors to be independent. There are different extensions of the Naive Bayes Classifier, including Multinomial Naive Bayes, Bernoulli Naive Bayes, and the Gaussian Naive Bayes. Gaussian Naive Bayes is a simple approach that follows Gaussian normal distribution and supports continuous data that only needs to estimate the mean and standard deviation from the training data.

To tune the hyper-parameters of Gaussian Naive Bayes, we implemented a grid search across different values for the variance smoothing parameter, including 15 values sampled between 1 and 10^{-14}. The most accurate results were achieved with 10^{-4}.

1.5.5. K-Nearest Neighbors

In the K-Nearest Neighbors classification, a point is assigned to the data class that has the majority vote (e.g. the label that has the most representation for a given data point, know also as ‘plurality voting’) within each point's nearest neighbors. The algorithm stores all available cases and then classifies new cases considering a similarity measure.

We tuned the following hyper-parameters: the number of neighbors, the query resolution (i.e., leaf size), the weight function used in the predictions, the distance metric, and the algorithm for computing the nearest neighbors. A grid search optimization was performed and the best results were achieved with four neighbors, a leaf size of one, the Minkowski distance metric (power=1), the “distance” weight function (inverse of distance for the weight points), and the ball_tree algorithm.

1.5.6. Random Forest

Another supervised learning approach that we tried is the Random Forest classifier. This method contains several individual decision trees, each of which outputs a prediction, but operates as an ensemble at training time. Decision trees split data into small sets of

data based on the data features and continue this splitting until ending up with a group of data with a single label. The final output prediction is the class selected by most trees. Typically, the greater number of trees in the forest leads to higher accuracy and prevents overfitting.

The model hyperparameters were optimized with grid search, and the most accurate results were achieved with 200 estimators, maximum features at 21, a maximum tree depth of 20, a minimum number of samples of 2, a minimum number of leaf samples of 1, and without bootstrapping.

1.5.7. Ridge Regression

Ridge Regression, also known as the L2 Regularization model, is an extension of linear regression. In linear regression, the model treats all the features equally by assuming a linear relationship between the input and the target variable. While in ridge regression to increase model stability and avoid overfitting problems, the model tries to reduce the effect of unimportant input variables for the task prediction. For this purpose, the model penalizes the loss function of the Linear Regression based on the sum of the squared coefficient values (beta), known as the L2 penalty. This means that the model brings the coefficients of variables with low relationships closer to zero. For this study, we ran this model using the Scikit-learn package and its Ridge class.

A grid search optimization was applied and the most accurate results were achieved with alpha=0.1, normalize=True, and the lsqr solver.

1.5.8. Multi-layer Perceptron

We trained a Multilayer Perceptron (MLP) classifier with two dense layers (‘ReLU’ activation function) and one output layer. MLPs are one of the most basic architectures of artificial neural networks. Perceptron is a neural network that performs a simple binary classification that can be considered as ‘true’ or ‘false’. MLPs model consists of input and output layers as well as hidden layers (in most cases) that transform the input into usable data for the output layer. Like other machine learning models for supervised learning, an MLP initially undergo a training phase. During this supervised phase, the network is taught what to look for and what is the desired output. The difference between the results is adjusted using backpropagation (going from model output to input) to achieve the lowest possible difference between the actual and desired outcome. In this study, we implemented a basic MLP architecture by creating a sequential model and adding Dense layers with the ‘ReLU’ activation function. The layers consisted of the input layer, one output layer, and just one hidden layer, using Keras libraries.

Hyper-parameters were tuned with a grid search optimization. The model output size is the number of decoding classes (21 classes), and the input size is the parcellation resolution for the input layer. The best results were achieved with a single hidden layer, and (resolution/2) as the size of the hidden layer. ReLU was selected as the hidden layer activation function, adam as the optimization algorithm and alpha = 0.05 for L2 regularization strength, with a constant learning rate.

1.5.9. Graph Convolutional Network

We adapted the Graph Convolutional Network (GCN) model from Zhang et al. (2021) to decode high-order cognitive functions from neural activity distributed across brain networks. One of the main components of the GCN approach is the brain graph, which provides a network representation of brain organization. The brain graph, implemented as a connectivity matrix, associates graph vertices to brain regions, while graph edges represent anatomical or functional brain connections (Figure 2). The current GCN relied on a fixed human connectome as a static graph, which we generated by extracting time series from a full brain atlas for each subject. More specifically, we generated the brain graph from the weighted average of a complete run of the hcptrt using the ConnectivityMeasure class of the Nilearn package. Our GCN model consisted of four ChebNet graph convolutional layers with 64 graph filters per layer, followed by a layer of global average pooling and an MLP with two fully connected layers before classification (21 classes; Figure 2). We used backpropagation to train the model in PyTorch. The network was trained for 10 epochs with batch sizes of 10. After each round of training, 10% of the data were used as a validation set to calculate the average accuracy and loss. After completing all epochs, we used the unseen test data (20%) to evaluate the performance of the final model.

This GCN implementation took a 1.49s (single volume) time window of fMRI signal as input and mapped this high-dimensional input onto a k-nearest-neighbor (k-NN) graph, which was generated by connecting each brain region to its k neighbors with the strongest connectivity. To define the brain regions, we tested several brain graph architectures based on three different predefined atlases of various dimensionalities, including MIST (122, 444), DiFuMo (128, 256, 512, 1024), and Schaefer (100, 400, 1000). Since the Schafer-1000 achieved the best results, we performed our experiment using this parcellation atlas.

For optimizing our GCN model we manually evaluated several hyper-parameters. The following parameters and values were chosen:

- constraining the graph from clusters with 16 neighboring nodes with the strongest connectivity,
- 5e-4 for weight regularization,
- 1e-4 learning rate with lr_scheduler.ReduceLROnPlateau,
- Four ChebNet graph convolutional layers with 64 ChebNet graph filters of size 3 at each layer,
- A global average pooling layer after the four convolutional layers,
- MLP model with two hidden layers, resolution of chosen parcellation method as input, 21 task conditions as output, zero dropouts, and bias=True.

1.5.10. Baseline

To have a baseline to evaluate model performances, we estimated the chance level using the ‘DummyClassifier’ class from Scikit-learn. Since this model makes predictions while ignoring the input feature values, we used the results of this simple baseline to benchmark a chance level with the results of our other classifiers.

Chapter 2

Article

Brain decoding of the Human Connectome Project Tasks in a Dense Individual fMRI Dataset

Available as a preprint: (Rastegarnia et al., n.d.)

Submission for publication pending final revisions.

Abstract

Brain decoding studies aim to identify patterns of brain activity that reflect the participant's cognitive state. Substantial inter-individual variations in functional brain organization represent a challenge to accurate brain decoding. In this paper, we introduce a dense individual functional magnetic resonance imaging (fMRI) dataset for which a task battery popular for brain decoding was executed repeatedly: six participants completed the entire Human Connectome Project (HCP) task battery >13 times over ten separate fMRI sessions. We used these data to assess whether accurate brain decoding models can be successfully trained entirely at the individual level.

We compared several decoding methods, from classic Support Vector Machines (SVM) to complex Graph Convolutional Neural Networks (GCN). All subject-specific decoders were trained to classify single fMRI volumes into 21 experimental conditions simultaneously, using around seven hours of fMRI data per participant.

The best prediction accuracy results were achieved with our support vector machine model with a test accuracy ranging from 64-79%, with a chance level of around 7%. Both multilevel perceptrons and graph convolutional networks also performed very well (63-78% and 63-77%, respectively). Feature importance maps derived from the best model

revealed informative features in regions relevant to particular cognitive domains, notably in the motor cortex. Inter-subject classification achieved substantially lower accuracy than subject-specific models, indicating that individual brain decoders learned individual-specific features.

Our work demonstrates that deep neuroimaging datasets can be used successfully to train accurate brain decoding models. We expect such datasets to provide useful accuracy targets for techniques aimed to train group-level models generalizing over multiple subjects.

Keywords

Artificial Intelligence, Machine Learning, fMRI, Brain Decoding, Cognitive States

2.1. Introduction

Brain decoding aims to infer cognitive states from snapshots of brain activity captured with techniques like functional magnetic resonance imaging (Figure 1). Most studies aim to generalize brain decoding to data from new, previously unseen subjects, but this approach is challenging due to large inter-subject variability in brain structure and function (Gordon et al. 2017; Gratton et al. 2018). Therefore, a domain-general decoder that can classify several different cognitive tasks simultaneously must usually be trained on very large-scale collections of brain images to achieve good prediction accuracy (Y. Zhang et al. 2021). Alternatively, it might be possible to assess the feasibility of brain decoding at the level of a single subject, if sufficient data are acquired. The main objective of the current study was to benchmark a variety of brain decoding models trained at the individual level using one of the largest collections of multi-task fMRI data available to date and a popular set of tasks designed by the Human Connectome Project (HCP).

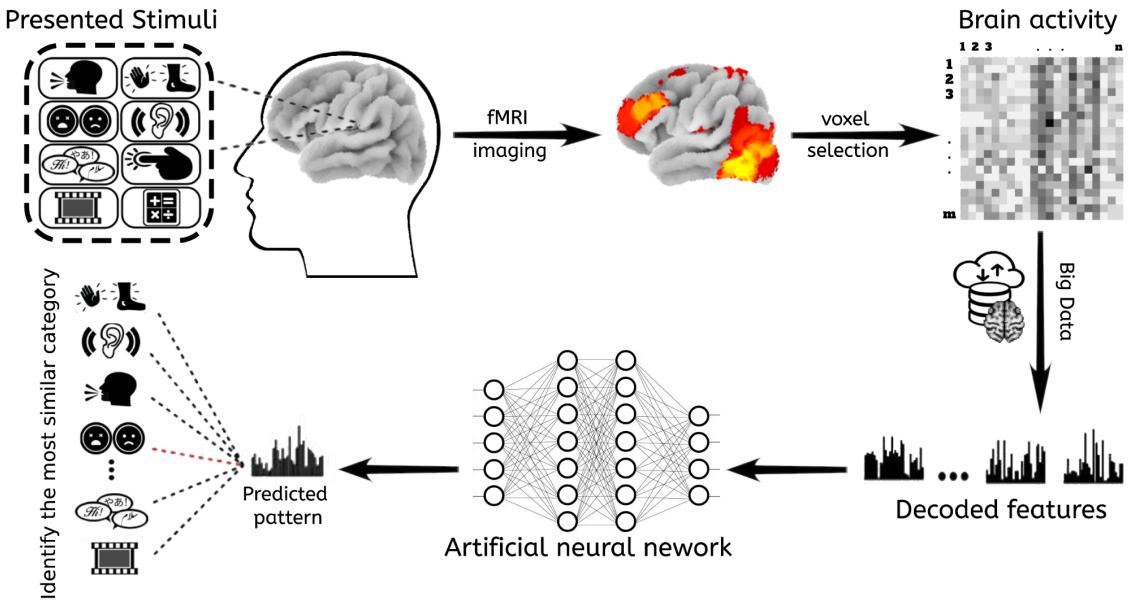


Figure 2.1. Demonstration of the brain decoding procedure. Voxel activity is extracted from fMRI data files and passed to the decoding model, which attempts to predict the presented stimulus or cognitive state by inferring meaningful features from the brain response.

Brain decoding research has progressed significantly since its introduction over 20 years ago (J. V. Haxby et al. 2001). To date, multiple computational methods have been proposed to decode mental states from brain imaging data, with dominant approaches including multi-voxel pattern classification (MVPA) (James V. Haxby, Connolly, and Guntupalli 2014; Poldrack 2011), classic statistical analyses based on linear regression, and the general linear model (Nishimoto et al. 2011), as well as advanced techniques that rely on deep artificial neural networks (Horikawa and Kamitani 2017; Y. Zhang et al. 2021). New brain decoding applications include the successful reconstruction of movies from brain activation patterns (Han et al. 2019), extracting semantic context from words (Grand et al. 2018), and decoding fluid mental processes such as intentions (Haynes et al. 2007; Gilbert and Fung 2018), mental imagery (Horikawa and Kamitani 2017; Bergmann, Morgan, and Muckli 2019), and neural reactivity during sleep (Sterpenich et al. 2021; Rué-Queralt et al. 2021). These methods either require a large-scale collection of brain imaging data (Y. Zhang et al. 2021) or they can gather massive input data through meta-analytic evidence (Bartley et al. 2018). For instance, the Graph Convolutional Network model (GCN) proposed by Zhang et al. (2021) achieved high brain decoding prediction accuracy on 21 experimental conditions from the HCP dataset, requiring over 200 hours of data to perform this type of group-level decoding.

The need to train decoders on such large datasets likely originates from substantial inter-subject variability in both brain structure and function, with important differences in brain activity patterns observed when participants perform the same cognitive tasks (Gordon et al. 2017; Gratton et al. 2018). Most previous brain decoding studies have analyzed data averaged across groups of subjects, which causes a loss of individual brain features by restricting brain data to a common functional connectivity space. These limitations make between-subject decoding more challenging than decoding performed at the individual level.

One solution is to use functional alignment techniques like hyper-alignment, which matches subjects' neural signals according to their functional similarities, to avoid naive averaging across participants (James V. Haxby et al. 2011). While these methods have proved promising to improve decoding across individuals (Bazeille et al. 2021), the best method to identify functional similarity remains an open research question. An interesting alternative is to train decoders directly on individual participants' brain activity patterns, bypassing the need to align features across subjects. Importantly, training within-subject critically depends on the availability of sufficient data from individual participants who can be scanned extensively and reliably over long time periods.

Deep phenotyping datasets (e.g., (Naselaris, Allen, and Kay 2021)), for which a limited number of individuals get scanned intensively with a wide range of stimuli, make it possible to train classifiers on individuals. One prominent set of deep phenotyping data is the midnight scan club (MSC) dataset (Gordon et al. 2017), for which data were collected

from ten participants over 12 sessions that included resting-state, a motor task, mixed blocks of event-related perceptual and language tasks, and a memory task. In a recent study, (Porter et al. 2022) trained classifiers to decode functional states with the MSC dataset, and showed that within-subject decoding performance was significantly better than between-subject decoding. However, the range of stimuli and cognitive states decoded in this study was limited compared to other domain-general decoders (e.g., (Y. Zhang et al. 2021)). Another deep collection of individual-specific data is the natural scenes dataset (NSD) (Allen et al. 2022), for which eight participants were scanned extensively while performing an image recognition task on tens of thousands of annotated images of natural scenes. However, even though NSD includes tens of hours worth of human brain activity per subject and many image categories, it investigated only one cognitive domain. The Doctor Who dataset is another deep phenotyping dataset for which a single individual watched 30 episodes (approx. 23 h) of the Doctor Who TV series over a six months period (Seeliger et al. 2019). Convolutional neural networks have been successfully trained to predict neural responses to visual and auditory content from naturalistic stimuli with this dataset (Seeliger et al. 2021). However, easily classifiable cognitive states are difficult to pinpoint in movie stimuli. Also, studies conducted on a single subject do not allow for assessing the model's generalizability across subjects.

With the current project, we aimed to train individual domain-general brain decoding models that could decode many different cognitive tasks to determine whether high decoding accuracy can be achieved with fewer hours of fMRI training data acquired within rather than across individual subjects. We trained several classifiers on the HCP test-retest (hcptrt) dataset acquired by the Courtois project on Neural Modelling (“Datasets — Courtois NeuroMod 2020-Beta Documentation” n.d.), for which six participants underwent extensive scans while performing a series of tasks from different cognitive domains adapted from the Human Connectome Project (HCP) (Van Essen et al. 2013). To identify the best-performing decoder for this task, we tested various methods popular in the group-level decoding literature, from simple models like Support Vector Machine (SVM) (Cortes and Vapnik 1995) and multilayer perceptron (MLP) (Bishop and Bishop 1995) to sophisticated deep learning models such as GCN (Y. Zhang et al. 2021). SVM (Hanson and Halchenko 2008; Bi et al. 2018; Eslami and Saeed 2019; C. Zhang et al. 2019) and MLP (Eslami and Saeed 2019; Hacker et al. 2013) decoders are established techniques in supervised classification that are well-adapted for high-dimensional datasets. As for GCNs, our team has used them to achieve state-of-the-art between-subject decoding accuracy in previous experiments (Y. Zhang et al. 2021; Y. Zhang, Farrugia, and Bellec 2022). Of note, all models were trained and tested to decode single fMRI volumes (rather than blocks of volumes or single trials modeled with a canonical HRF function). In total, we benchmarked nine different classifiers with the hcptrt dataset.

We also assessed how well individual brain decoders generalize across subjects, and whether they can uncover general markers of cognitive states or are mainly sensitive to idiosyncratic features. To do so, we performed cross-subject analyses that tested individual models on data acquired from other subjects. In addition, we generated permutation importance maps from the trained SVM model using the Radial Basis Function (RBF) kernel to assess the homogeneity of decoding features learned across subjects and to evaluate whether decoding was inferred from cognitively relevant brain areas.

2.2. Materials and Methods

2.2.1. fMRI data

2.2.1.1. The Courtois NeuroMod hcptrt dataset

In this study, we used the HCP test-retest (hcptrt) dataset acquired from the Courtois NeuroMod databank ("Datasets — Courtois NeuroMod 2020-Beta Documentation" n.d.). *Hcptrt* is a dataset of block-design fMRI tasks that include over 8h of BOLD fMRI data per participant. The principal purpose of hcptrt is to offer a benchmark for individual-level brain decoding using naturalistic and easy-to-decode tasks.

2.2.1.2. Participants

Six healthy participants, all right-handed, were recruited for this project with no record of neurological disorders (aged 31 to 47 at acquisition), including three females (sub-03, sub-04, and sub-06) and three males (sub-01, sub-02, and sub-05). All had normal hearing and vision for their age. Three participants reported being native francophone speakers (sub-01, sub-02, and sub-04), one native anglophone (sub-06), and two bilingual native speakers (sub-03 and sub-05).

2.2.1.3. Magnetic resonance imaging

Magnetic resonance imaging (MRI) data were acquired with a 3T Siemens Prisma Fit scanner using a 64-channel head/neck coil. Functional MRI data were acquired using an accelerated simultaneous multi-slice, gradient echo-planar imaging sequence (Setsompop et al. 2012; Xu et al. 2013) developed at the Center for Magnetic Resonance Research

(CMRR) at the University of Minnesota, as part of the Human Connectome Project (Glasser et al. 2016). The sequence is available on the Siemens PRISMA scanner at UNF through a concept-to-production (C2P) agreement, and was used with the following parameters: slice acceleration factor = 4, TR = 1.49s, TE = 37 ms, flip angle = 52 degrees, 2mm isotropic spatial resolution, 60 slices, 96x96 acquisition matrix. Participant responses were collected using a 4-button MRI-compatible response pad made by Current Designs. Structural data were acquired using a T1-weighted MPRAGE 3D sagittal sequence (duration 6:38 min, TR = 2.4 s, TE = 2.2 ms, flip angle = 8 deg, voxel size = 0.8 mm isotropic, R=2 acceleration). For more information on sequences and data acquisition (including fMRI setup), see the CNeuroMod documentation ("Datasets — Courtois NeuroMod 2020-Beta Documentation" n.d.).

2.2.1.4. Preprocessing

All fMRI data were preprocessed using the fMRIprep pipeline version 20.2.3 ("long-term support") (Esteban et al. 2019). We used a volume-based spatial normalization to standard space (MNI152NLin2009cAsym). The anatomical mask derived from the data processing phase was used to identify and select the brain voxels. 2D data matrices (TR x ROI) were generated from 4-dimensional fMRI volumes using the NiftiLabelsMasker tool from Nilearn. The BOLD time series were smoothed spatially with a 5mm Gaussian kernel. A denoising strategy that regressed out basic motion parameters and mean global, white matter, and cerebrospinal fluid signals ('simple" denoising implemented with load_confounds_strategy in Nilearn) was used to remove confounds without compromising the temporal degrees of freedom.

2.2.1.5. Task design/paradigm

The fMRI paradigm was based on the Human Connectome Project (HCP)'s task battery, which assesses seven cognitive domains and 23 different task conditions. Detailed descriptions of the HCP tasks and hcptrt protocol can be found in Barch et al., 2013 and CNeuroMod documentation ("Datasets — Courtois NeuroMod 2020-Beta Documentation" n.d.), respectively. Seven different tasks were included in the hcptrt dataset: emotion processing, gambling, language processing, motor, relational processing, social cognition, and working memory. In a previous study (Y. Zhang et al. 2021), our group excluded the gambling task due to its short event design (1.5s for button press, 1s for feedback, and 1s for ITI). In the current study, we tested our benchmark decoding models with and without the gambling task conditions, and also opted to exclude the gambling task because of its low precision and recall scores compared to the remaining six tasks (listed in Table 1).

Participants were given detailed instructions and the possibility to practice each task using an example session before the first session. All participants completed between 13 and 18 runs of each task over 10 to 13 separate fMRI sessions. While some participants completed more than 14 runs for certain tasks, we included only 14 runs per task to match the number of repetitions across tasks and participants, which in total included around 7h 51min of fMRI data per subject (except sub-03, for which only 13 runs of the working memory task were available at that time). Each run was dedicated to a single task (block design). Each session either included two runs for each of the seven tasks, or one run per task plus one 15 min resting-state run. Task order was counterbalanced across sessions.

Task	# Runs	# Conditions	# Volumes (per run)
Emotion	14	2	92
Language	14	2	159
Motor	14	5	144
Relational Processing	14	2	119
Social Cognition	14	2	139
Working Memory	*13-14	8	202
6 subjects (3F | 3M), Total 7h 51min per subject (* except 7h 46min for sub-03) * only 13 runs of the WM condition were available for sub-03.			

Table 2.1. Scanning task conditions and data size per run

For each HCP task except WM, task conditions (e.g., the “mental” and “random” conditions from the social cognition task) were used as task labels to identify cognitive states for brain decoding. To be consistent with our previous work, combinations of stimulus type (faces, tools, etc) and trial type (0-back and 2-back) were considered as eight different task labels for the WM task, for a total of 21 task labels across the hcptrt dataset (Table 2, “Labels” column).

