Improving Whole-Brain Neural Decoding of fMRI with Domain Adaptation

Shuo Zhou[a], Christopher R. Cox[b,c], Haiping Lu[a,d,*]

[a] *Department of Computer Science, the University of Sheffield, Sheffield, UK*
[b] *School of Biological Sciences, the University of Manchester, Manchester, UK*
[c] *Department of Psychology, Louisiana State University, Baton Rouge, Louisiana, USA*
[d] *Sheffield Institute for Translational Neuroscience, Sheffield, UK*

Abstract

In neural decoding, there has been a growing interest in *machine learning* on *whole-brain* functional magnetic resonance imaging (fMRI). However, the massive size discrepancy between the feature space and the training set poses serious challenges. Simply increasing the number of training examples is infeasible and costly. In this paper, we proposed a *domain adaptation framework for whole-brain fMRI* (DawfMRI) to improve whole-brain neural decoding on *target* data leveraging pre-existing *source* data. DawfMRI consists of three steps: 1) *feature extraction* from whole-brain fMRI, 2) source and target *feature adaptation*, and 3) source and target *classifier adaptation*. We evaluated its eight possible variations, including two non-adaptation and six adaptation algorithms, using a collection of seven task-based fMRI datasets (129 unique subjects and 11 cognitive tasks in total) from the OpenNeuro project. The results demonstrated that appropriate source domain can help improve neural decoding accuracy for challenging classification tasks. The best-case improvement is 8.94% (from 78.64% to 87.58%). Moreover, we discovered a plausible relationship between psychological similarity and adaptation effectiveness. Finally, visualizing and interpreting voxel weights showed that the adaptation can provide additional insights into neural decoding.

*Corresponding author
Email addresses: szhou20@sheffield.ac.uk (Shuo Zhou), chriscox@lsu.edu (Christopher R. Cox), h.lu@sheffield.ac.uk (Haiping Lu)

Keywords: Whole-brain fMRI, neural decoding, classification, domain adaptation, machine learning

1. Introduction

Functional magnetic resonance imaging (fMRI) is a medical imaging technique that records the Blood Oxygenation Level Dependent (BOLD) signal caused by changes in blood flow (Ogawa et al., 1990). Typically, an fMRI sequence is composed of MRI volumes sampled every few seconds, where each MRI volume has over 100,000 voxels and each voxel represents the aggregate activity within about $2mm^3$ volume. fMRI can measure the neural activity associated with multiple cognitive behaviors and examine brain function in healthy relative to disordered individuals. This makes fMRI very useful for neural decoding (Haynes and Rees, 2006) and brain disease diagnosis (Bullmore et al., 1999), which are both *classification* problems in *machine learning*. Moreover, both neural decoding and diagnosis require good machine learning models, as well as the capability of model interpreting and understanding which brain areas are important for classification. The primary challenge with using an fMRI dataset to train a machine learning model is that there are an enormous number of voxels but relatively small number of training examples, e.g., typically less than one hundred per brain state. This makes accurate "out of sample" prediction a challenging problem.

Traditionally, this challenge is dealt with by preselecting voxels that belong to regions of interest (ROIs) (Poldrack, 2007) based on prior work and the established knowledge of domain experts, or by performing a "searchlight" analysis (Kriegeskorte et al., 2006). While making the problem computationally more tractable, it may ignore a significant portion of information in fMRI, and drop potentially valid and superior solutions from consideration in the first place. Recently, studies of *whole-brain fMRI* are becoming increasingly popular (Allen et al., 2014; Gonzalez-Castillo et al., 2015; Vu et al., 2015). This approach not only broadens the scope of potentially important differences between cognitive

states, but also lifts the need for a priori assumptions about which parts of the brain are most relevant. Whole-brain fMRI analysis can take all available information into account in a more *data-driven* workflow, however they can be severely hindered by their small training sets.

While any individual fMRI dataset will provide limited number of training examples, there are many prior experiments designed to support fundamentally related classification problems, e.g., in terms of the cognitive mechanisms that underlie the brain states in different experimental conditions. If we can leverage these related data to improve the performance of machine learning, we can reveal the real power of big data resources, such as OpenNeuro (Gorgolewski et al., 2017) and Human Connectome Project (HCP) (Van Essen et al., 2012).

Domain adaptation, or more broadly *transfer learning* (Pan and Yang, 2010), is an emerging machine learning scheme for solving such a problem. It aims at improving the performance of a particular classification problem by utilizing the knowledge from different but related data. In domain adaptation terminology, the data to be classified are called *target domain data*, while the data to be leveraged are called *source domain data* (Pan and Yang, 2010). The effectiveness of domain adaptation has been shown in varied domains such as computer vision and natural language processing (Patel et al., 2015; Weiss et al., 2016). Domain adaptation has also been applied to brain imaging data for both neural decoding and disease diagnosing, as summarized below.

For neural decoding, Zhang et al. (2018) proposed two transfer learning approaches by making use of shared subjects between target and source datasets. They employed two factorization models (Varoquaux et al., 2011; Chen et al., 2015) to learn subject-specific bases, which are assumed to be invariant across datasets. Using a source dataset with shared subjects can help the models to learn better subject bases, and therefore to improve the prediction accuracy. However, the approaches are not applicable when no subjects are shared between target and source datasets, which is common for multi-site data sharing projects.