Task	Conditions		Labels	Total volumes	* Final volumes	
Emotion	response face		fear	1288	662 - 672	7641 - 7801
	response shape		shape			
Language	presentation math, question math, response math		math	2226	1882 - 1925	
	presentation story, question story, response story		story			
Motor	response left foot		footL	2016 - 2032	700	
	response left hand		handL			
	response right foot		footR			
	response right hand		handR			
	response tongue		tongue			
Relational	Control		match	1666	925 - 979	
	Relational		relational			
Social	mental		mental	1946	1441 - 1477	
	random		random			
WM	Body	0-Back	body0b	2626 - 2828	1936 - 2088	
		2-Back	body2b			
	Face	0-Back	face0b			
		2-Back	face2b			
	Place	0-Back	place0b			
		2-Back	place2b			
	Tools	0-Back	tool0b			
		2-Back	tool2b			

Table 2.2. Brain volumes per task and experimental condition for the hcptrt dataset. Total volumes correspond to the number of volumes per functional task before data processing. Final volumes reflect the post-processing number of volumes after discarding the volumes of no interest (e.g. cue condition, resting states between task blocks, and the three volumes at the onset of each task block).

2.2.1.6. Brain parcellation

The BOLD time series were projected onto parcellation atlases. The average time series of each brain parcel was used as a feature in the classification analyses. We evaluated classification performance for different brain atlases at various scales, including Multiresolution Intrinsic Segmentation Template (MIST; 122 and 444 parcels) (Urchs et al. 2019), dictionaries of functional modes (DiFuMo; 128, 256, 512, and 1024 parcels) (Dadi et al. 2020) and Local-Global Parcellation of the Human Cerebral Cortex (Schaefer; 100, 400 and 1000 parcels) (Schaefer et al. 2018). MIST is a hierarchical parcellation of various cortical and subcortical regions. DiFuMo is a fine-resolution atlas of functional modes,

which includes different networks ranging from 64 to 1024 parcels. Schaefer is a gradient-weighted Markov Random Field (gwMRF) model based on local gradient and global similarity approaches.

2.2.1.7. Time series extraction

We excluded the first three volumes after the task onset of each trial (3X1.49s=4.47s), accounting for the delay of the hemodynamic response. All classifiers were trained for single-volume prediction, that is, to classify single TRs into one of 21 cognitive states (Table 1).

2.2.2. Decoding methods

We evaluated nine machine-learning methods for single-volume classification. Models included eight conventional machine learning algorithms: SVM, multi-layer perceptron (MLP), Logistic Regression, KNN, Ridge Regression, Bagging, Random Forest Classifier (RFC), and Gaussian Naive Bayes models, all implemented using Scikit-learn (Pedregosa, Varoquaux, and Gramfort, n.d.). We also evaluated a state-of-the-art GCN whose implementation was adapted from our previous work. For each subject, the dataset was split into 80-20% for train and test sets respectively and the results are reported from the test set. Since our data structure consisted of several imbalanced functional tasks from different runs (volumes per task, Table 2), we used a stratified 10-fold cross-validation approach with three repetitions (RepeatedStratifiedKFold in Scikit-learn) for our GridSearchCV function, to ensure that each fold had an equal proportion of observations with a given class label. We ran three repetitions for all experiments. The learning algorithms and hyperparameters for the Scikit-learn models were tuned using a grid search approach performed on the train set, as detailed below for each model.

2.2.2.1. Support Vector Machine

Hyper-parameters (gamma, C, and kernel type) were tuned with a grid search optimization across the following values:

- C in [0.1, 1, 10, 100, 1000],
- Gamma in [1, 0.1, 0.01, 0.001, 0.0001],
- Kernel type equal to RBF or linear kernel.

The most accurate results were achieved with C=10, gamma=0.001, and the ‘RBF’ kernels.

2.2.2.2. Logistic Regression

Grid search optimization was conducted across the following parameter values:

- Solver, the algorithm used for the optimization problem, in [‘newton-cg’, ‘lbfgs’, ‘liblinear’, ‘sag’, ‘saga’],
- Penalty in [‘l1’, ‘l2’, ‘elasticnet’, ‘none’],
- C in [100, 10, 1.0, 0.1, 0.01],
- Maximum number of iterations in [10, 20, 21, 50, 100, 1000],
- Class weights in ['balanced', None].

The best prediction results were obtained with Liblinear as the solver algorithm, L1 penalty, C=0.1, 20 as the maximum iterations, and all classes weighted as one (class weight = ‘None’).

2.2.2.3. Bagging Ensemble

We used grid-search to optimize a Bagged Decision Trees classifier implemented with Scikit-learn. This algorithm aggregates predictions from weak base models with a deterministic averaging process.

Our grid search was optimized over the following parameters and values:

- Number of base estimators in the ensemble in [10,100,200,800,1000,2000],
- Maximum features in [2,5,10,21,25,50,100,500,'sqrt'],
- Bootstrap in [True, False],
- Warm_start in [True, False]

The most accurate results were achieved with n_estimators=1000, max_features=100, bootstrap=False, and warm_start=True.

2.2.2.4. Gaussian Naive Bayes

To tune the hyper-parameters of Gaussian Naive Bayes, we implemented a grid search across different values for the variance smoothing parameter, including 15 values sampled between 1 and 10^{-14}. The most accurate results were achieved with 10^{-4}.

2.2.2.5. K-Nearest Neighbors

We tuned the following hyper-parameters: the number of neighbors, the query resolution (i.e., leaf size), the weight function used in the predictions, the distance metric, and the algorithm for computing the nearest neighbors. A grid search optimization was performed across the following parameter values:

- Number of neighbors (k) in [2, 4, 8, 10, 16],
- Leaf size in [1, 2, 10, 20, 30],
- Distance metric in ['minkowski', 'euclidean', 'manhattan'],
- p in [1, 2],
- Weight function in ['uniform', 'distance']
- Algorithm in ['ball_tree', 'kd_tree', 'brute'].

The best results were achieved with four neighbors, a leaf size of one, the Minkowski distance metric (power=1), the “distance” weight function (inverse of distance for weight points), and the ball_tree algorithm.

2.2.2.6. Random Forest

Our grid search for optimizing this model included the following hyper-parameters and values:

- Number of decision trees in [10, 50, 100, 200, 1000],
- Maximum number of features in [2, 5, 10, 21, 25, 50, 'sqrt'],
- Maximum tree depth in [10, 20, 50, None],
- Minimum number of samples for splitting an internal node in [1, 2, 5, 8],
- Minimum number of samples at the leaf node in [1, 2, 4, 8],
- Bootstrap in [True, False].

The most accurate results were achieved with 200 estimators, maximum features at 21, a maximum tree depth of 20, a minimum number of samples of 2, a minimum number of leaf samples of 1, and without bootstrapping.

2.2.2.7. Ridge Regression

A grid search optimization was applied to the following parameter values:

- alpha in [0, 0.01, 0.1, 0.2, 0.3, 0.4, 0.5, 0.6, 0.7, 0.8, 0.9, 1.0],
- Normalize parameter in [True, False],
- Computational routines solver in ['svd', 'cholesky', 'lsqr', 'sparse_cg', 'sag', 'saga', 'lbfgs'].

The most accurate results were achieved with alpha=0.1, normalize=True, and the lsqr solver.

2.2.2.8. Multi-layer Perceptron

We implemented a basic Multilayer Perceptron (MLP) classifier with two dense layers (‘ReLU’ activation function) and one output layer.

We tuned the model hyper-parameters over the following values:

- Having two, one, or no hidden layers with layer sizes in [
(resolution/2, resolution/4, resolution/8)
(resolution/2, resolution/4),
(resolution/2)] respectively, where resolution corresponds to the number of parcels in the atlas (e.g., 1000).

- Activation function for the hidden layer in ['identity', 'logistic', 'tanh', 'relu']
- Solver for weight optimization in ['lbfgs', sgd', 'adam']
- Alpha, the strength of the L2 regularization, in [0.0001, 0.05, 0.1]
- Learning rate in ['constant', 'adaptive']

The best results were achieved with a single hidden layer. The model output size is the number of decoding classes (21), the input size is the parcellation resolution for the input layer, and (resolution/2) is the size of the hidden layer. ReLU was selected as the hidden layer activation function, adam as the optimization algorithm and alpha = 0.05 for L2 regularization strength, with a constant learning rate.

2.2.2.9. Graph Convolutional Network

We adapted the Graph Convolutional Network (GCN) model from Zhang et al. (2021) to decode high-order cognitive functions from neural activity distributed across brain

networks. One of the main components of the GCN approach is the brain graph, which provides a network representation of brain organization. The brain graph, implemented as a connectivity matrix, associates graph vertices to brain regions, while graph edges represent anatomical or functional brain connections (Figure 2). The current GCN relied on a fixed human connectome as a static graph, which we generated by extracting time series from a full brain atlas for each subject. More specifically, we generated the brain graph from the weighted average of a complete run of the hcptrt using the ConnectivityMeasure class of the Nilearn package.

Our GCN implementation took a 1.49s (single volume) time window of fMRI signal as input and mapped this high-dimensional input onto a k-nearest-neighbor (k-NN) graph, which was generated by connecting each brain region to its k neighbors with the strongest connectivity. To define the brain regions, we tested several brain graph architectures based on three different predefined atlases of various dimensionalities, including MIST (122, 444), DiFuMo (128, 256, 512, 1024), and Schaefer (100, 400, 1000). Since the Schafer-1000 achieved the best results, we performed our experiment using this parcellation atlas.

The GCN model itself consisted of four ChebNet graph convolutional layers with 64 graph filters per layer, followed by a layer of global average pooling and an MLP with two fully connected layers before classification (21 classes; Figure 2). We used backpropagation to train the model in PyTorch. The network was trained for 10 epochs with batch sizes of 10. After each round of training, 10% of the data were used as a validation set to calculate the average accuracy and loss. After completing all epochs, we used the unseen test data (20%) to evaluate the performance of the final model.

Since tuning all GCN parameters with a full grid-search approach is computationally too expensive, we manually tried the following hyper-parameter values:

- Graph sparsification (k value) in [4, 8, 16],
- Weight regularization in [0, 5e-4, 0.1],
- Learning rate in [1e-3, 1e-4, 1e-5] and lr_scheduler with [ReduceLROnPlateau, StepLR],
- Using bias or not,
- Dropout in [0, 0.25, 0.5],
- Number of ChebNet graph convolutional layers in [2, 4, 6, 8],
- Number of ChebNet graph filters per layer in [16, 32, 64],
- ChebNet filter size in [1, 2, 3, 4],
- Number of fully connected layers in [1, 2, 3]
- Size of input sample for MLP hidden layer1 and layer 2 in [resolution size, 256] and [resolution size/2, 128] respectively,

The following parameters and values were chosen: (Figure 2)

- constraining the graph from clusters with 16 neighboring nodes with the strongest connectivity,
- 5e-4 for weight regularization,
- 1e-4 learning rate with lr_scheduler.ReduceLROnPlateau,
- Four ChebNet graph convolutional layers with 64 ChebNet graph filters of size 3 at each layer,
- A global average pooling layer after the four convolutional layers,
- MLP model with two hidden layers, resolution of chosen parcellation method as input, 21 task conditions as output, zero dropouts, and bias=True.

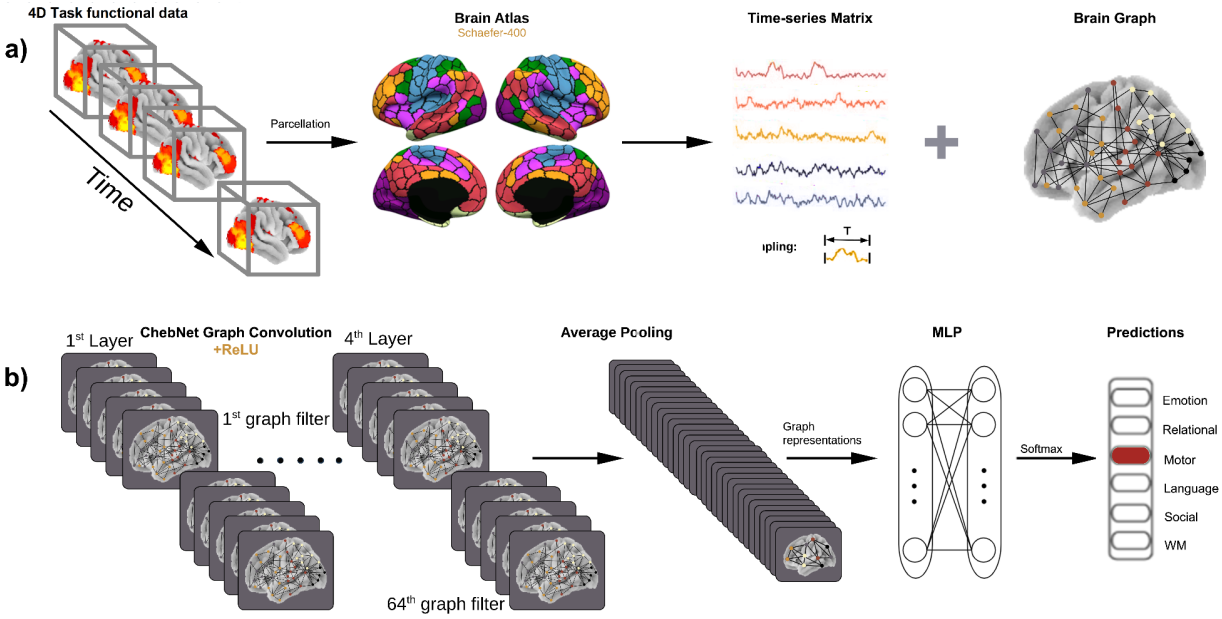


Figure 2.2. Schematic view of the analysis proposed for the GCN model. **a)** Bold time series are used to construct the brain graph, a representation of brain organization that associates nodes to brain regions (parcels) and edges to their functional connections. **b)** Our GCN model consisted of four graph convolutional layers with 64 ChebNet graph filters at each convolutional layer, followed by a layer of global average pooling and two fully connected (MLP) layers before classification.

2.2.2.10. Baseline

To have a baseline to evaluate model performances, we estimated the chance level using the ‘DummyClassifier’ class from Scikit-learn.

2.2.3. Permutation importance maps

We plotted permutation importance maps from the motor and WM tasks to evaluate features involved in the models’ decoding process. Those two functional tasks were selected since they were one of the easiest and the most challenging decoding tasks, respectively (Figure 7). We trained SVM (RBF kernel, Schaefer-100) models on fMRI data from runs of motor (700 brain volumes) and WM tasks (1936-2088 brain volumes). The features' importance for training the SVM model was estimated using the permutation importance package from Scikit-learn and means of feature importance over 40 repetitions were calculated. The importance maps were compared with the task-specific activation maps generated with the Nilearn first-level analysis package.

2.3. Results

2.3.1. Accurate decoding of cognitive states at the individual level

We assessed performance based on classification accuracy for single brain volumes from 21 different cognitive tasks across nine different classifiers, from simple SVM to state-of-the-art GCN. Our best results were achieved with Schaefer 1000 among all evaluated parcellations approaches. Generally, as was expected and in line with previous work (Hahn et al. 2022), we observed that higher spatial resolution leads to better decoding performance, with the exception of DiFuMo1024.

Figure 2.3 shows the accuracy of brain decoding models trained at the individual level for each task for each subject. All models showed above-chance performance (estimated ~7% with our baseline model) on decoding cognitive tasks. The top-performing method was SVM, reaching 79% prediction accuracy across the 21 task conditions (sub-01), followed by MLP and GCN. Performance varied substantially across subjects, ranging from 64% average prediction accuracy (sub-06, SVM) up to 79% (sub-01, SVM). This inter-subject variability was observed across all models, with sub-01 to sub-03 showing consistently higher decoding performance than sub-04 to sub-06. Multilayer perceptron, a simple and fast deep learning model, obtained competitive results ranging from 63%

(sub-06) to 78% (sub-01). Finally, state-of-the-art GCN also achieved high performance (63-77%), but not above the simpler algorithms.

Overall, we found it possible to achieve high brain decoding accuracy at the individual level, using a high-resolution brain parcel and a relatively simple method (SVM) as well as deep learning methods (MLP, GCN).

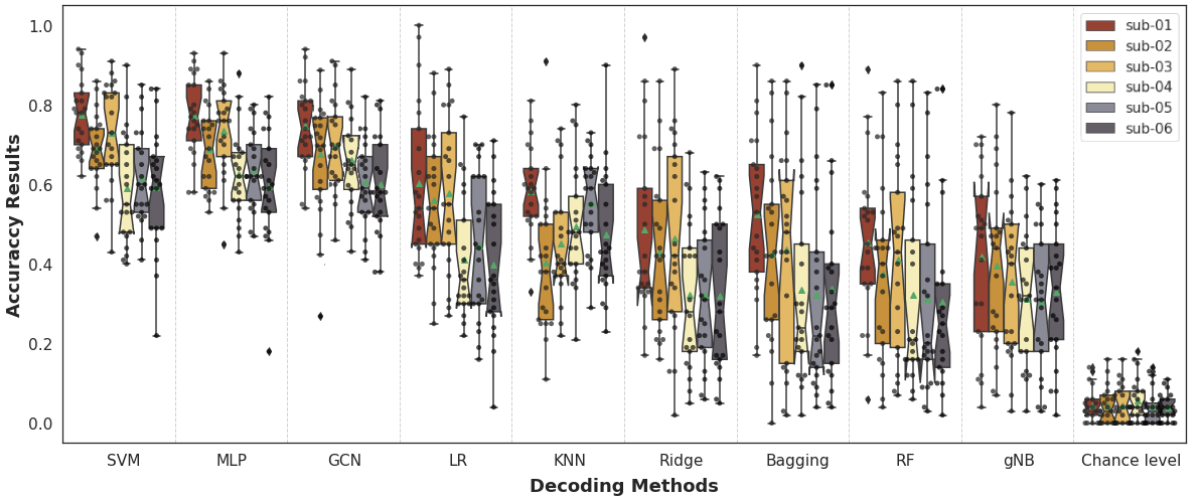


Figure 2.3. Accuracy of individual-wise brain decoding across 21 task conditions, as a function of the classifier algorithm for 6 subjects. Substantial variations in performance were observed across classifiers, with the top-performing method being SVM. The error bar shows the prediction accuracy ranges from all 21 different cognitive tasks.

2.3.2. Decoding cognitive states using different parcellation atlases

To investigate the impact of parcellation atlases and dimensionality, we trained and tested our individual decoding models using several parcellation approaches and resolutions, including MIST (122, 444), DiFuMo (128, 256, 512, 1024), and Schaefer (100, 400, 1000). Figure 4 shows brain decoding results for the tuned SVM model, which performed best among all models, for different parcellation atlases and dimensions for all six subjects. We found that higher resolution generally yields better results (with the exception of DiFuMo-1024), with Schaefer-1000 achieving the highest accuracy [ranging from 79% (sub-01) to 67% (sub-04)], followed by MIST-444 [ranging from 76% (sub-01) to 63% (sub-05)]. In scenarios where memory and computation time are of concern, MIST-444 offers a reasonable alternative to higher-resolution atlases (Figure 4).

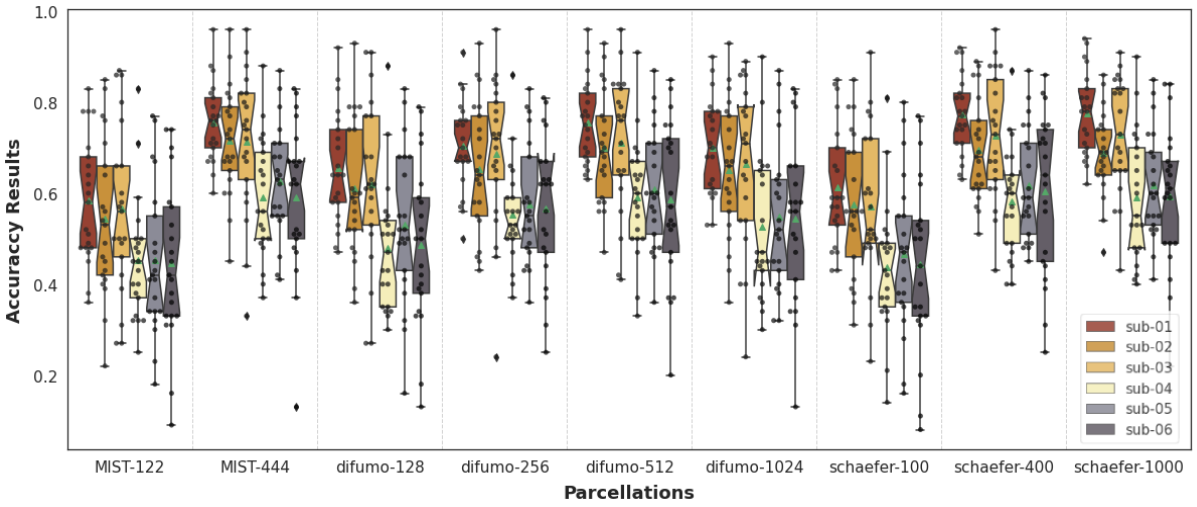


Figure 2.4. SVM model prediction accuracy results per subject for 21 task conditions, using different parcellation approaches.

2.3.3. Between-subject decoding

In order to explore the generalizability of intra-subject models, we tested SVM models trained on one subject on brain volumes from the other subjects. We chose SVM since this model returned the highest average accuracy results (RBF kernel, Schaefer-1000). The model was trained on 80% of data from one subject and was tested using 20% of all data points randomly selected from other subjects. We compared between-subject prediction accuracy from all pairs of subjects with our within-subject results. As shown in Figure 5, within-subject results are substantially higher than all other between-subject results. Figure 6 shows the prediction accuracy of each pair of subjects (6a), and the differences in accuracy between inter and intra-subject decoding (6b).

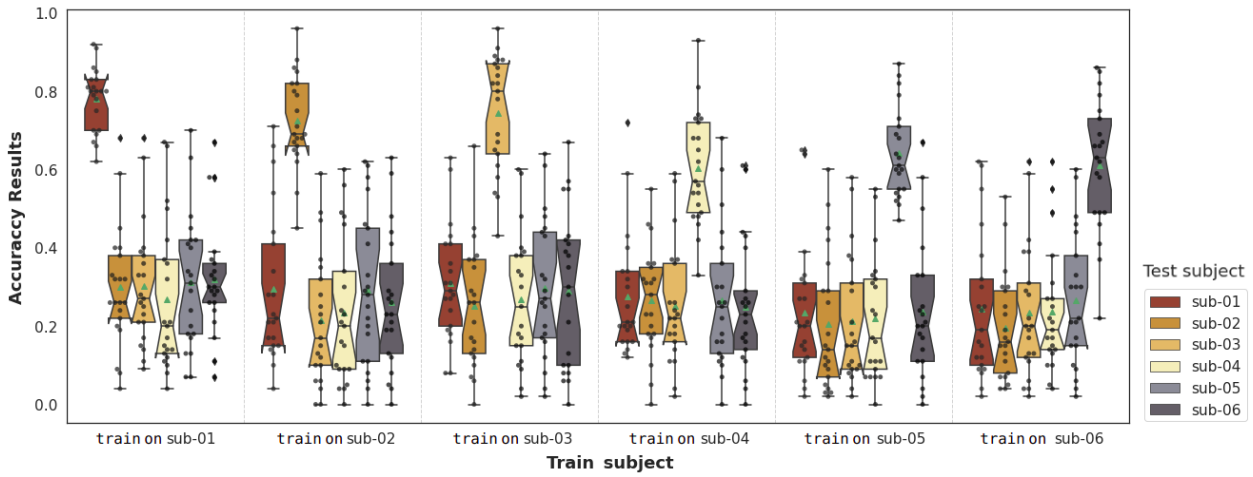


Figure 2.5. Prediction accuracy from the SVM model across 21 task conditions, trained on one subject and tested on brain volumes from each of the six subjects.

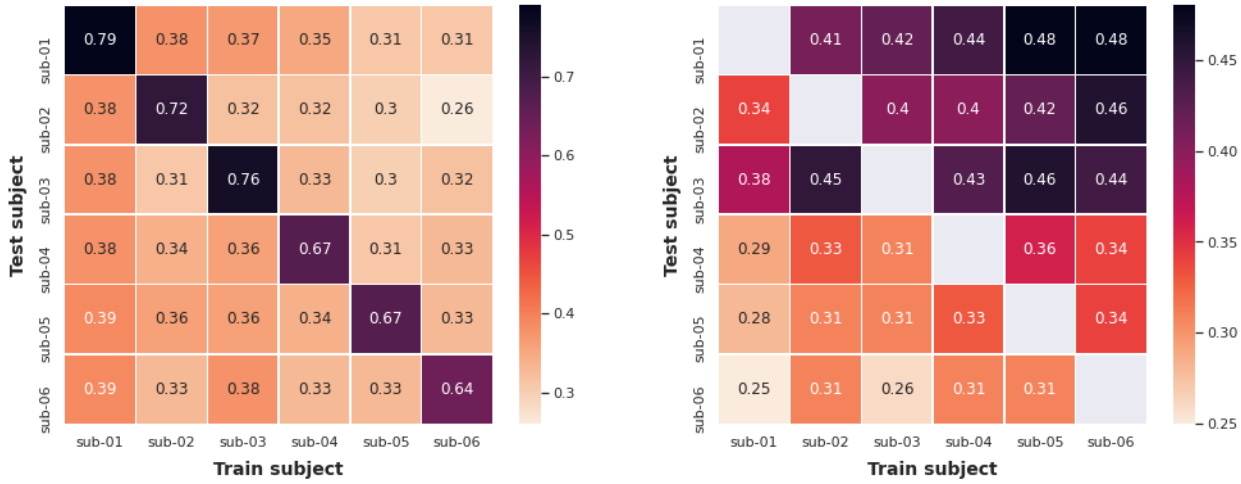


Figure 2.6. a) Mean accuracy scores across 21 task conditions from the SVM model trained and tested on each pair of subjects. **b)** Differences in accuracy scores between inter and intra-subject decoding.

2.3.4. Brain permutation importance maps

To assess which brain regions were important for the brain decoding approach, we first retrained brain decoding models for separate groups of tasks from the HCP task battery, using the SVM model Radial Basis Function (RBF) kernel and the Schaeffer 100 brain

parcellation. Once a model was trained for a specific group of tasks, we used an interpretation technique called permutation importance to highlight which features (brain regions) were important for classification in that group of tasks. We present here the importance map of the task group with the highest decoding accuracy, motor (handL, handR, footR, footL, and tongue), as well as the most challenging group of tasks, working memory (body0b, bod20b, face0b, face2b, place0b, place2b, tool0b, tool2b). This choice of tasks was similar to what was implemented by (3). Figure 7a shows the permutation importance maps for motor tasks in all six participants. The maps are highly consistent, with prominent contributions from the dorsal sensorimotor cortex. The ventral and medial sensorimotor cortices also show contributions that are more marginal and variable across subjects. Figure 7b shows the permutation importance maps for working memory tasks in all six participants. The maps are mostly variable across subjects and involve a wide range of cortical territories. The left dorsolateral prefrontal cortex is for instance a contributor for subjects 2, 3, and 5, while absent from the maps of subjects 1, 4, and 6. The regions with the highest contributions include the high-order visual processing cortex and regions from the visual ventral stream. Those regions are consistently found in all 6 participants, which is consistent with the group-level brain decoding saliency maps presented by Zhang et al. (2021), and expected in a visual working memory task. Results for four other task groups can be found in Supplementary Material; each of these groups led to highly heterogeneous maps across participants. Overall, we found evidence of the recruitment of features in brain areas known for their biological involvement in the corresponding tasks (motor cortex in motor tasks, ventral visual areas in a visual working memory task) as well as considerable inter-individual variations (with the exception of motor tasks). These findings support both that the models learned biologically meaningful features, and that these models were largely subject-specific, consistent with our prior inter-subject decoding experiment.

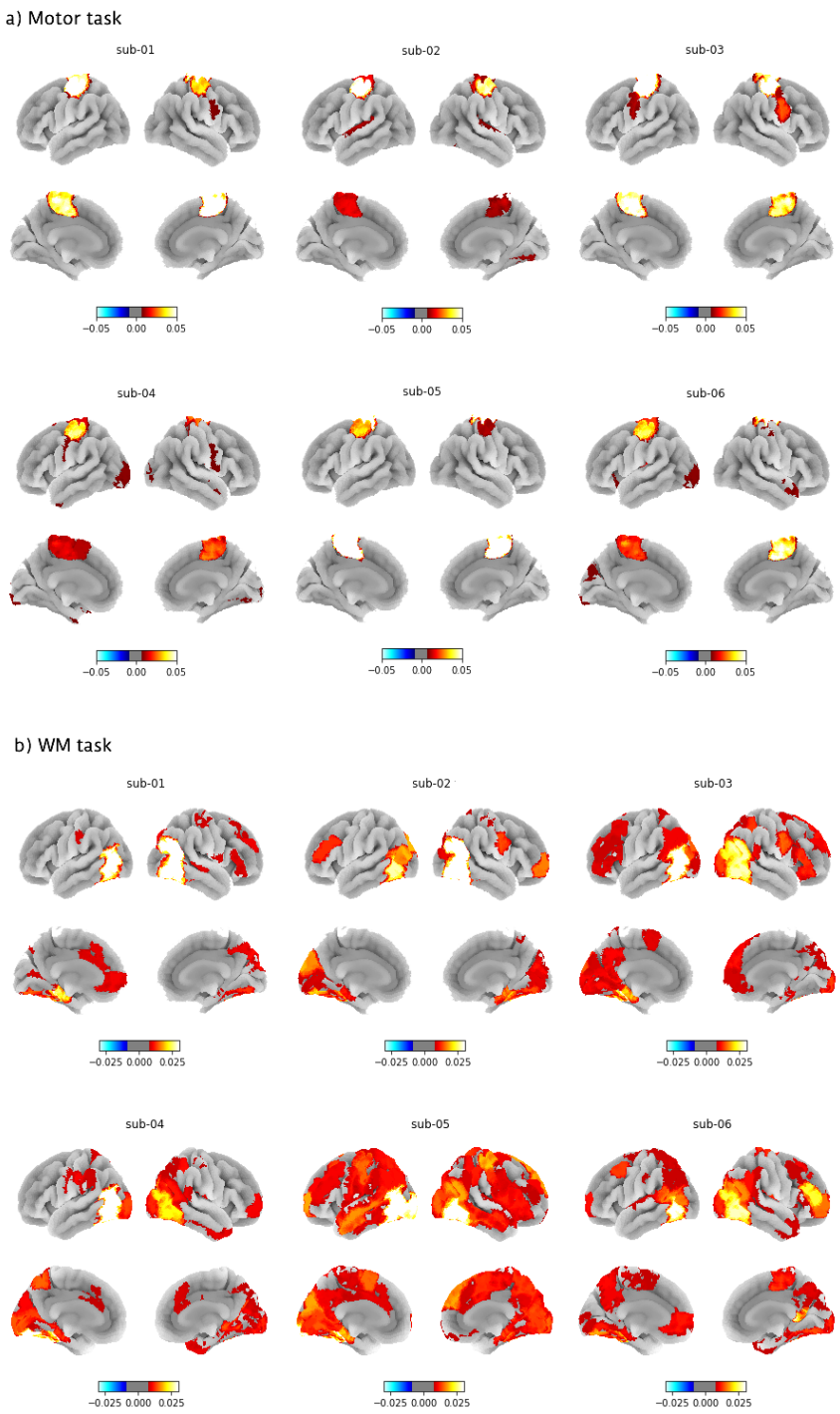


Figure 2.7. Feature importance maps for the decoding of motor and WM tasks. We generated the permutation importance feature maps, averaged across sub-conditions, from the prediction accuracy results of our parcel-vised SVM (RBF kernel, Schaefer-100) model. These maps show parcels with high contributions to the SVM model classification. **a)** Demonstrates the maps for the motor task, plotted with the threshold of

0.008 and the upper bound of 0.05. **b)** Working memory task maps, plotted with the threshold of 0.008 and the upper bound of 0.03.

2.3.5. Decoding functional states using different parcellation atlases of varying resolution

The confusion matrix shown in Figure 2.8 depicts the mean predictions for all six subjects from our SVM model (RBF kernel, Schaefer-1000 parcellation). The block diagonal architecture indicates that the majority of cognitive tasks were accurately identified among the 21 conditions. The most common misclassifications were observed within cognitive domains tested with comparable experimental designs. Similar to Zhang et al. (2021), the language tasks (story vs math) and motor tasks (left/right hand, left/right foot, and tongue) were the easiest to decode, while working memory tasks (body0b/2b, face0b/2b, place0b/2b, and tool0b/2b) were the least recognizable conditions for the decoder.

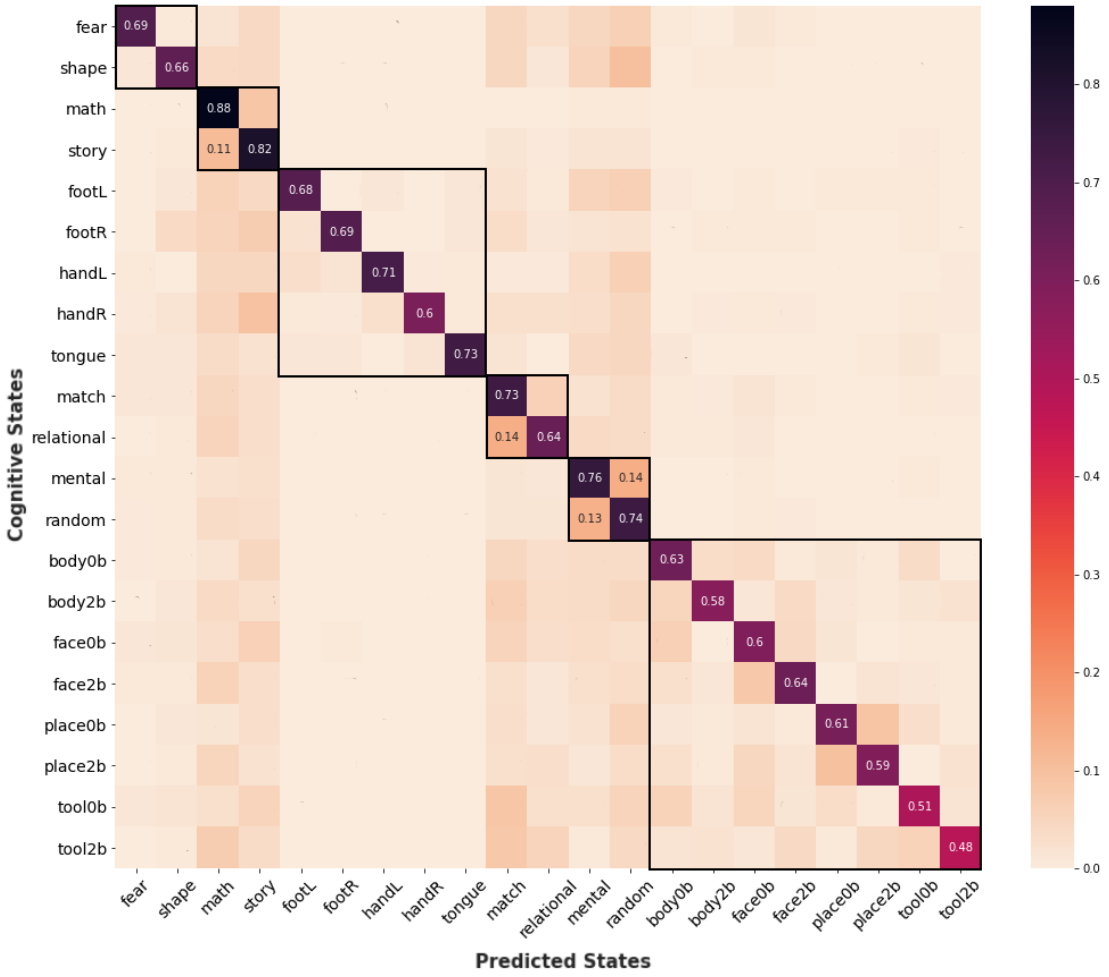


Figure 2.8. The mean confusion matrix shows accuracy averaged across all subjects for the SVM model (RBF kernel, Schaefer-1000 parcellation). Each row and column represents a cognitive state, whereas each matrix cell shows the recall score, the model's correct positive predictions out of actual positives. The matrix illustrates a nice block diagonal architecture that shows how the majority of functional tasks were classified accurately, among which the language tasks achieved the highest sensitivity. The functional conditions of the motor task achieved some of the highest prediction accuracies, except for one subject (sub-04; supplementary). Working memory tasks were the most challenging conditions and obtained the lowest results. These results are consistent with the previous work on the entire HCP sample by Zhang et. al (2021).

2.3.6. Decoding using a session-based split for test and training sets

Since we trained classifiers for predicting functional tasks based on data from a single individual, we added an extra experiment to ensure there is no risk of information leakage. For this purpose, we evaluated our best model (SVM with RBF kernel) using leave-one-session-out cross-validation. The model was trained on a subset of data from a given participant (10-11 runs from 13-14 available runs) and tested it using two independent sessions (three runs) from the same subject.

The confusion matrix shown in Figure 2.8 demonstrates the mean prediction accuracy results for all six subjects from our SVM model (RBF kernel, Schaefer-1000 parcellation). Although the results are slightly lower than the previous approach, the nice block diagonal architecture indicates that the majority of cognitive tasks were accurately identified. These results are consistent with the earlier experiment, yet most misclassifications happened within cognitive tasks with similar experimental designs.

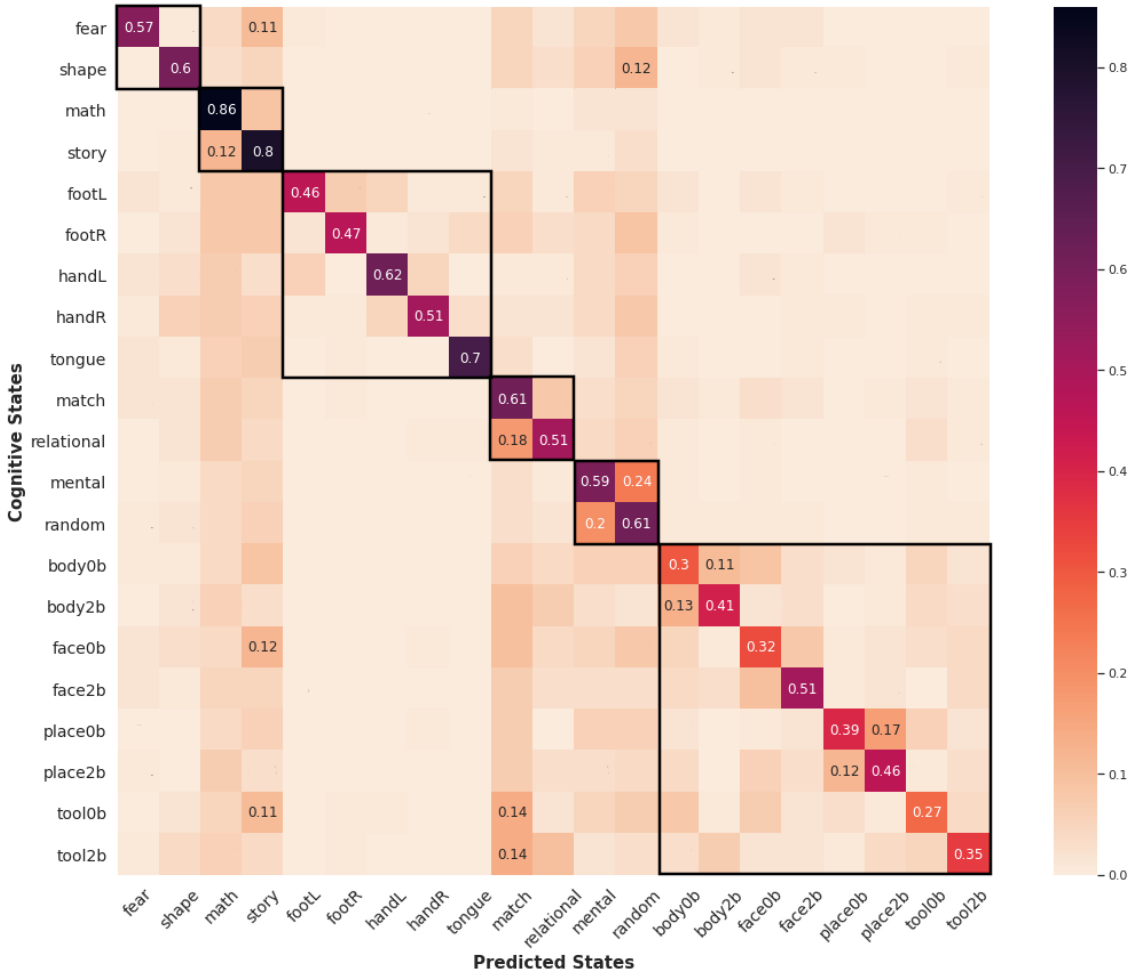


Figure 2.9. The mean confusion matrix shows accuracy averaged across all subjects for the SVM model (RBF kernel, Schaefer-1000 parcellation). Each row and column represents a cognitive state, whereas each matrix cell shows the recall score, the model's correct positive predictions out of actual positives.

2.4. Discussion

In this study, we built accurate intra-subject brain decoding models using CNeuroMod's hcptrt, a dense individual fMRI dataset. We classified the brain dynamics of 21 experimental conditions across six cognitive tasks using single volumes (sampled every 1.49s) of fMRI signal and reached ~80% accuracy on unseen volumes using less than 8 hours of single-subject data.

Building accurate individual brain decoding

The majority of functional brain imaging research focuses on extracting patterns from group-averaged brain data. By definition, group averaging captures only fundamental, systematic principles of human brain organization and ignores individual brain variance. Thus, classifying cognitive states across individuals requires having access to massive input data to achieve high accuracy. In this work, we showed that with sufficient and reliable data from a single subject, we could achieve high prediction accuracy with a wide diversity of decoding models. Our findings demonstrate the effectiveness of individual decoding. While it is possible that better results could be achieved with a larger training set, especially for higher capacity nonlinear models like GCN, the current results nevertheless approach the decoding accuracy achieved by Zhang and colleagues (2021, 2022) for HCP task conditions using data from a vast group of subjects (over 200 hours of training data from HCP sample, compared to our <8 hours/subject). The temporal resolution in the HCP dataset (TR=0.72s) is higher than the HCPtrt dataset (TR=1.49s). The shorter repetition time makes single-volume decoding using the HCP dataset more challenging than the HCPtrtr dataset. However, this shorter repetition time leads to acquiring more brain volume in each time unit. Thus, we have almost a double number of brain volumes for the same duration of data in the HCP dataset compared to HCPtrt, which can be beneficial for training decoding models.

Finding best-adapted models and parcellations for individual brain decoding

We achieved similar performance with SVM, MLP, and GCN, demonstrating two points. First, those simple algorithms are sufficient to extract cognitively relevant features from single-subject data. And second, comparatively “modest” amounts of training data are sufficient to train deep graph convolutional models. It is plausible that deep learning methods may gain an advantage over more naive classifiers when they can extract features from larger datasets, as suggested by results from Zhang et al., (2021) on the entire HCP set. On the other hand, using simpler models such as SVM and MLP is advantageous when there are time and memory constraints. We also observed that classification improved with higher resolution parcellation (with the exception of the highest dimensionality (1024) DiFuMo atlas, which achieved worse results than 512 DiFuMo). Overall, the DiFuMo atlases of functional nodes did not perform as well as other parcellations despite their greater data compression. One possible explanation for this poor performance could be the data compression technique used by DiFuMo, which relies on spatial ridge regression to extract functional signals.

Are brain decoders really individual?

We also showed that individual brain decoders learn individual-specific patterns by testing trained classifiers across participants. Although results varied across subjects, we observed a large performance reduction for all inter-subject classifications compared to intra-subject results for which models were trained and tested on the same participant. Specifically, average accuracy drops ranged from 25% to 48% when individual models were tested on a different subject, demonstrating individual specificity, which is expected according to previous reports of superior within-subject compared to between-subject decoding (Gordon et al. 2017; Gratton et al. 2018).

Are features useful for brain decoding similar across subjects?

We plotted permutation importance maps for the working memory (WM) and motor tasks to assess whether brain features that supported classification performance were localized in regions known to support these different cognitive domains, as reflected in the hemodynamic response to each task. The maps reveal task-relevant features that are common across subjects, as well as evidence for individual specificity, in line with studies on functional human brain network variants for each individual (Gordon et al. 2017; Gratton et al. 2018).

Limitations

We performed subject-level brain decoding over ~14 repetitions of the HCP functional localizers per subject, totaling around 8h of fMRI data per participant. Our approach did not consider the effect of task repetition on participants' performance. Repeating a task several times increases the probability of fatigue, boredom, drowsiness, and practice effect, among other effects. As we show in the current work, one of the notable benefits of using individual brain decoding is that we can achieve high prediction results from fewer hours of training data without struggling with variability in functional and anatomical systems across human brains. Our work has not addressed whether increased repetitions of cognitive tasks per subject impact individual cognitive brain decoding and whether repetition-induced variability may lead to irreproducible results across folds in the limited size of the current fMRI dataset.

Future directions

Our findings demonstrate the potential of dense individual datasets to train brain decoders, and they highlight the need to further develop this line of work. The subject-level approaches could be used as a benchmark for group model studies that apply functional hyper-realignment techniques to improve inter-subject generalization, as proposed by Bazeille and colleagues (2021). Another potential next step for our work would be to apply our models to individual datasets that include a greater variety of cognitive tasks and conditions, and possibly explore how the size of the training dataset benefits different model architectures.

Acknowledgments

This work is part of the Courtois NeuroMod Project supported by the Courtois foundation. PB is a senior fellow of the “Fonds de recherche en Santé - Québec”.

Chapter 3

Conclusion and Future Work

In this study, we built accurate intra-subject brain decoding models using CNeuroMod's hcptrt, a dense individual fMRI dataset. We classified the brain dynamics of 21 experimental conditions across six cognitive tasks using single volumes (sampled every 1.49s) of fMRI signal and reached ~80% accuracy on unseen volumes using less than 8 hours of single-subject data (down to ~60% with session-level cross-validation).

We showed that with sufficient and reliable data from a single subject, we could achieve high prediction accuracy with a wide diversity of decoding models. We worked on a hard classification task, which is to discriminate between 21 different conditions using a single fMRI brain volume. Many previous studies on HCP tasks have looked at either multiple time points or activation maps derived through averaging (H. Li and Fan 2019; X. Li et al. 2021; Gao et al. 2019). Some studies also restricted the number of labels, e.g. only discriminating between motor conditions, where it is possible to achieve over 95% accuracy (Y. Zhang, Farrugia, and Bellec 2022). The study by Porter and colleagues did reach over 90% accuracy with individualized models but it considered a 5-class discrimination task and also averaged data across multiple time points. We selected this challenging benchmark because we were able to accumulate a large number of samples to train individual models, and also because it leaves room for improvement with future methods.

Our findings on this challenging benchmark demonstrate the effectiveness of individual decoding. While it is possible that better results could be achieved with a larger training set, especially for higher capacity nonlinear models like GCN, the current results nevertheless approach the decoding accuracy achieved by Zhang and colleagues (2021, 2022) for HCP task conditions using data from a vast group of subjects (over 1000 hours of fMRI training data from the group HCP sample, compared to our <8 fMRI of hours/subject). In other words, there is a difference of two orders of magnitude in the size of the training data between the group HCP benchmark of Zhang and colleagues (2021) and the present individual benchmark, but only a marginal difference in the performance of the resulting brain decoding models.

It should be noted that these two benchmarks are not directly comparable: for instance, the temporal resolution in the HCP dataset (TR=0.72s) is higher than the HCPtrt dataset (TR=1.49s). The shorter repetition time makes single-volume decoding using the HCP dataset more challenging than the hcptrt dataset. However, this shorter repetition time leads to acquiring more brain volume in each time unit. Thus, the HCP dataset features almost a double number of brain volumes for the same duration of data compared to hcptrt, which can be beneficial for training decoding models in addition to a large number of subjects. There are other methodological differences as well, for example regarding the choice of preprocessing tools. We can still note that many studies have used the HCP group benchmark and found consistent performance with heterogeneous approaches e.g. (H. Li and Fan 2019; X. Li et al. 2021; Gao et al. 2019). Our own work also shows consistent performance across several models (SVM, MLP, GCN). Overall, our results show that individual brain models reach a competitive performance with group decoding models. This level of performance is likely robust to methodological differences, although this would need to be further tested in future work.

We achieved similar performance with SVM, MLP, and GCN, demonstrating two points. First, classic machine learning algorithms are sufficient to extract cognitively relevant features from single-subject data. And second, comparatively “modest” amounts of training data (~7h) are sufficient to train individual deep graph convolutional models. It is plausible that deep learning methods may gain an advantage over more basic classifiers when they can extract features from larger datasets, as suggested by results from Zhang and colleagues, (2021) on the entire HCP sample. We also observed that classification improved with higher-resolution parcellation. This was expected to a degree, as high resolution leads to an improved homogeneity of brain parcels (Urchs and colleagues, 2021), and was also reported previously to lead to improved prediction power in other benchmarks (Dadi et al. 2020). A notable exception is the highest dimensionality (1024) of the DiFuMo atlas, which achieved worse results than 512 DiFuMo. Overall, the DiFuMo atlases of functional nodes did not perform as well as other parcellations despite their superior data compression quality (Dadi et al. 2020). One possible explanation for this poor performance could be the data compression technique used by DiFuMo, which relies on spatial ridge regression to extract functional signals instead of a simple averaging as implemented with “hard” parcellations such as Glasser or MIST. Moreover, since some types of parcellations are suited for group-level decoding and some are more reliable for individual decoding studies, so exploring the effect of different types of parcellation could be another future approach. For instance, the study by Porter and colleagues (Porter et al. 2022) reported improved decoding results when using individual-specific brain parcellation, rather than group parcellations. We did not include this type of parcellation in the present benchmark but would be an important direction for future work.

We also showed that individual brain decoders learn individual-specific patterns by testing trained classifiers across participants. Although results varied across subjects, we observed a large performance reduction for all inter-subject classifications compared to intra-subject results for which models were trained and tested on the same participant. Specifically, average accuracy drops ranged from 25% to 48% when individual models were tested on a different subject, demonstrating individual specificity, which is expected according to previous reports of superior within-subject compared to between-subject decoding (Gordon et al. 2017; Gratton et al. 2018). This dramatic drop in performance is also in line with the previous study by Porter and colleagues (2022).

We plotted permutation importance maps for the working memory (WM) and motor tasks to assess whether brain features that supported classification performance were localized in regions known to support these different cognitive domains, as reflected in the hemodynamic response to each task, and also to check whether similar features emerged across subjects. The maps reveal task-relevant features that are common across subjects, as well as evidence for individual specificity, in line with studies on functional human brain network variants for each individual (Gordon et al. 2017; Gratton et al. 2018). The most consistent feature maps were obtained for the motor tasks, which are also the easiest to decode in the group HCP sample. The working memory, which is much more challenging to decode, showed much more variations across subjects. This may reflect more noise or instability in the model, as well as more individual-specific features. We did not have a sufficient amount of individual data to test for the reliability of the feature maps, so we cannot draw specific conclusions from our experiments. It would also be possible to use more advanced interpretation techniques to identify features for each task condition separately, as was done for example (Y. Zhang, Farrugia, and Bellec 2022), instead of grouping them by domains as we did here.

Limitations

A limitation of our study is the potential for habituation effects, as participants were exposed to the same tasks multiple times. We indeed performed subject-level brain decoding with ~14 repetitions of the HCP functional localizers per subject, totaling around ~7h of fMRI data per participant. Our approach did not consider the effect of task repetition on participants' performance. Repeating a task several times increases the probability of fatigue, boredom, drowsiness, and practice effect, among other effects. This may be particularly important for conditions such as math, which are challenging and which level of difficulty dynamically adapts to the performance of the subject. These types of conditions could lead to learning effects or changes in cognitive strategies over time in some subjects. As we show in the current work, one of the notable benefits of using individual brain decoding is that we can achieve high prediction results from fewer hours of training data without struggling with variability in functional and anatomical systems across

human brains. Our work has not addressed whether increased repetitions of cognitive tasks per subject impact individual cognitive brain decoding and whether repetition-induced variability may lead to irreproducible results across folds in the limited size of the current fMRI dataset. The study by Zhang and colleagues (Y. Zhang et al. 2021) examined the correlation between decoding performance and behavior, such as the accuracy of responses and reaction times.

Future directions

Our findings demonstrate the potential of dense individual datasets to train brain decoders, and they highlight the need to further develop this line of work. The subject-level approach could be used as a benchmark for group model studies that apply functional hyper-realignment techniques to improve inter-subject generalization, as proposed by Bazeille and colleagues (Bazeille et al. 2021). Hyper-realignment typically uses movie data to better align brain representations between subjects, with the objective of improving accuracy on downstream decoding tasks. The CNeuroMod project contains a broad range of datasets besides hcptrt, which can be used to assess the impact of hyper-realignment. For instance, movie10 is a collection of four different full movies watched by each participant, and in the friends, dataset participants watched the seasons of the Friends TV show. This extensive naturalistic data could be used to explore the impact of the size of the training dataset, as well as the exact nature of the stimuli being tested on hyper-realignment, and how such realignment techniques could benefit different model architectures.

Moreover, hcptrt could be used to test the transferability of brain activity models. For example, Thomas and colleagues (Thomas, Ré, and Poldrack 2022) trained a self-supervised deep learning model of brain activity on a large and diverse set of group fMRI data, and showed that this model could be successfully transferred for brain decoding across different contexts. This type of approach can be applied with the CNeuroMod sample entirely at the individual level, e.g by training an individual model on the movie10 dataset, and then transfer to brain decoding on hcptrt. Besides, another type of transfer learning is to generalize brain decoding models across subjects, tasks, and datasets with different characteristics, such as acquisition sequences (including TR), and scanners. At least three datasets are currently available featuring the HCP tasks: the original HCP sample, with a large number of subjects (N=1200), as well as two deep fMRI datasets, IBC and CNeuroMod. Combining different participants and tasks across those three rich, complementary datasets opens exciting perspectives to better understand the generalizability of brain decoding.

Conclusion

This work is a benchmark for brain decoding models trained at the individual level based on some of the most popular cognitive tasks currently used for brain decoding, as designed by HCP (19). We attempted a challenging classification task: decoding 21 conditions simultaneously using a single fMRI brain volume. We optimized a wide variety of model architectures and showed that, with sufficient training data from a single subject (~8 h), we could achieve high prediction accuracy in the range of 54%-62%. This accuracy was similar, or even higher, than what was achieved at the group level on the full HCP sample (n>1000) in the benchmark of Zhang and colleagues (Y. Zhang et al. 2021). Our results thus confirm the feasibility of training accurate brain decoding models with a limited amount of fMRI data, provided the model is trained purely at the individual level. Those individual models however did not generalize well to other participants and appeared to capture idiosyncratic brain features. This dataset and our benchmark can be useful in the future to assess brain decoding models in order to reveal individual-specific aspects of brain organization, in combination with techniques such as hyper-alignment or transfer learning. The dense individual functional magnetic resonance imaging dataset used in this work, hcptrt, is publicly available as part of the Courtois NeuroMod databank (https://cneuromod.ca).

References

Allen, Emily J., Ghislain St-Yves, Yihan Wu, Jesse L. Breedlove, Jacob S. Prince, Logan T. Dowdle, Matthias Nau, et al. 2022. “A Massive 7T fMRI Dataset to Bridge Cognitive Neuroscience and Artificial Intelligence.” *Nature Neuroscience* 25 (1): 116–26.

Bartley, Jessica E., Emily R. Boeving, Michael C. Riedel, Katherine L. Bottenhorn, Taylor Salo, Simon B. Eickhoff, Eric Brewe, Matthew T. Sutherland, and Angela R. Laird. 2018. “Meta-Analytic Evidence for a Core Problem Solving Network across Multiple Representational Domains.” *Neuroscience and Biobehavioral Reviews* 92 (September): 318–37.

Bazeille, Thomas, Elizabeth DuPre, Hugo Richard, Jean-Baptiste Poline, and Bertrand Thirion. 2021. “An Empirical Evaluation of Functional Alignment Using Inter-Subject Decoding.” *NeuroImage* 245 (December): 118683.

Bergmann, Johanna, Andrew T. Morgan, and Lars Muckli. 2019. “Two Distinct Feedback Codes in V1 for ‘real’ and ‘imaginary’ Internal Experiences.” *bioRxiv*. bioRxiv. https://doi.org/10.1101/664870.

Bishop, Christopher M., and of Neural Computing Christopher Bishop. 1995. *Neural Networks for Pattern Recognition*. Clarendon Press.

Bi, Xia-An, Yang Wang, Qing Shu, Qi Sun, and Qian Xu. 2018. “Classification of Autism Spectrum Disorder Using Random Support Vector Machine Cluster.” *Frontiers in Genetics* 9 (February): 18.

Cortes, Corinna, and Vladimir Vapnik. 1995. “Support-Vector Networks.” *Machine Learning* 20 (3): 273–97.

Dadi, Kamalaker, Gaël Varoquaux, Antonia Machlouzarides-Shalit, Krzysztof J. Gorgolewski, Demian Wassermann, Bertrand Thirion, and Arthur Mensch. 2020. “Fine-Grain Atlases of Functional Modes for fMRI Analysis.” *NeuroImage* 221 (November): 117126.

“Datasets — Courtois NeuroMod 2020-Beta Documentation.” n.d. Accessed September 20, 2022. https://docs.cneuromod.ca/en/latest/DATASETS.html.

Défossez, Alexandre, Charlotte Caucheteux, Jérémy Rapin, Ori Kabeli, and Jean-Rémi King. 2022. “Decoding Speech from Non-Invasive Brain Recordings.” *arXiv [eess.AS]*. arXiv. http://arxiv.org/abs/2208.12266.

Eslami, Taban, and Fahad Saeed. 2019. “Auto-ASD-Network: A Technique Based on Deep Learning and Support Vector Machines for Diagnosing Autism Spectrum Disorder Using fMRI Data.” In *Proceedings of the 10th ACM International Conference on Bioinformatics, Computational Biology and Health Informatics*, 646–51. BCB ’19.

New York, NY, USA: Association for Computing Machinery.

Esteban, Oscar, Christopher J. Markiewicz, Ross W. Blair, Craig A. Moodie, A. Ilkay Isik, Asier Erramuzpe, James D. Kent, et al. 2019. “fMRIPrep: A Robust Preprocessing Pipeline for Functional MRI.” *Nature Methods* 16 (1): 111–16.

Gao, Yufei, Yameng Zhang, Hailing Wang, Xiaojuan Guo, and Jiacai Zhang. 2019. “Decoding Behavior Tasks From Brain Activity Using Deep Transfer Learning.” *IEEE Access* 7: 43222–32.

Gilbert, Sam J., and Hoki Fung. 2018. “Decoding Intentions of Self and Others from fMRI Activity Patterns.” *NeuroImage* 172 (May): 278–90.

Glasser, Matthew F., Stephen M. Smith, Daniel S. Marcus, Jesper L. R. Andersson, Edward J. Auerbach, Timothy E. J. Behrens, Timothy S. Coalson, et al. 2016. “The Human Connectome Project’s Neuroimaging Approach.” *Nature Neuroscience* 19 (9): 1175–87.

Gordon, Evan M., Timothy O. Laumann, Adrian W. Gilmore, Dillan J. Newbold, Deanna J. Greene, Jeffrey J. Berg, Mario Ortega, et al. 2017. “Precision Functional Mapping of Individual Human Brains.” *Neuron* 95 (4): 791–807.e7.

Grand, Gabriel, Idan Asher Blank, Francisco Pereira, and Evelina Fedorenko. 2018. “Semantic Projection: Recovering Human Knowledge of Multiple, Distinct Object Features from Word Embeddings.” *arXiv [cs.CL]*. arXiv. http://arxiv.org/abs/1802.01241.

Gratton, Caterina, Timothy O. Laumann, Ashley N. Nielsen, Deanna J. Greene, Evan M. Gordon, Adrian W. Gilmore, Steven M. Nelson, et al. 2018. “Functional Brain Networks Are Dominated by Stable Group and Individual Factors, Not Cognitive or Daily Variation.” *Neuron* 98 (2): 439–52.e5.

Hacker, Carl D., Timothy O. Laumann, Nicholas P. Szrama, Antonello Baldassarre, Abraham Z. Snyder, Eric C. Leuthardt, and Maurizio Corbetta. 2013. “Resting State Network Estimation in Individual Subjects.” *NeuroImage* 82 (November): 616–33.

Hahn, Sage, Max M. Owens, Dekang Yuan, Anthony C. Juliano, Alexandra Potter, Hugh Garavan, and Nicholas Allgaier. 2022. “Performance Scaling for Structural MRI Surface Parcellations: A Machine Learning Analysis in the ABCD Study.” *Cerebral Cortex* , March. https://doi.org/10.1093/cercor/bhac060.

Han, Kuan, Haiguang Wen, Junxing Shi, Kun-Han Lu, Yizhen Zhang, Di Fu, and Zhongming Liu. 2019. “Variational Autoencoder: An Unsupervised Model for Encoding and Decoding fMRI Activity in Visual Cortex.” *NeuroImage* 198 (September): 125–36.

Hanson, Stephen José, and Yaroslav O. Halchenko. 2008. “Brain Reading Using Full Brain Support Vector Machines for Object Recognition: There Is No ‘Face’ Identification Area.” *Neural Computation* 20 (2): 486–503.

Haxby, James V., Andrew C. Connolly, and J. Swaroop Guntupalli. 2014. “Decoding Neural

Representational Spaces Using Multivariate Pattern Analysis." *Annual Review of Neuroscience* 37 (June): 435–56.

Haxby, James V., J. Swaroop Guntupalli, Andrew C. Connolly, Yaroslav O. Halchenko, Bryan R. Conroy, M. Ida Gobbini, Michael Hanke, and Peter J. Ramadge. 2011. "A Common, High-Dimensional Model of the Representational Space in Human Ventral Temporal Cortex." *Neuron* 72 (2): 404–16.

Haxby, J. V., M. I. Gobbini, M. L. Furey, A. Ishai, J. L. Schouten, and P. Pietrini. 2001. "Distributed and Overlapping Representations of Faces and Objects in Ventral Temporal Cortex." *Science* 293 (5539): 2425–30.

Haynes, John-Dylan, Katsuyuki Sakai, Geraint Rees, Sam Gilbert, Chris Frith, and Richard E. Passingham. 2007. "Reading Hidden Intentions in the Human Brain." *Current Biology: CB* 17 (4): 323–28.

Horikawa, Tomoyasu, and Yukiyasu Kamitani. 2017. "Generic Decoding of Seen and Imagined Objects Using Hierarchical Visual Features." *Nature Communications* 8 (May): 15037.

Kasliwal, Manish Kumar. 2013. "Functional Neuroimaging: Current Status." *OMICS Journal of Radiology* 01 (04). https://doi.org/10.4172/2167-7964.1000e111.

Li, Hongming, and Yong Fan. 2019. "Interpretable, Highly Accurate Brain Decoding of Subtly Distinct Brain States from Functional MRI Using Intrinsic Functional Networks and Long Short-Term Memory Recurrent Neural Networks." *NeuroImage* 202 (November): 116059.

Li, Xiaoxiao, Yuan Zhou, Nicha Dvornek, Muhan Zhang, Siyuan Gao, Juntang Zhuang, Dustin Scheinost, Lawrence H. Staib, Pamela Ventola, and James S. Duncan. 2021. "BrainGNN: Interpretable Brain Graph Neural Network for fMRI Analysis." *Medical Image Analysis* 74 (December): 102233.

Naselaris, Thomas, Emily Allen, and Kendrick Kay. 2021. "Extensive Sampling for Complete Models of Individual Brains." *Current Opinion in Behavioral Sciences* 40 (August): 45–51.

Nishimoto, Shinji, An T. Vu, Thomas Naselaris, Yuval Benjamini, Bin Yu, and Jack L. Gallant. 2011. "Reconstructing Visual Experiences from Brain Activity Evoked by Natural Movies." *Current Biology: CB* 21 (19): 1641–46.

Pedregosa, Varoquaux, and Gramfort. n.d. "Scikit-Learn: Machine Learning in Python." *Of Machine Learning …*. https://www.jmlr.org/papers/volume12/pedregosa11a/pedregosa11a.pdf?ref=https://githubhelp.com.

Pinho, Ana Luísa, Alexis Amadon, Baptiste Gauthier, Nicolas Clairis, André Knops, Sarah Genon, Elvis Dohmatob, et al. 2020. "Individual Brain Charting Dataset Extension, Second Release of High-Resolution fMRI Data for Cognitive Mapping." *Scientific Data*

7 (1): 353.

Poldrack, Russell A. 2011. “Inferring Mental States from Neuroimaging Data: From Reverse Inference to Large-Scale Decoding.” *Neuron* 72 (5): 692–97.

Poldrack, Russell A., Jeanette A. Mumford, and Thomas E. Nichols. 2011. *Handbook of Functional MRI Data Analysis*. Cambridge University Press.

Porter, Alexis, Ashley Nielsen, Megan Dorn, Ally Dworetsky, Donnisa Edmonds, and Caterina Gratton. 2022. “Masked Features of Task States Found in Individual Brain Networks.” *Cerebral Cortex* , July. https://doi.org/10.1093/cercor/bhac247.

Projecto, Licenciatura em Engenharia Informática Relatório de. n.d. “Instituto Politécnico de Portalegre Escola Superior de Tecnologia E Gestão.” Accessed December 16, 2022. https://joaompfe.github.io/assets/bachelor-final-project.pdf.

Rastegarnia, Shima, Loic Tetrel, Basile Pinsard, Elizabeth DuPre, Yu Zhang, and Pierre Bellec. n.d. “Brain Decoding of the Human Connectome Project Tasks in a Dense Individual fMRI Dataset.” https://doi.org/10.31234/osf.io/9t5nh.

“Read Principles of fMRI.” n.d. Accessed October 13, 2022. https://leanpub.com/principlesoffmri/read.

Rué-Queralt, Joan, Angus Stevner, Enzo Tagliazucchi, Helmut Laufs, Morten L. Kringelbach, Gustavo Deco, and Selen Atasoy. 2021. “Decoding Brain States on the Intrinsic Manifold of Human Brain Dynamics across Wakefulness and Sleep.” *Communications Biology* 4 (1): 854.

Schaefer, Alexander, Ru Kong, Evan M. Gordon, Timothy O. Laumann, Xi-Nian Zuo, Avram J. Holmes, Simon B. Eickhoff, and B. T. Thomas Yeo. 2018. “Local-Global Parcellation of the Human Cerebral Cortex from Intrinsic Functional Connectivity MRI.” *Cerebral Cortex* 28 (9): 3095–3114.

Seeliger, K., L. Ambrogioni, Y. Güçlütürk, L. M. van den Bulk, U. Güçlü, and M. A. J. van Gerven. 2021. “End-to-End Neural System Identification with Neural Information Flow.” *PLoS Computational Biology* 17 (2): e1008558.

Seeliger, K., R. P. Sommers, U. Güçlü, S. E. Bosch, and M. A. J. van Gerven. 2019. “A Large Single-Participant fMRI Dataset for Probing Brain Responses to Naturalistic Stimuli in Space and Time.” *bioRxiv*. https://doi.org/10.1101/687681.

Setsompop, Kawin, Borjan A. Gagoski, Jonathan R. Polimeni, Thomas Witzel, Van J. Wedeen, and Lawrence L. Wald. 2012. “Blipped-Controlled Aliasing in Parallel Imaging for Simultaneous Multislice Echo Planar Imaging with Reduced G-Factor Penalty.” *Magnetic Resonance in Medicine: Official Journal of the Society of Magnetic Resonance in Medicine / Society of Magnetic Resonance in Medicine* 67 (5): 1210–24.

Shen, Junxiao, Boyin Yang, John J. Dudley, and Per Ola Kristensson. 2022. “KWickChat: A Multi-Turn Dialogue System for AAC Using Context-Aware Sentence Generation by

Bag-of-Keywords.” In *27th International Conference on Intelligent User Interfaces*, 853–67. IUI ’22. New York, NY, USA: Association for Computing Machinery.

Sterpenich, Virginie, Mojca K. M. van Schie, Maximilien Catsiyannis, Avinash Ramyead, Stephen Perrig, Hee-Deok Yang, Dimitri Van De Ville, and Sophie Schwartz. 2021. “Reward Biases Spontaneous Neural Reactivation during Sleep.” *Nature Communications* 12 (1): 4162.

Tang, Jerry, Amanda LeBel, Shailee Jain, and Alexander G. Huth. 2022. “Semantic Reconstruction of Continuous Language from Non-Invasive Brain Recordings.” *bioRxiv*. https://doi.org/10.1101/2022.09.29.509744.

Thomas, Armin W., Christopher Ré, and Russell A. Poldrack. 2022. “Self-Supervised Learning of Brain Dynamics from Broad Neuroimaging Data.” *arXiv [q-bio.NC]*. arXiv. http://arxiv.org/abs/2206.11417.

Urchs, Sebastian, Jonathan Armoza, Clara Moreau, Yassine Benhajali, Jolène St-Aubin, Pierre Orban, and Pierre Bellec. 2019. “MIST: A Multi-Resolution Parcellation of Functional Brain Networks.” *MNI Open Research* 1: 3.

Van Essen, David C., Stephen M. Smith, Deanna M. Barch, Timothy E. J. Behrens, Essa Yacoub, Kamil Ugurbil, and WU-Minn HCP Consortium. 2013. “The WU-Minn Human Connectome Project: An Overview.” *NeuroImage* 80 (October): 62–79.

Xu, Junqian, Steen Moeller, Edward J. Auerbach, John Strupp, Stephen M. Smith, David A. Feinberg, Essa Yacoub, and Kâmil Uğurbil. 2013. “Evaluation of Slice Accelerations Using Multiband Echo Planar Imaging at 3 T.” *NeuroImage* 83 (December): 991–1001.

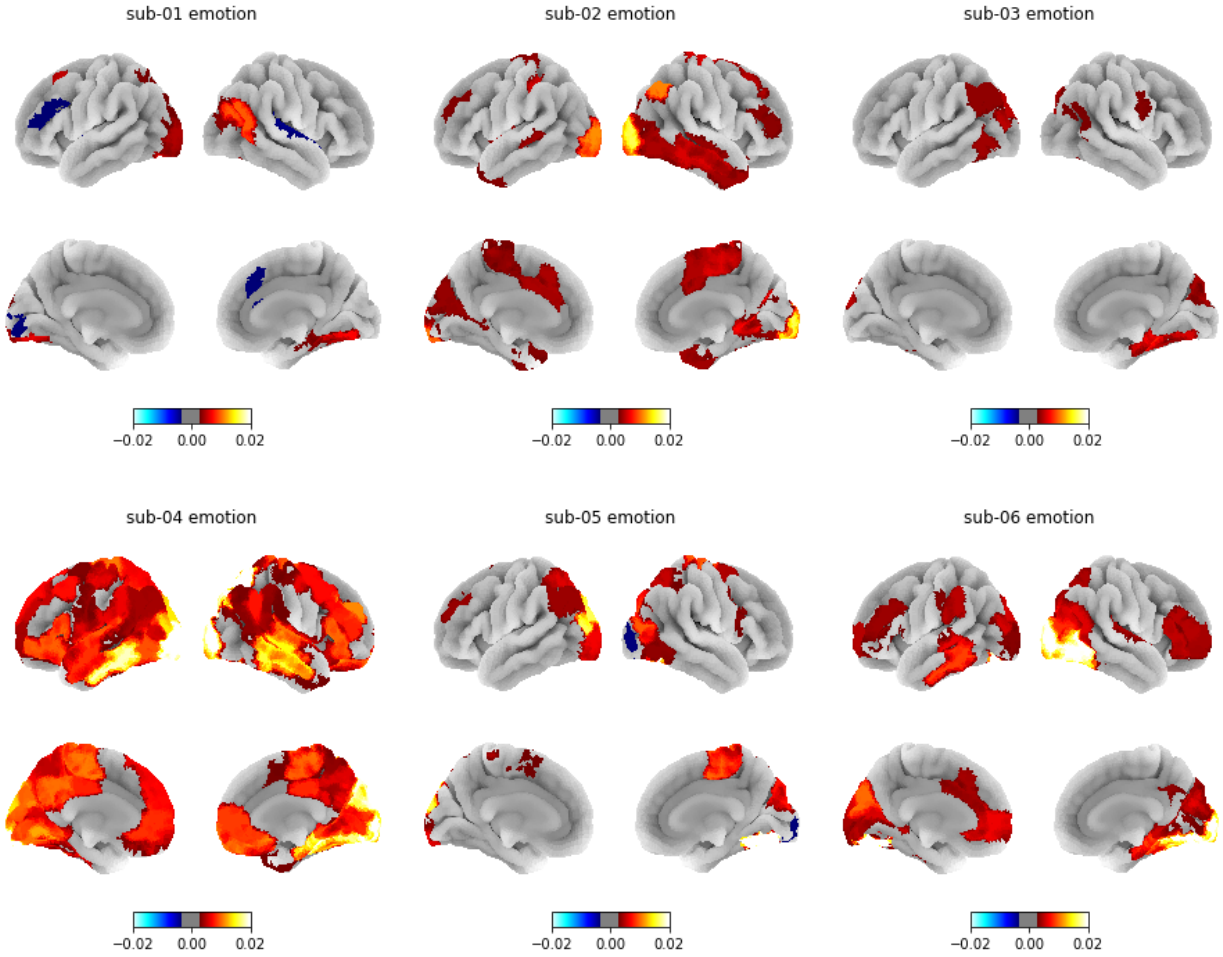
Zhang, Chao, Binru Dou, Jiali Wang, Kai Xu, Haiyan Zhang, Muhammad Umair Sami, Chunfeng Hu, et al. 2019. “Dynamic Alterations of Spontaneous Neural Activity in Parkinson’s Disease: A Resting-State fMRI Study.” *Frontiers in Neurology* 10 (October): 1052.

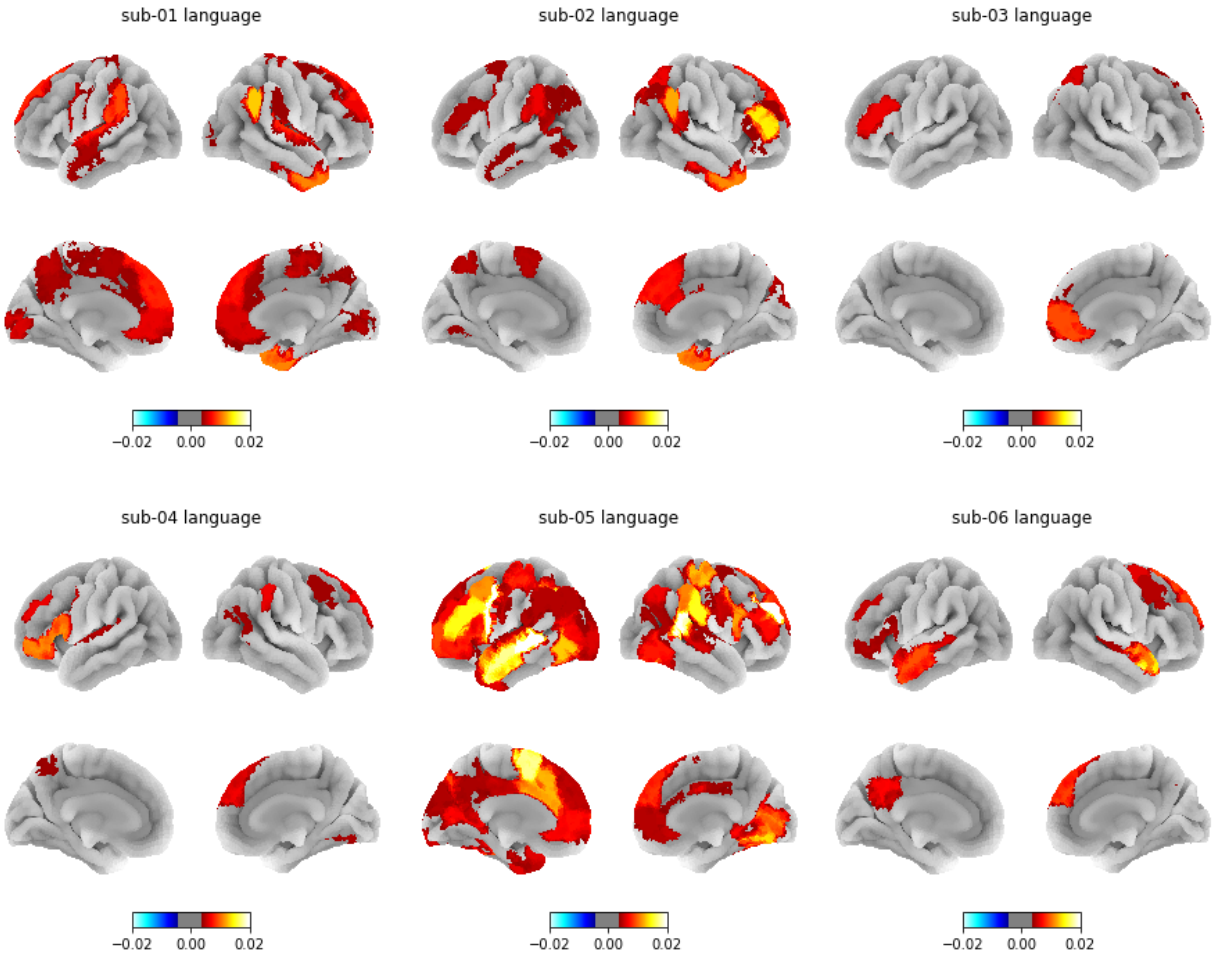
Zhang, Yu, Nicolas Farrugia, and Pierre Bellec. 2022. “Deep Learning Models of Cognitive Processes Constrained by Human Brain Connectomes.” *Medical Image Analysis* 80 (August): 102507.

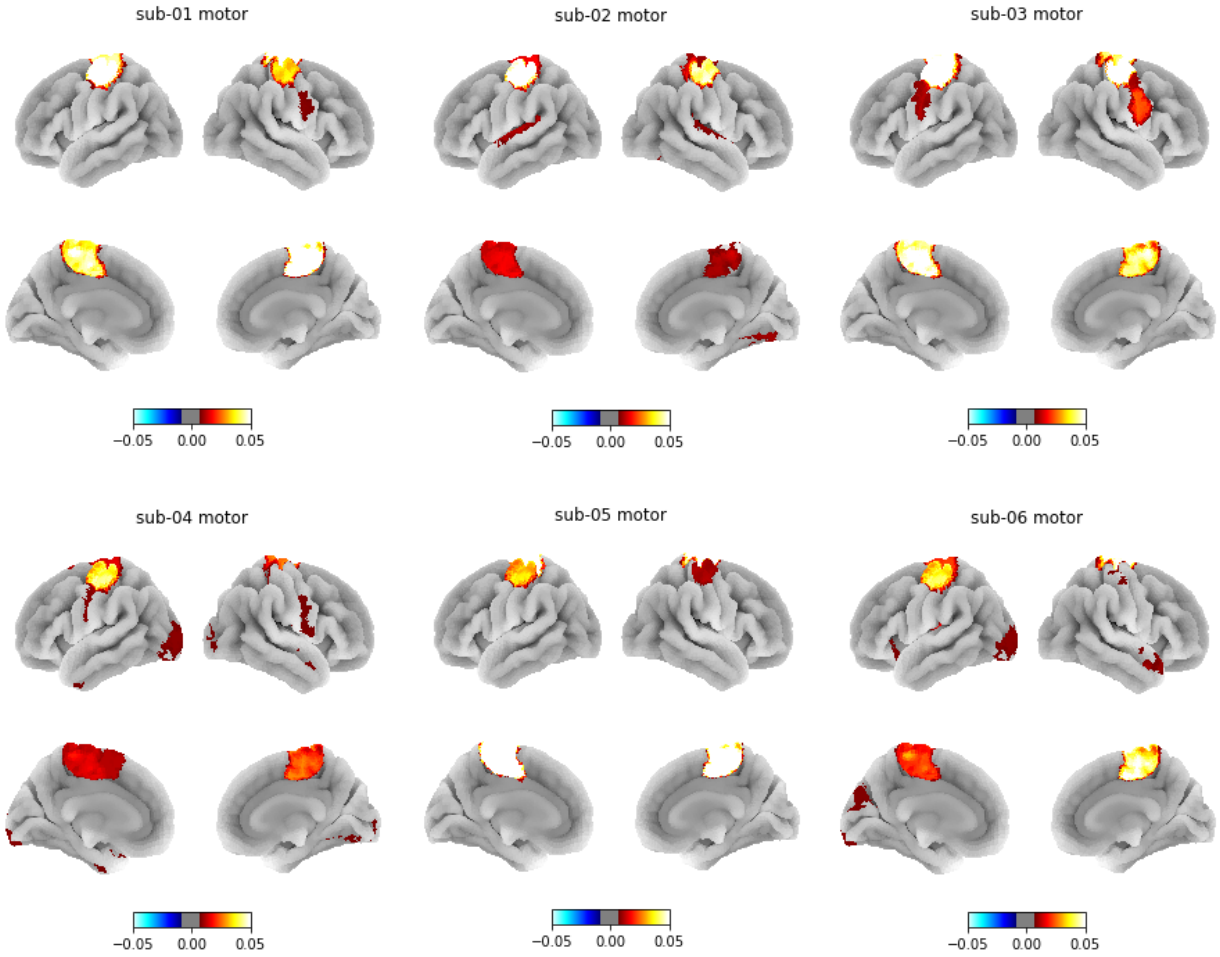
Zhang, Yu, Loïc Tetrel, Bertrand Thirion, and Pierre Bellec. 2021. “Functional Annotation of Human Cognitive States Using Deep Graph Convolution.” *NeuroImage* 231 (May): 117847.

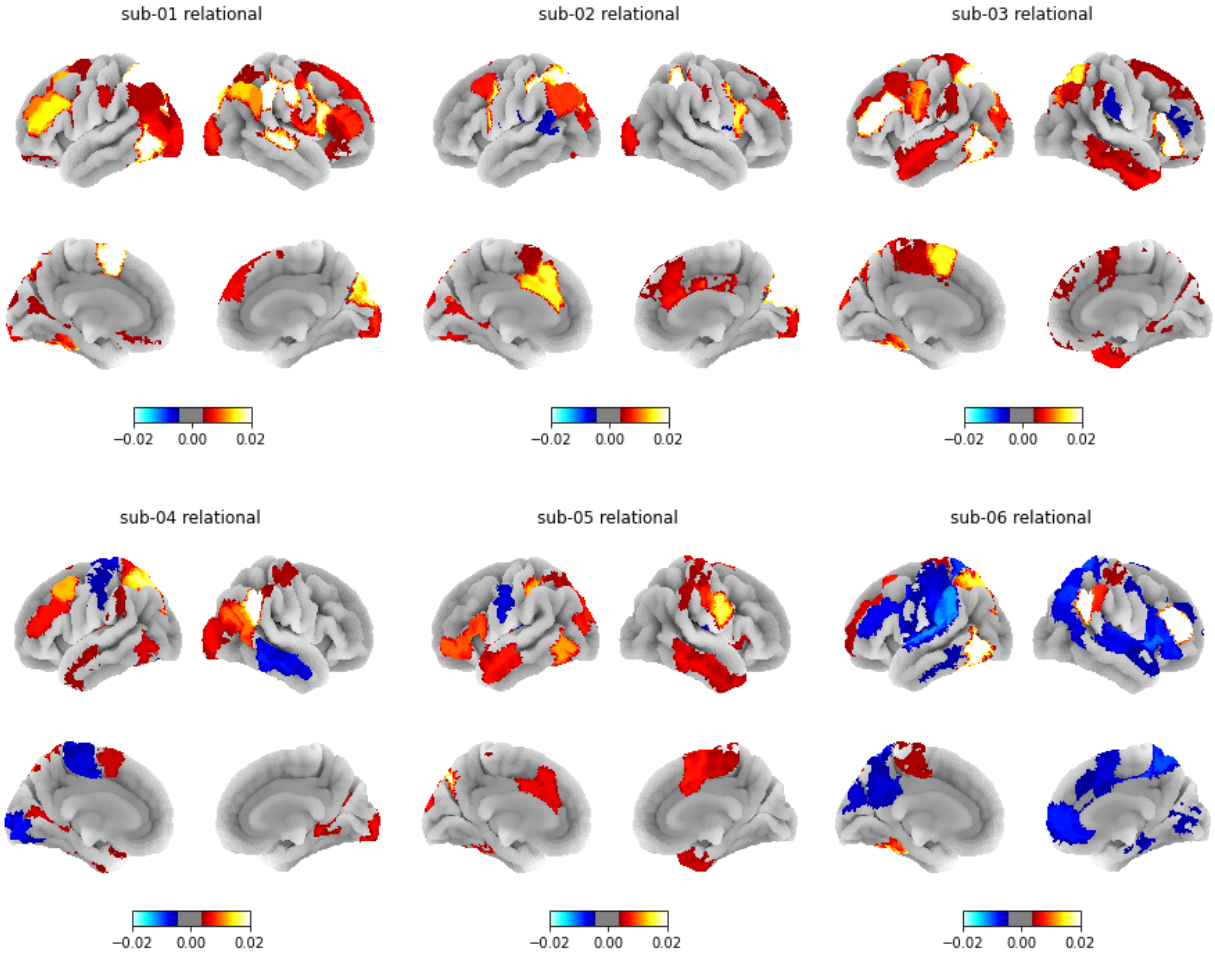
Supplementary material

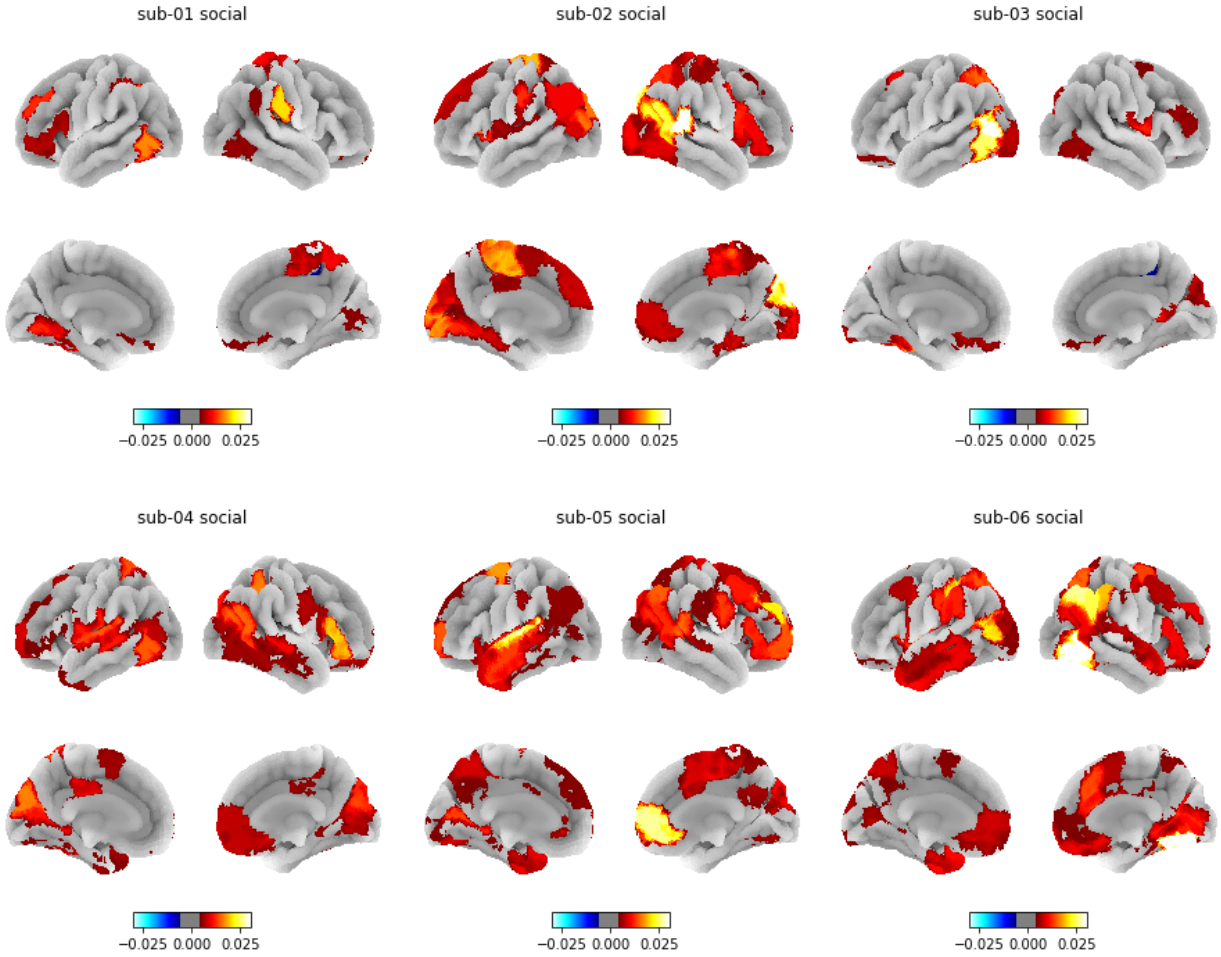
Permutation importance maps:

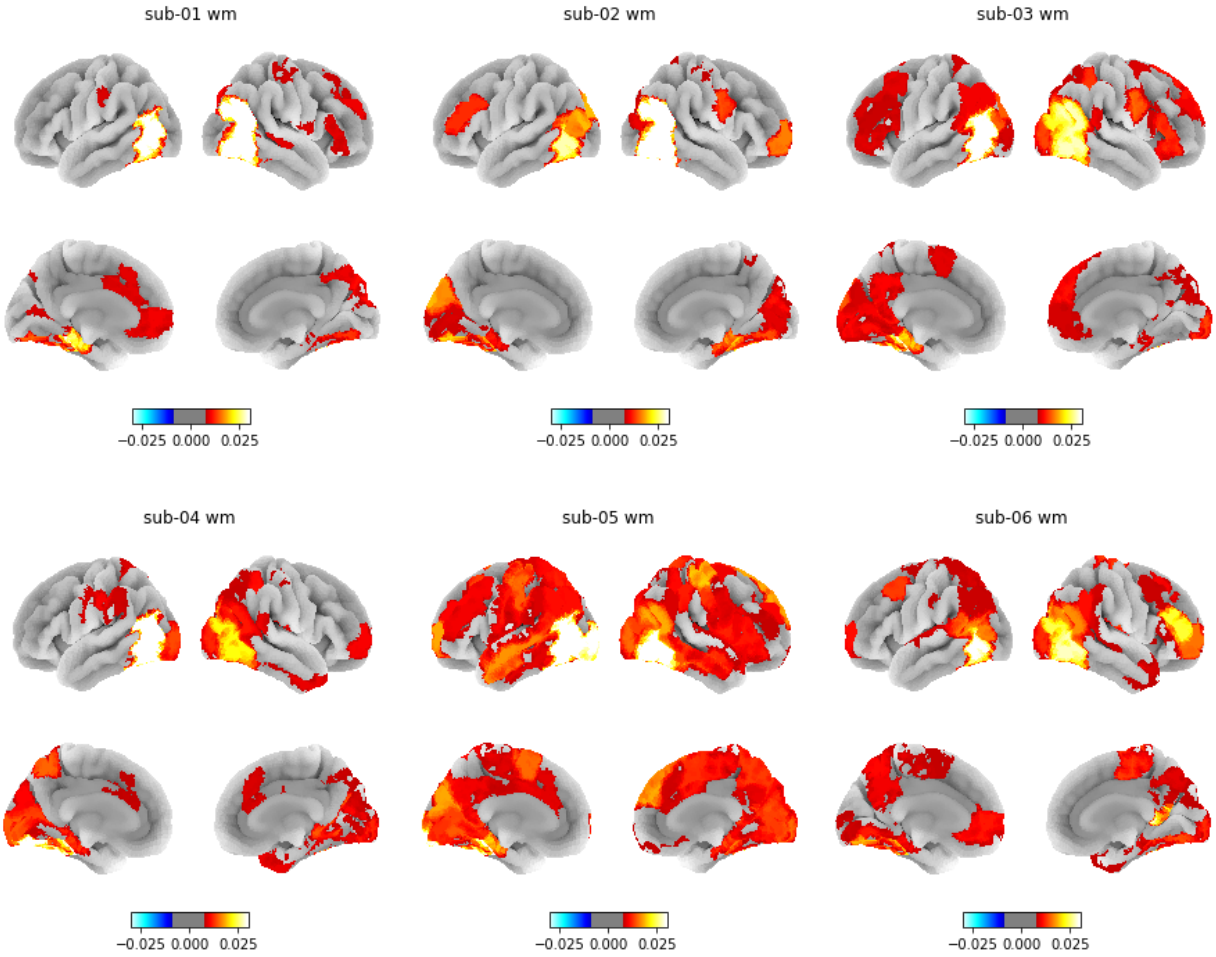



sub-01 language
sub-02 language
sub-03 language
sub-04 language
sub-05 language
sub-06 language
−0.02
0.00
0.02

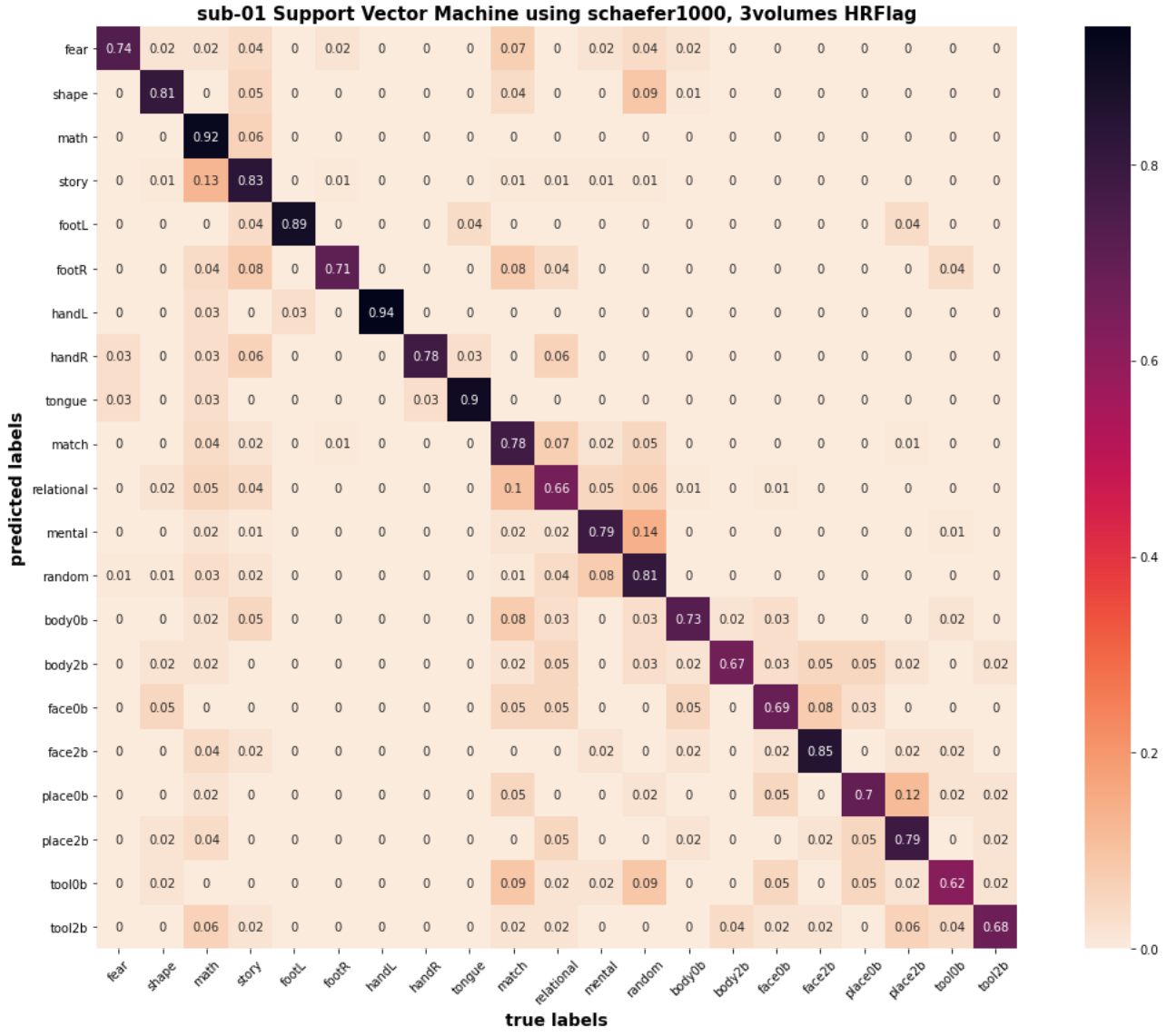

sub-01 motor
sub-02 motor
sub-03 motor
sub-04 motor
sub-05 motor
sub-06 motor
−0.05 0.00 0.05

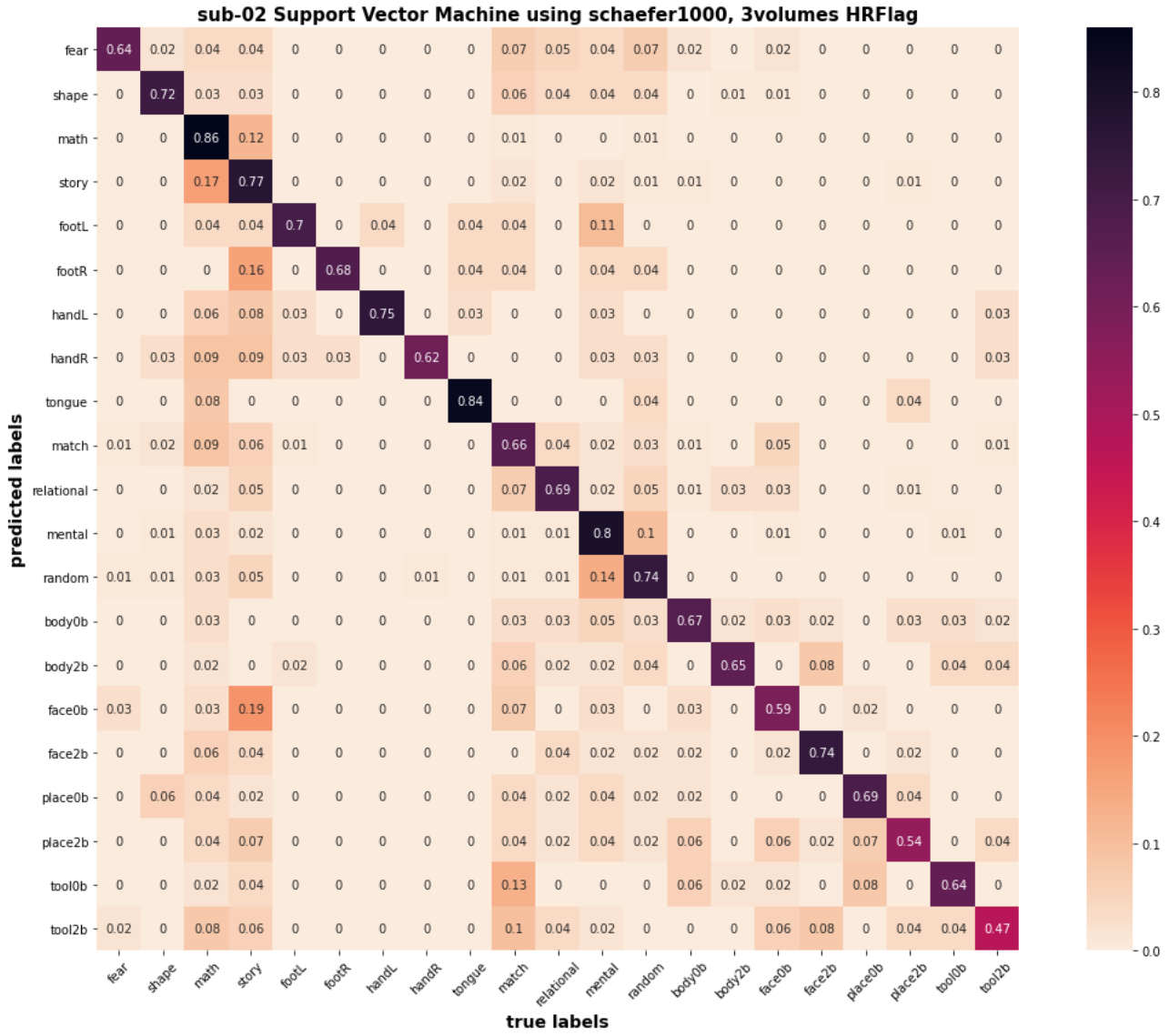

sub-01 relational
sub-02 relational
sub-03 relational
sub-04 relational
sub-05 relational
sub-06 relational
−0.02
0.00
0.02

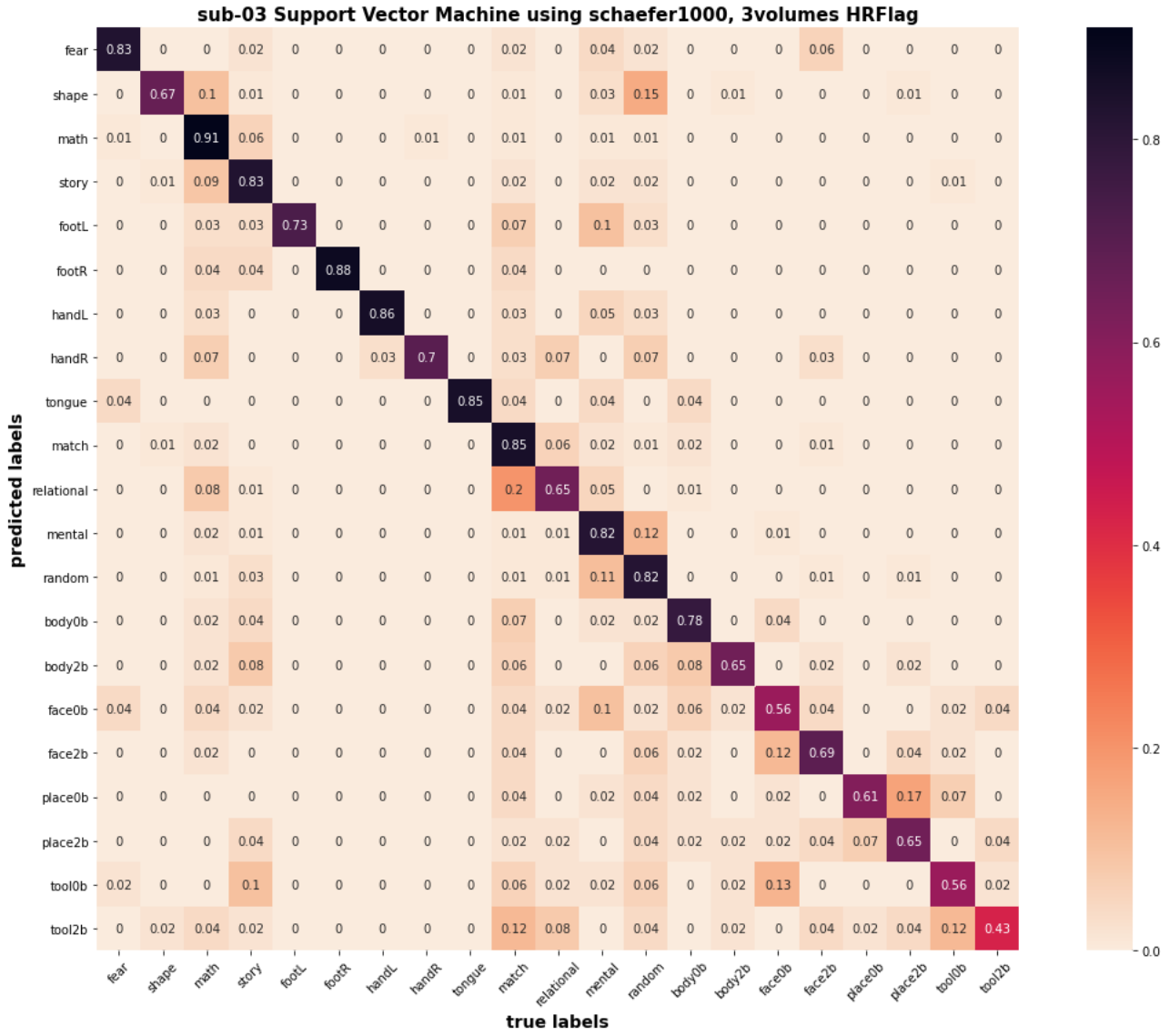

sub-01 social
sub-02 social
sub-03 social
sub-04 social
sub-05 social
sub-06 social
−0.025 0.000 0.025

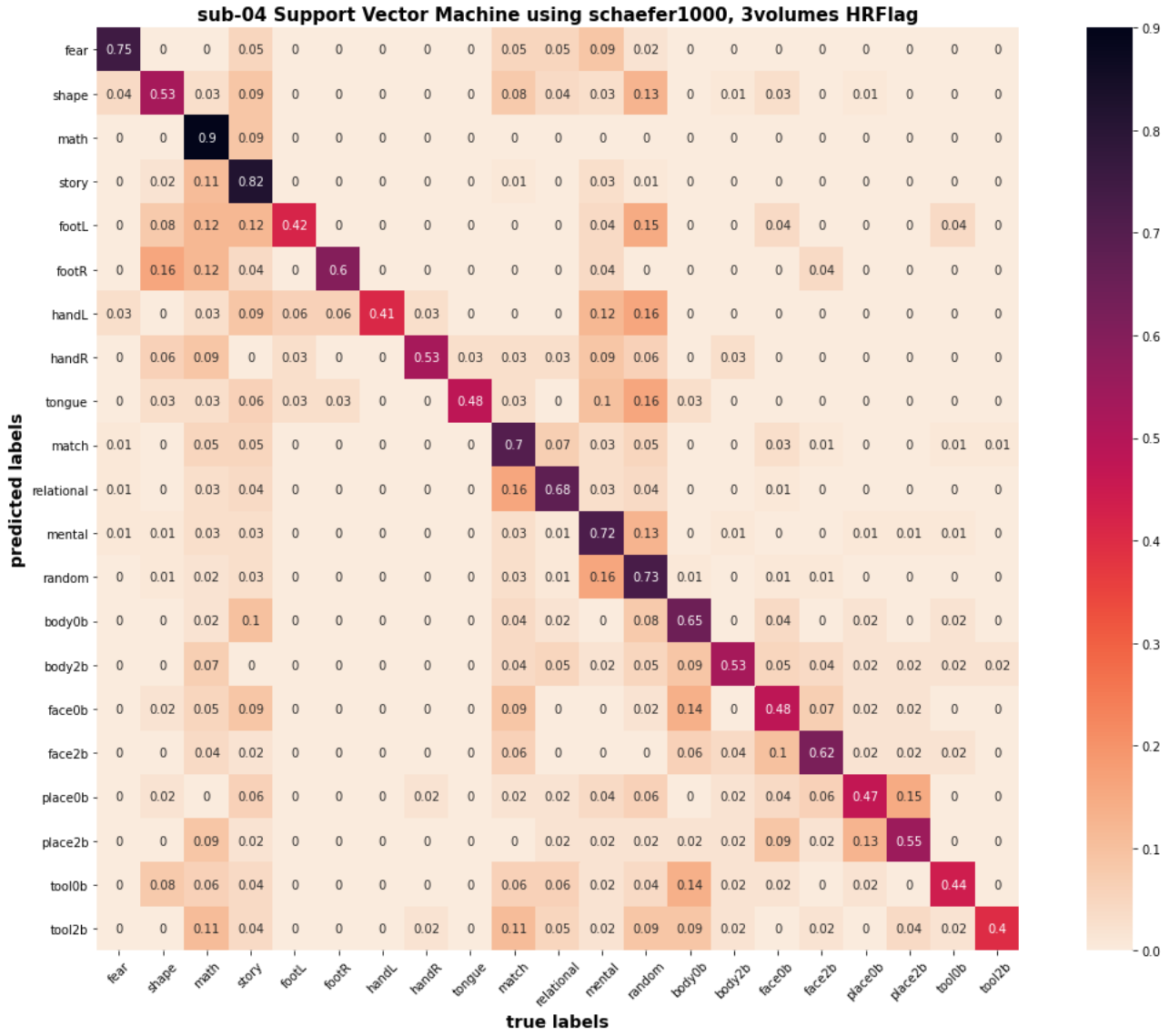

sub-01 wm
sub-02 wm
sub-03 wm
sub-04 wm
sub-05 wm
sub-06 wm
−0.025 0.000 0.025

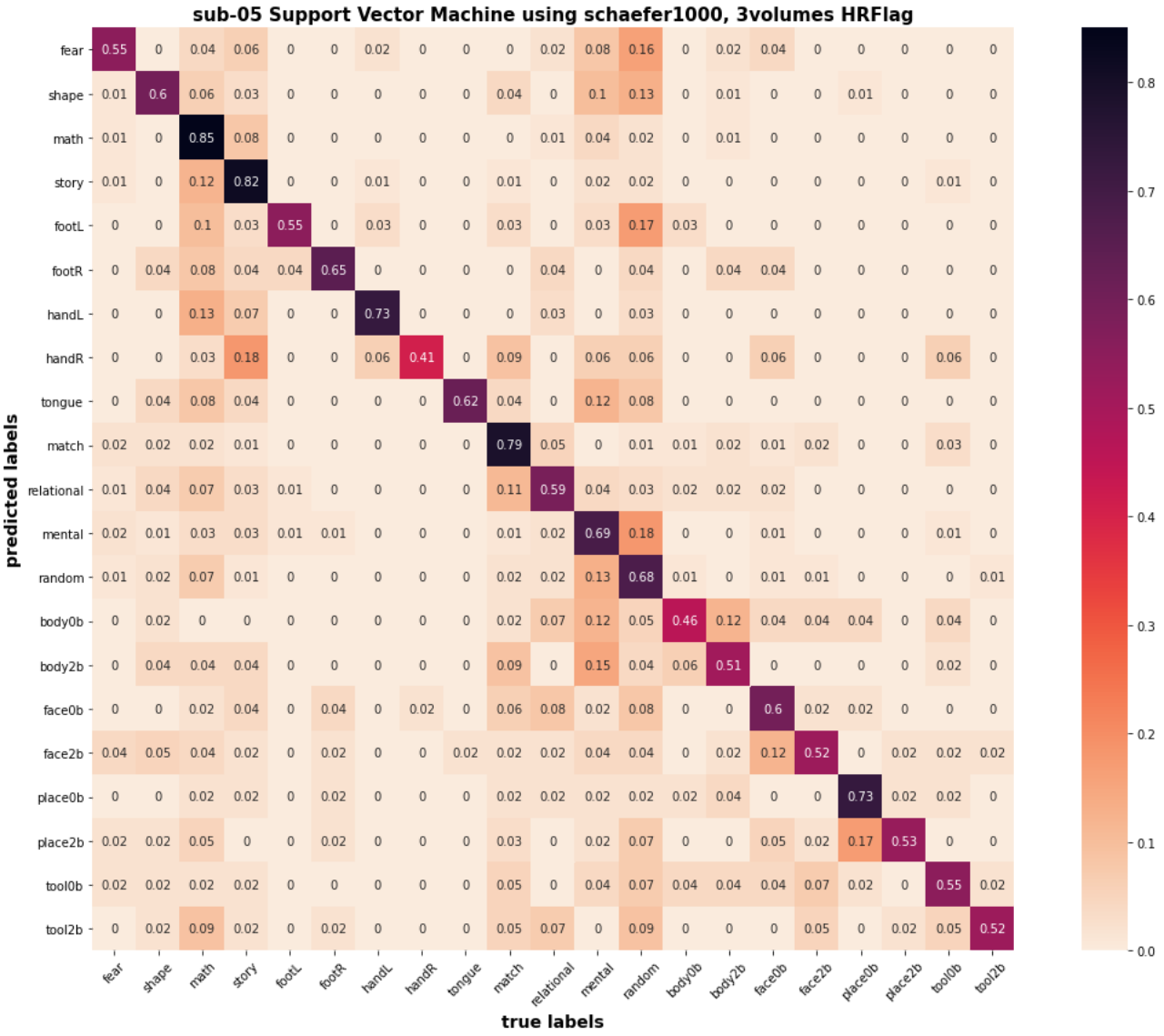
The best decoding model confusion matrices:

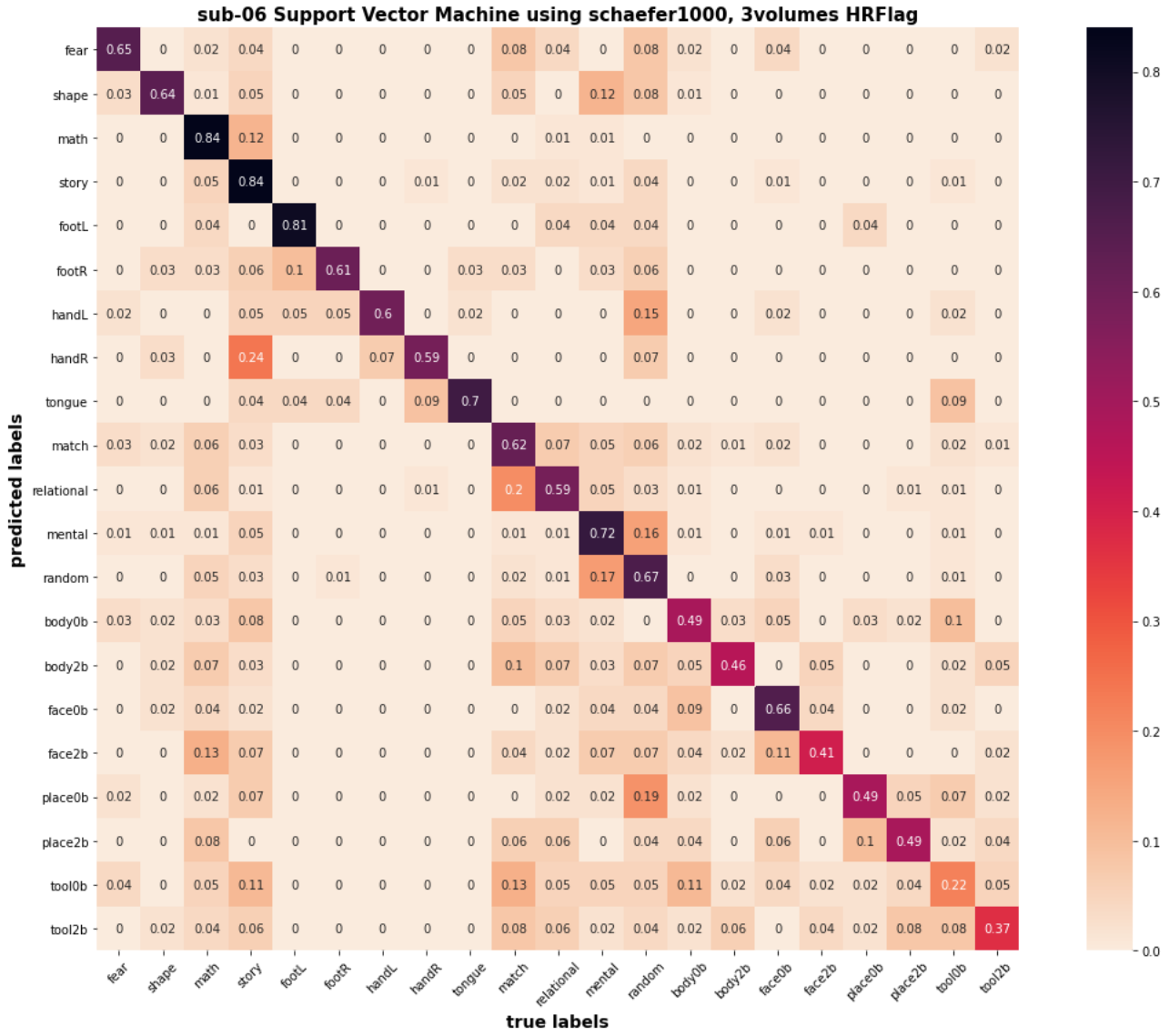



sub-02 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear
shape
math
story
footL
footR
handL
handR
tongue
match
relational
mental
random
body0b
body2b
face0b
face2b
place0b
place2b
tool0b
tool2b
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8

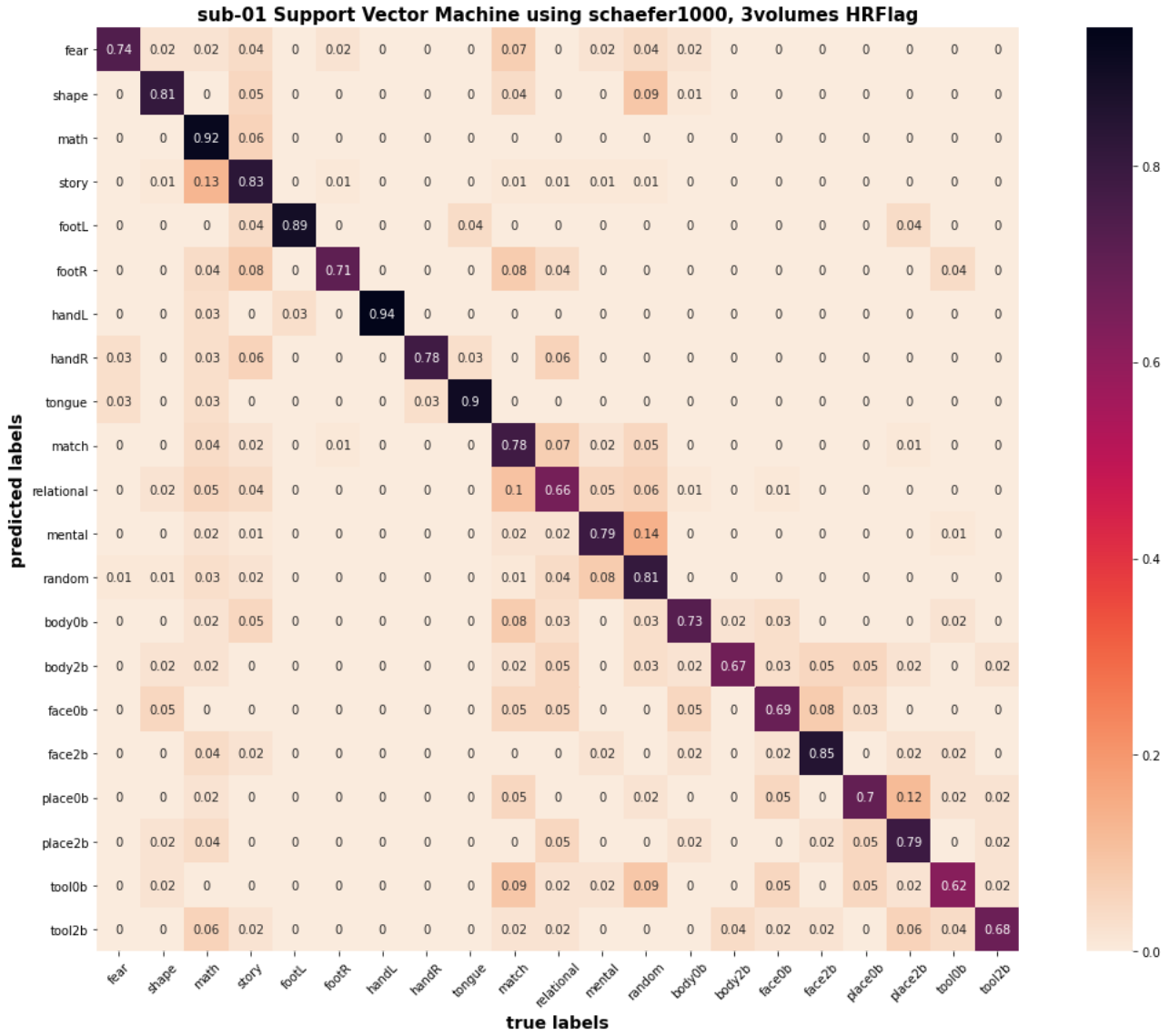

sub-03 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear
shape
math
story
footL
footR
handL
handR
tongue
match
relational
mental
random
body0b
body2b
face0b
face2b
place0b
place2b
tool0b
tool2b
0.0
0.2
0.4
0.6
0.8

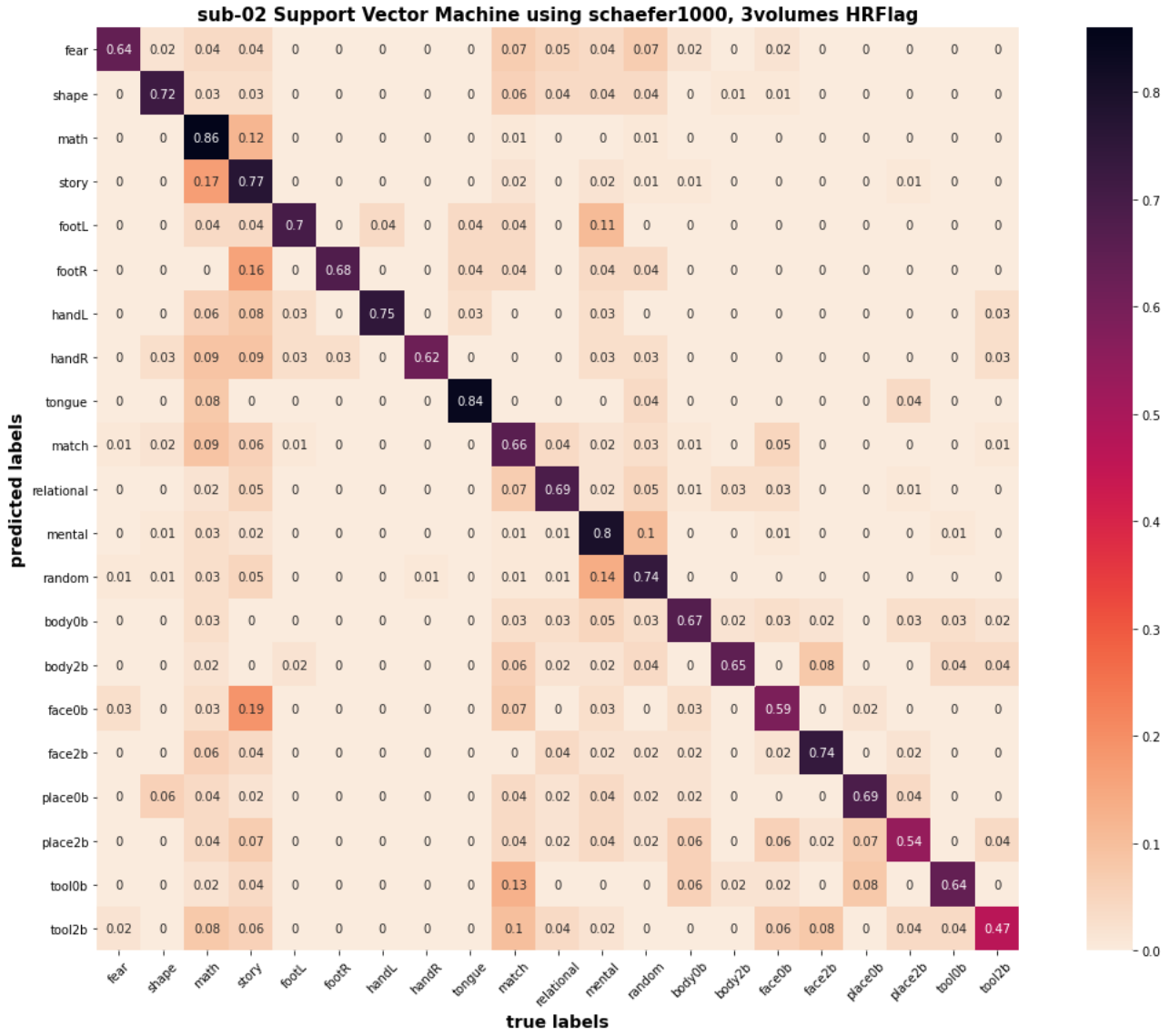

sub-04 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear 0.75 0 0 0.05 0 0 0 0 0 0.05 0.05 0.09 0.02 0 0 0 0 0 0 0 0
shape 0.04 0.53 0.03 0.09 0 0 0 0 0 0.08 0.04 0.03 0.13 0 0.01 0.03 0 0.01 0 0 0
math 0 0 0.9 0.09 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
story 0 0.02 0.11 0.82 0 0 0 0 0 0.01 0 0.03 0.01 0 0 0 0 0 0 0 0
footL 0 0.08 0.12 0.12 0.42 0 0 0 0 0 0 0.04 0.15 0 0 0.04 0 0 0 0.04 0
footR 0 0.16 0.12 0.04 0 0.6 0 0 0 0 0 0.04 0 0 0 0 0.04 0 0 0 0
handL 0.03 0 0.03 0.09 0.06 0.06 0.41 0.03 0 0 0 0.12 0.16 0 0 0 0 0 0 0 0
handR 0 0.06 0.09 0 0.03 0 0 0.53 0.03 0.03 0.03 0.09 0.06 0 0.03 0 0 0 0 0 0
tongue 0 0.03 0.03 0.06 0.03 0.03 0 0 0.48 0.03 0 0.1 0.16 0.03 0 0 0 0 0 0 0
match 0.01 0 0.05 0.05 0 0 0 0 0 0.7 0.07 0.03 0.05 0 0 0.03 0.01 0 0 0.01 0.01
relational 0.01 0 0.03 0.04 0 0 0 0 0 0.16 0.68 0.03 0.04 0 0 0.01 0 0 0 0 0
mental 0.01 0.01 0.03 0.03 0 0 0 0 0 0.03 0.01 0.72 0.13 0 0.01 0 0 0.01 0.01 0.01 0
random 0 0.01 0.02 0.03 0 0 0 0 0 0.03 0.01 0.16 0.73 0.01 0 0.01 0.01 0 0 0 0
body0b 0 0 0.02 0.1 0 0 0 0 0 0.04 0.02 0 0.08 0.65 0 0.04 0 0.02 0 0.02 0
body2b 0 0 0.07 0 0 0 0 0 0 0.04 0.05 0.02 0.05 0.09 0.53 0.05 0.04 0.02 0.02 0.02 0.02
face0b 0 0.02 0.05 0.09 0 0 0 0 0 0.09 0 0 0.02 0.14 0 0.48 0.07 0.02 0.02 0 0
face2b 0 0 0.04 0.02 0 0 0 0 0 0.06 0 0 0 0.06 0.04 0.1 0.62 0.02 0.02 0.02 0
place0b 0 0.02 0 0.06 0 0 0 0.02 0 0.02 0.02 0.04 0.06 0 0.02 0.04 0.06 0.47 0.15 0 0
place2b 0 0 0.09 0.02 0 0 0 0 0 0 0.02 0.02 0.02 0.02 0.02 0.09 0.02 0.13 0.55 0 0
tool0b 0 0.08 0.06 0.04 0 0 0 0 0 0.06 0.06 0.02 0.04 0.14 0.02 0.02 0 0.02 0 0.44 0
tool2b 0 0 0.11 0.04 0 0 0 0.02 0 0.11 0.05 0.02 0.09 0.09 0.02 0 0.02 0 0.04 0.02 0.4
fear shape math story footL footR handL handR tongue match relational mental random body0b body2b face0b face2b place0b place2b tool0b tool2b
0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0

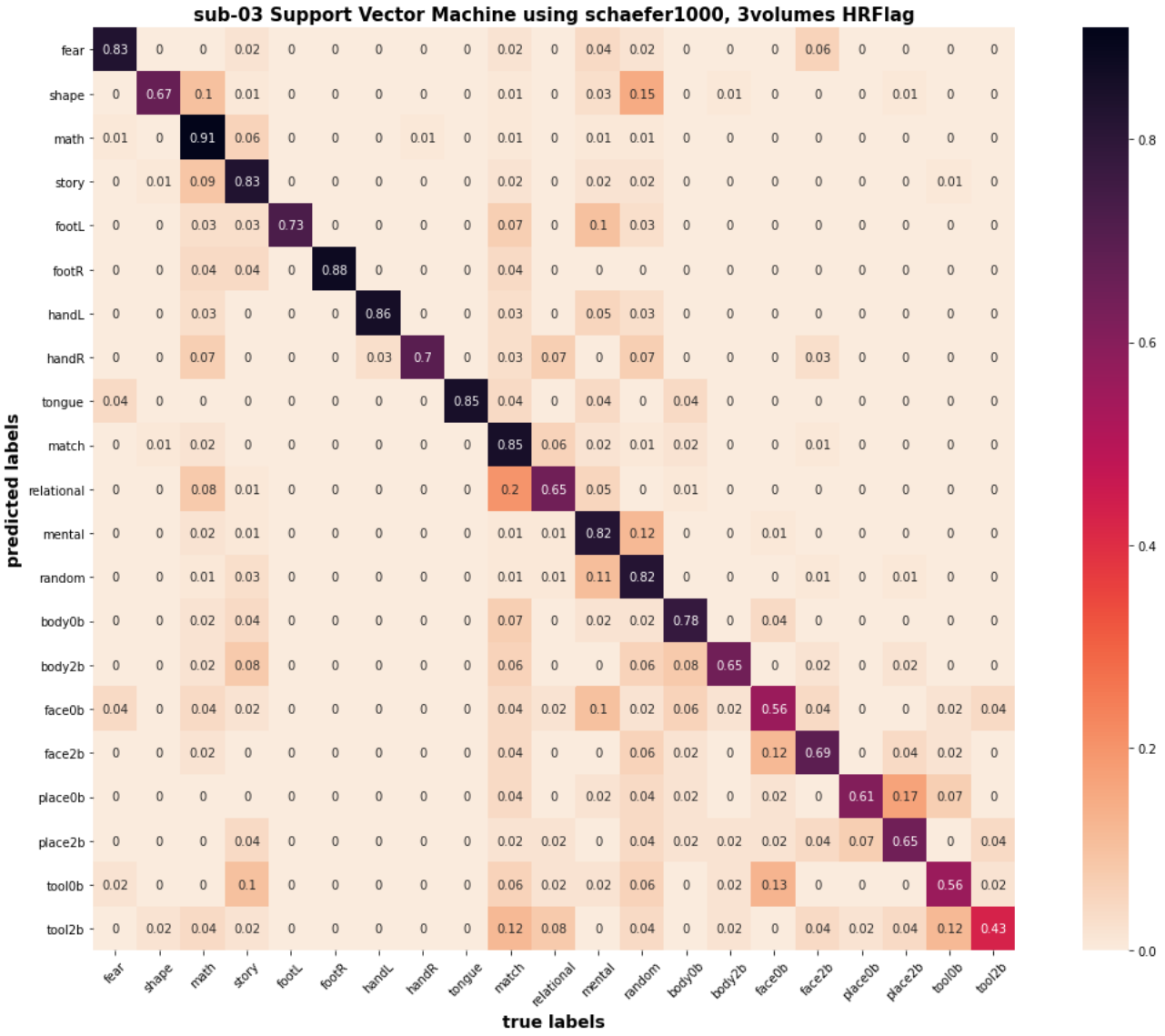

sub-05 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear 0.55 0 0.04 0.06 0 0 0.02 0 0 0 0.02 0.08 0.16 0 0.02 0.04 0 0 0 0 0
shape 0.01 0.6 0.06 0.03 0 0 0 0 0 0.04 0 0.1 0.13 0 0.01 0 0 0.01 0 0 0
math 0.01 0 0.85 0.08 0 0 0 0 0 0 0.01 0.04 0.02 0 0.01 0 0 0 0 0 0
story 0.01 0 0.12 0.82 0 0 0.01 0 0 0.01 0 0.02 0.02 0 0 0 0 0 0 0.01 0
footL 0 0 0.1 0.03 0.55 0 0.03 0 0 0.03 0 0.03 0.17 0.03 0 0 0 0 0 0 0
footR 0 0.04 0.08 0.04 0.04 0.65 0 0 0 0 0.04 0 0.04 0 0.04 0.04 0 0 0 0 0
handL 0 0 0.13 0.07 0 0 0.73 0 0 0 0.03 0 0.03 0 0 0 0 0 0 0 0
handR 0 0 0.03 0.18 0 0 0.06 0.41 0 0.09 0 0.06 0.06 0 0 0.06 0 0 0 0.06 0
tongue 0 0.04 0.08 0.04 0 0 0 0 0.62 0.04 0 0.12 0.08 0 0 0 0 0 0 0 0
match 0.02 0.02 0.02 0.01 0 0 0 0 0 0.79 0.05 0 0.01 0.01 0.02 0.01 0.02 0 0 0.03 0
relational 0.01 0.04 0.07 0.03 0.01 0 0 0 0 0.11 0.59 0.04 0.03 0.02 0.02 0.02 0 0 0 0 0
mental 0.02 0.01 0.03 0.03 0.01 0.01 0 0 0 0.01 0.02 0.69 0.18 0 0 0.01 0 0 0 0.01 0
random 0.01 0.02 0.07 0.01 0 0 0 0 0 0.02 0.02 0.13 0.68 0.01 0 0.01 0.01 0 0 0 0.01
body0b 0 0.02 0 0 0 0 0 0 0 0.02 0.07 0.12 0.05 0.46 0.12 0.04 0.04 0.04 0 0.04 0
body2b 0 0.04 0.04 0.04 0 0 0 0 0 0.09 0 0.15 0.04 0.06 0.51 0 0 0 0 0.02 0
face0b 0 0 0.02 0.04 0 0.04 0 0.02 0 0.06 0.08 0.02 0.08 0 0 0.6 0.02 0.02 0 0 0
face2b 0.04 0.05 0.04 0.02 0 0.02 0 0 0.02 0.02 0.02 0.04 0.04 0 0.02 0.12 0.52 0 0.02 0.02 0.02
place0b 0 0 0.02 0.02 0 0.02 0 0 0 0.02 0.02 0.02 0.02 0.02 0.04 0 0 0.73 0.02 0.02 0
place2b 0.02 0.02 0.05 0 0 0.02 0 0 0 0.03 0 0.02 0.07 0 0 0.05 0.02 0.17 0.53 0 0
tool0b 0.02 0.02 0.02 0.02 0 0 0 0 0 0.05 0 0.04 0.07 0.04 0.04 0.04 0.07 0.02 0 0.55 0.02
tool2b 0 0.02 0.09 0.02 0 0.02 0 0 0 0.05 0.07 0 0.09 0 0 0 0.05 0 0.02 0.05 0.52
fear shape math story footL footR handL handR tongue match relational mental random body0b body2b face0b face2b place0b place2b tool0b tool2b
0.8
0.7
0.6
0.5
0.4
0.3
0.2
0.1
0.0

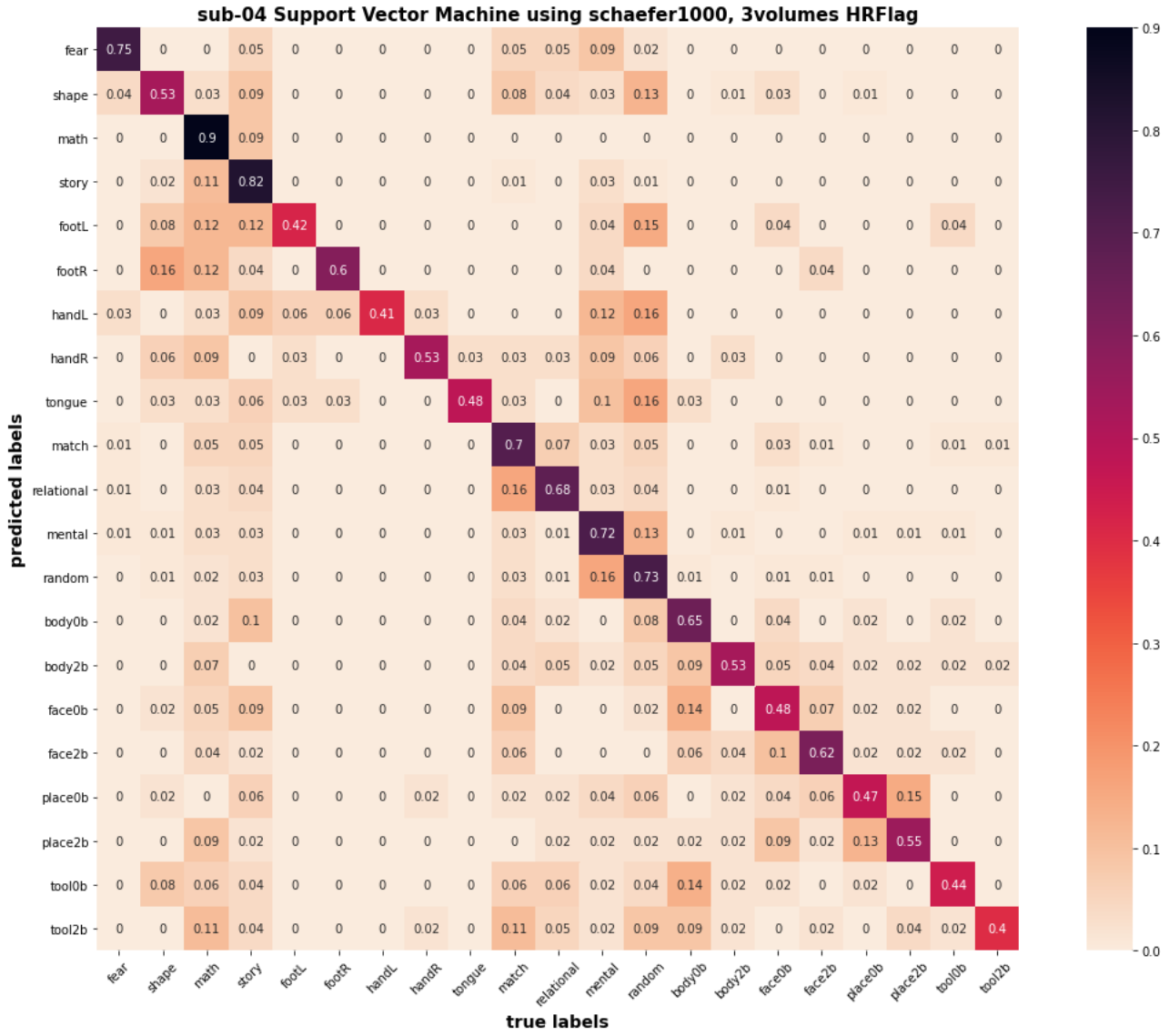

sub-06 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear
shape
math
story
footL
footR
handL
handR
tongue
match
relational
mental
random
body0b
body2b
face0b
face2b
place0b
place2b
tool0b
tool2b
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8

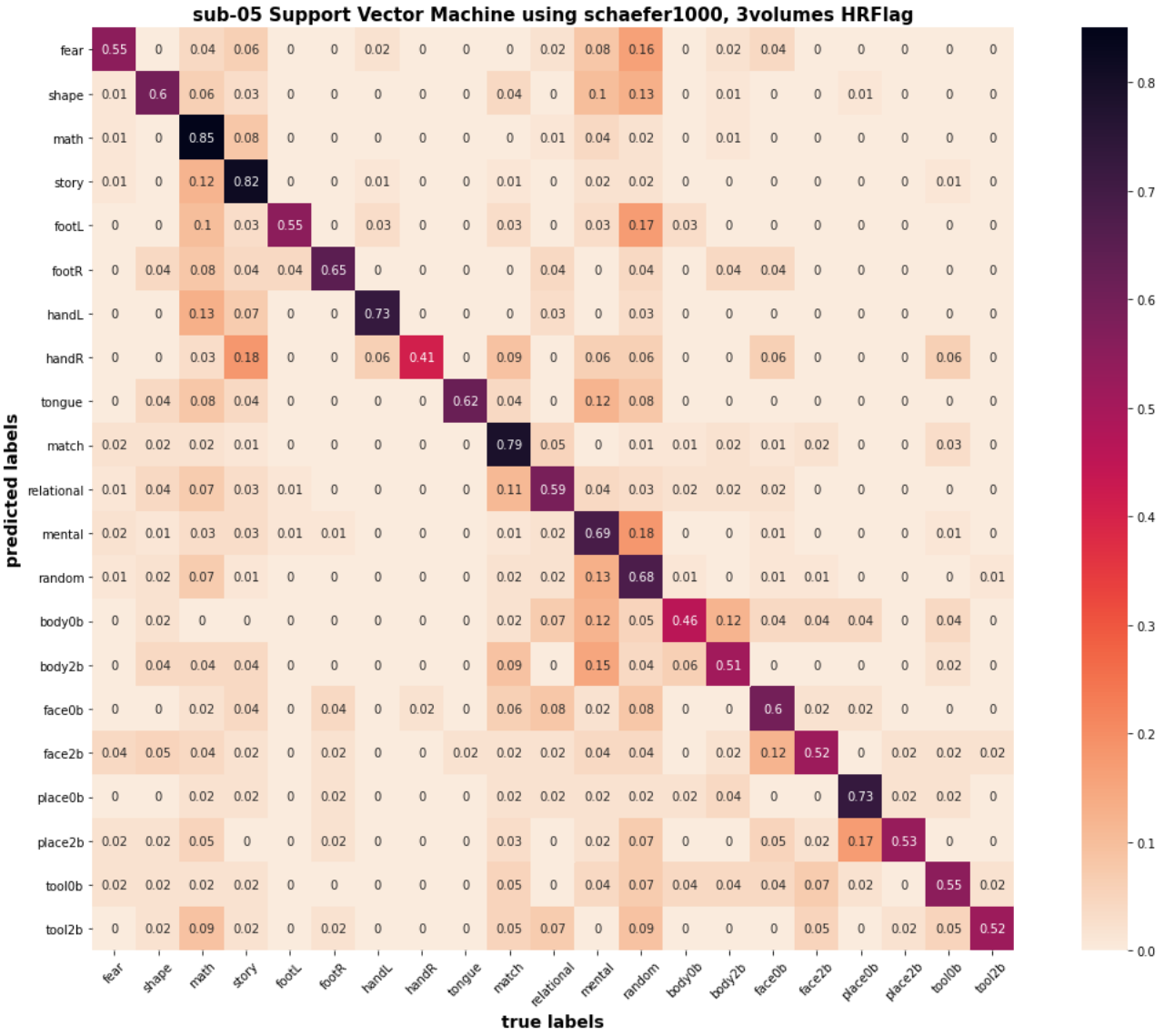
The SVM model with session-based split train and test set confusion matrices:

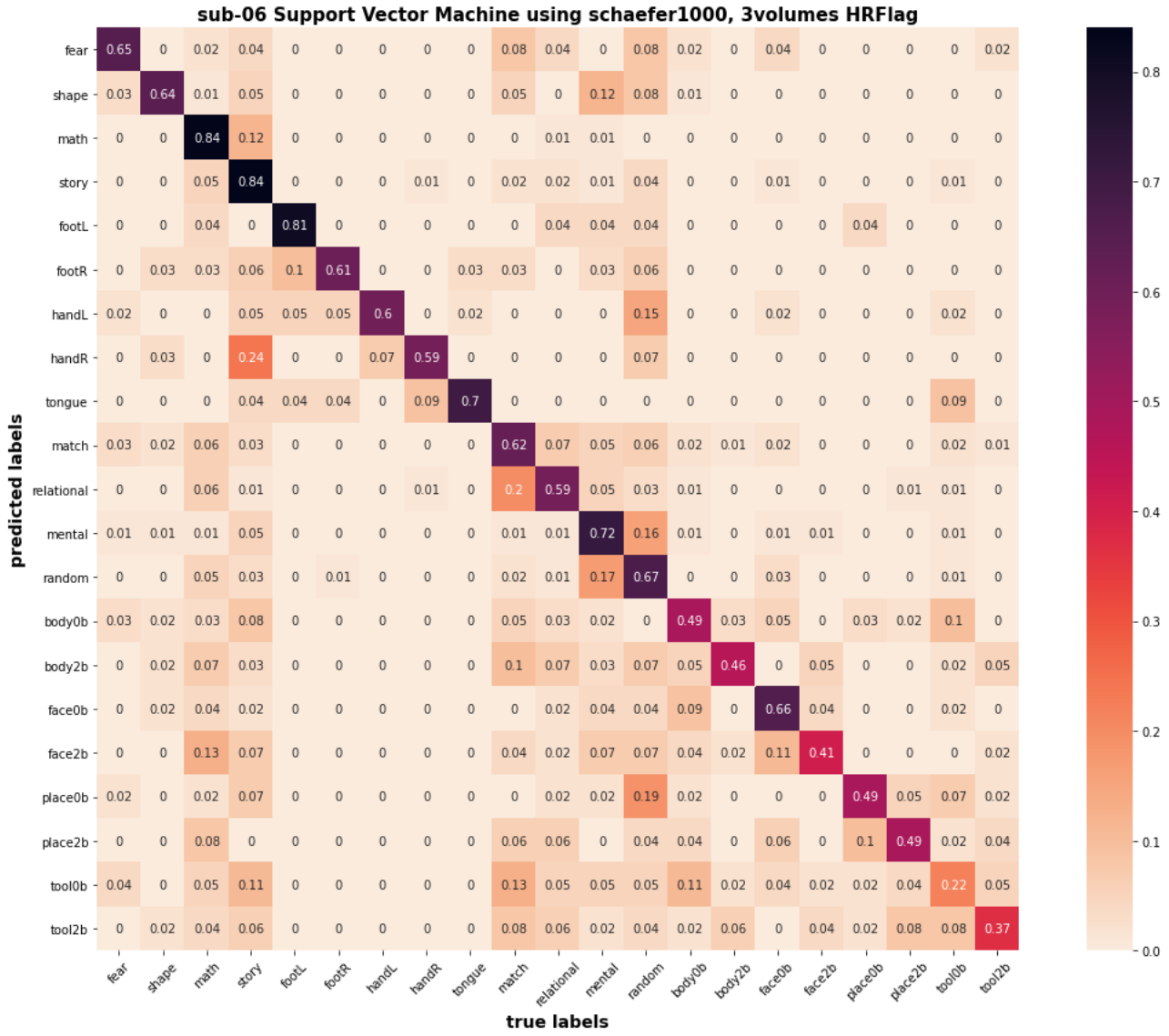



sub-02 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear 0.64 0.02 0.04 0.04 0 0 0 0 0 0.07 0.05 0.04 0.07 0.02 0 0.02 0 0 0 0 0
shape 0 0.72 0.03 0.03 0 0 0 0 0 0.06 0.04 0.04 0.04 0 0.01 0.01 0 0 0 0 0
math 0 0 0.86 0.12 0 0 0 0 0 0.01 0 0 0.01 0 0 0 0 0 0 0 0
story 0 0 0.17 0.77 0 0 0 0 0 0.02 0 0.02 0.01 0.01 0 0 0 0 0.01 0 0
footL 0 0 0.04 0.04 0.7 0 0.04 0 0.04 0.04 0 0.11 0 0 0 0 0 0 0 0 0
footR 0 0 0 0.16 0 0.68 0 0 0.04 0.04 0 0.04 0.04 0 0 0 0 0 0 0 0
handL 0 0 0.06 0.08 0.03 0 0.75 0 0.03 0 0 0.03 0 0 0 0 0 0 0 0 0.03
handR 0 0.03 0.09 0.09 0.03 0.03 0 0.62 0 0 0 0.03 0.03 0 0 0 0 0 0 0 0.03
tongue 0 0 0.08 0 0 0 0 0 0.84 0 0 0 0.04 0 0 0 0 0 0.04 0 0
match 0.01 0.02 0.09 0.06 0.01 0 0 0 0 0.66 0.04 0.02 0.03 0.01 0 0.05 0 0 0 0 0.01
relational 0 0 0.02 0.05 0 0 0 0 0 0.07 0.69 0.02 0.05 0.01 0.03 0.03 0 0 0.01 0 0
mental 0 0.01 0.03 0.02 0 0 0 0 0 0.01 0.01 0.8 0.1 0 0 0.01 0 0 0 0.01 0
random 0.01 0.01 0.03 0.05 0 0 0 0.01 0 0.01 0.01 0.14 0.74 0 0 0 0 0 0 0 0
body0b 0 0 0.03 0 0 0 0 0 0 0.03 0.03 0.05 0.03 0.67 0.02 0.03 0.02 0 0.03 0.03 0.02
body2b 0 0 0.02 0 0.02 0 0 0 0 0.06 0.02 0.02 0.04 0 0.65 0 0.08 0 0 0.04 0.04
face0b 0.03 0 0.03 0.19 0 0 0 0 0 0.07 0 0.03 0 0.03 0 0.59 0 0.02 0 0 0
face2b 0 0 0.06 0.04 0 0 0 0 0 0 0.04 0.02 0.02 0.02 0 0.02 0.74 0 0.02 0 0
place0b 0 0.06 0.04 0.02 0 0 0 0 0 0.04 0.02 0.04 0.02 0.02 0 0 0 0.69 0.04 0 0
place2b 0 0 0.04 0.07 0 0 0 0 0 0.04 0.02 0.04 0.02 0.06 0 0.06 0.02 0.07 0.54 0 0.04
tool0b 0 0 0.02 0.04 0 0 0 0 0 0.13 0 0 0 0.06 0.02 0.02 0 0.08 0 0.64 0
tool2b 0.02 0 0.08 0.06 0 0 0 0 0 0.1 0.04 0.02 0 0 0 0.06 0.08 0 0.04 0.04 0.47
fear shape math story footL footR handL handR tongue match relational mental random body0b body2b face0b face2b place0b place2b tool0b tool2b
0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0


sub-03 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear
shape
math
story
footL
footR
handL
handR
tongue
match
relational
mental
random
body0b
body2b
face0b
face2b
place0b
place2b
tool0b
tool2b
0.0
0.2
0.4
0.6
0.8


sub-04 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear 0.75 0 0 0.05 0 0 0 0 0 0.05 0.05 0.09 0.02 0 0 0 0 0 0 0 0
shape 0.04 0.53 0.03 0.09 0 0 0 0 0 0.08 0.04 0.03 0.13 0 0.01 0.03 0 0.01 0 0 0
math 0 0 0.9 0.09 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0 0
story 0 0.02 0.11 0.82 0 0 0 0 0 0.01 0 0.03 0.01 0 0 0 0 0 0 0 0
footL 0 0.08 0.12 0.12 0.42 0 0 0 0 0 0 0.04 0.15 0 0 0.04 0 0 0 0.04 0
footR 0 0.16 0.12 0.04 0 0.6 0 0 0 0 0 0.04 0 0 0 0 0.04 0 0 0 0
handL 0.03 0 0.03 0.09 0.06 0.06 0.41 0.03 0 0 0 0.12 0.16 0 0 0 0 0 0 0 0
handR 0 0.06 0.09 0 0.03 0 0 0.53 0.03 0.03 0.03 0.09 0.06 0 0.03 0 0 0 0 0 0
tongue 0 0.03 0.03 0.06 0.03 0.03 0 0 0.48 0.03 0 0.1 0.16 0.03 0 0 0 0 0 0 0
match 0.01 0 0.05 0.05 0 0 0 0 0 0.7 0.07 0.03 0.05 0 0 0.03 0.01 0 0 0.01 0.01
relational 0.01 0 0.03 0.04 0 0 0 0 0 0.16 0.68 0.03 0.04 0 0 0.01 0 0 0 0 0
mental 0.01 0.01 0.03 0.03 0 0 0 0 0 0.03 0.01 0.72 0.13 0 0.01 0 0 0.01 0.01 0.01 0
random 0 0.01 0.02 0.03 0 0 0 0 0 0.03 0.01 0.16 0.73 0.01 0 0.01 0.01 0 0 0 0
body0b 0 0 0.02 0.1 0 0 0 0 0 0.04 0.02 0 0.08 0.65 0 0.04 0 0.02 0 0.02 0
body2b 0 0 0.07 0 0 0 0 0 0 0.04 0.05 0.02 0.05 0.09 0.53 0.05 0.04 0.02 0.02 0.02 0.02
face0b 0 0.02 0.05 0.09 0 0 0 0 0 0.09 0 0 0.02 0.14 0 0.48 0.07 0.02 0.02 0 0
face2b 0 0 0.04 0.02 0 0 0 0 0 0.06 0 0 0 0.06 0.04 0.1 0.62 0.02 0.02 0.02 0
place0b 0 0.02 0 0.06 0 0 0 0.02 0 0.02 0.02 0.04 0.06 0 0.02 0.04 0.06 0.47 0.15 0 0
place2b 0 0 0.09 0.02 0 0 0 0 0 0 0.02 0.02 0.02 0.02 0.02 0.09 0.02 0.13 0.55 0 0
tool0b 0 0.08 0.06 0.04 0 0 0 0 0 0.06 0.06 0.02 0.04 0.14 0.02 0.02 0 0.02 0 0.44 0
tool2b 0 0 0.11 0.04 0 0 0 0.02 0 0.11 0.05 0.02 0.09 0.09 0.02 0 0.02 0 0.04 0.02 0.4
fear shape math story footL footR handL handR tongue match relational mental random body0b body2b face0b face2b place0b place2b tool0b tool2b
0.9 0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0


sub-05 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear 0.55 0 0.04 0.06 0 0 0.02 0 0 0 0.02 0.08 0.16 0 0.02 0.04 0 0 0 0 0
shape 0.01 0.6 0.06 0.03 0 0 0 0 0 0.04 0 0.1 0.13 0 0.01 0 0 0.01 0 0 0
math 0.01 0 0.85 0.08 0 0 0 0 0 0 0.01 0.04 0.02 0 0.01 0 0 0 0 0 0
story 0.01 0 0.12 0.82 0 0 0.01 0 0 0.01 0 0.02 0.02 0 0 0 0 0 0 0.01 0
footL 0 0 0.1 0.03 0.55 0 0.03 0 0 0.03 0 0.03 0.17 0.03 0 0 0 0 0 0 0
footR 0 0.04 0.08 0.04 0.04 0.65 0 0 0 0 0.04 0 0.04 0 0.04 0.04 0 0 0 0 0
handL 0 0 0.13 0.07 0 0 0.73 0 0 0 0.03 0 0.03 0 0 0 0 0 0 0 0
handR 0 0 0.03 0.18 0 0 0.06 0.41 0 0.09 0 0.06 0.06 0 0 0.06 0 0 0 0.06 0
tongue 0 0.04 0.08 0.04 0 0 0 0 0.62 0.04 0 0.12 0.08 0 0 0 0 0 0 0 0
match 0.02 0.02 0.02 0.01 0 0 0 0 0 0.79 0.05 0 0.01 0.01 0.02 0.01 0.02 0 0 0.03 0
relational 0.01 0.04 0.07 0.03 0.01 0 0 0 0 0.11 0.59 0.04 0.03 0.02 0.02 0.02 0 0 0 0 0
mental 0.02 0.01 0.03 0.03 0.01 0.01 0 0 0 0.01 0.02 0.69 0.18 0 0 0.01 0 0 0 0.01 0
random 0.01 0.02 0.07 0.01 0 0 0 0 0 0.02 0.02 0.13 0.68 0.01 0 0.01 0.01 0 0 0 0.01
body0b 0 0.02 0 0 0 0 0 0 0 0.02 0.07 0.12 0.05 0.46 0.12 0.04 0.04 0.04 0 0.04 0
body2b 0 0.04 0.04 0.04 0 0 0 0 0 0.09 0 0.15 0.04 0.06 0.51 0 0 0 0 0.02 0
face0b 0 0 0.02 0.04 0 0.04 0 0.02 0 0.06 0.08 0.02 0.08 0 0 0.6 0.02 0.02 0 0 0
face2b 0.04 0.05 0.04 0.02 0 0.02 0 0 0.02 0.02 0.02 0.04 0.04 0 0.02 0.12 0.52 0 0.02 0.02 0.02
place0b 0 0 0.02 0.02 0 0.02 0 0 0 0.02 0.02 0.02 0.02 0.02 0.04 0 0 0.73 0.02 0.02 0
place2b 0.02 0.02 0.05 0 0 0.02 0 0 0 0.03 0 0.02 0.07 0 0 0.05 0.02 0.17 0.53 0 0
tool0b 0.02 0.02 0.02 0.02 0 0 0 0 0 0.05 0 0.04 0.07 0.04 0.04 0.04 0.07 0.02 0 0.55 0.02
tool2b 0 0.02 0.09 0.02 0 0.02 0 0 0 0.05 0.07 0 0.09 0 0 0 0.05 0 0.02 0.05 0.52
fear shape math story footL footR handL handR tongue match relational mental random body0b body2b face0b face2b place0b place2b tool0b tool2b
0.8 0.7 0.6 0.5 0.4 0.3 0.2 0.1 0.0


sub-06 Support Vector Machine using schaefer1000, 3volumes HRFlag
predicted labels
true labels
fear
shape
math
story
footL
footR
handL
handR
tongue
match
relational
mental
random
body0b
body2b
face0b
face2b
place0b
place2b
tool0b
tool2b
0.0
0.1
0.2
0.3
0.4
0.5
0.6
0.7
0.8