Rao et al. (2013) proposed sparse overlapping sets lasso (SOS Lasso) for fMRI, with a multi-task learning (MTL) approach, a branch of transfer learn-

ing. MTL aims at improving the performance of all tasks considered and does not differentiate source and target domains. In the context of SOS Lasso, the multiple "tasks" are datasets associated with each of several *participants* of *the same* experiment. The tasks are related as in multitask group lasso (Yuan and Lin, 2006), but groups can be overlapped with one another, and features can be sparsely selected both within and across groups. SOS Lasso regularizes the solution based on the ℓ_1 norm over all features and the ℓ_2 norm within groups. This technique uses all data relevant to a specific classification problem, but it does not leverage similarities among different classification problems.

In brain disease diagnosis, Ghafoorian et al. (2017) applied deep learning based domain adaptation to brain MRI for lesion segmentation. A convolutional neural network is trained on source domain of 280 patients and the last several layers are fine-tuned on target domain of 159 patients. Wachinger et al. (2016) proposed an instance re-weighting framework to improve the accuracy of Alzheimer's Disease (AD) diagnosis by making the source domain data to have similar distributions as target domain data. Cheng et al. (2012, 2015a,b, 2017) proposed several workflows to perform domain adaptation to improve AD diagnosis accuracy by leveraging data of mild cognitive impairment (MCI), which is considered as the early stage of AD.

Despite these progresses in the broad domain of neuroimaging, to the best of our knowledge, domain adaptation has not been studied systematically for whole-brain fMRI data. This paper proposes a ***D**omain **a**daptation framework for **w**hole-brain **fMRI*** (DawfMRI) to improve the classification performance of a *target domain* classification problem with the help of *source domain* data. This enables us to systematically study domain adaptation for whole-brain fMRI to evaluate and further develop feasible solutions, as well as to discover novel findings and identify key technical challenges. Our main contributions are twofold:

1. Workflow design and implementation: We formulated the DawfMRI framework consisting of three steps: 1) *feature extraction* from whole-brain fMRI, 2) source and target *feature adaptation*, and 3) source and target

classifier adaptation. Under this framework, we identified a state-of-the-art realization of each step and evaluated all eight possible variations, including two non-adaptation and six adaptation algorithms. Our source code is available at: `https://github.com/sz144/DawfMRI`.

2. Results and interpretation: We designed experiments systematically using a collection of tasks from the OpenNeuro/OpenfMRI[1] project. Results demonstrated a promising way of leveraging existing source data to improve neural decoding on target data. We discovered a plausible relationship between psychological similarity and adaptation effectiveness. We also provided visualization and interpretation to show how it works and provides additional insights.

2. Materials and Methods

2.1. OpenfMRI Data

We chose seven OpenfMRI datasets[2] used in the study reported in (Poldrack et al., 2013). Table 1 lists the details of the selected datasets. OpenfMRI is a public fMRI data sharing project which focuses on whole-brain task-based fMRI data. It provides functional MRI data, with structural MRI and metadata. Both functional and structural images are in the NIfTI format. The metadata record experiment-related information, such as onset time, length, and weighting.

The seven datasets contain 11 tasks in total. We used the same task ID as Poldrack et al. (2013). Tasks 2, 3, 4 and tasks 8, 9, 10, respectively, are contributed by the same subjects. We used the original version (revision version 1.0.0) of each dataset. In total, there are 202 fMRI sequences from 129 unique subjects for run 1 and 188 sequences from 116 unique subjects for run 2.

[1] We used the data from OpenfMRI project (`https://openfmri.org/`), now known as OpenNeuro (`https://openneuro.org/`). We will use OpenfMRI in the rest of this paper.

[2] Data used in this paper are available at: `https://legacy.openfmri.org/dataset/`, now in the new BIDS format at OpenNeuro: `https://openneuro.org/public/datasets`

Table 1: List of OpenfMRI datasets used in our experiments (ACN denotes accession number, #Sub denotes the number of subjects in each run, and #Run denotes the number of runs used in our experiments).

Task ID	ACN	#Sub	#Run	Task & Contrast Description
01	ds001	16	2	Balloon analog risk task: Parametric pump effect vs. control
02	ds002	17	2	Probabilistic classification: Task vs. baseline
03	ds002	17	2	Deterministic classification: Feedback vs. baseline
04	ds002	17	2	Mixed event-related probe: Task vs. baseline
05	ds003	13	1	Rhyme judgement: Task vs. baseline
06	ds005	16	2	Mixed-gambles task: Parametric gain response
08	ds007	20	2	Stop signal task: Letter classification vs. baseline
09	ds007	19	2	Stop signal task: Letter naming vs. baseline
10	ds007	20[3]	2	Stop signal task: Pseudoword naming vs. baseline
21	ds101	21	2	Simon task: Incorrect vs. correct
22	ds102	26	2	Flanker task: Incongruent vs. congruent

2.2. Data Preprocessing Pipeline

We implemented a standard data preprocessing pipeline as shown in Table 2, using FSL (Jenkinson et al., 2012) to process the data from OpenfMRI. The output of one step will be the input of next step. The procedure has five steps:

1. Perform motion correction to the BOLD signal sequences from OpenfMRI using MCFLIRT (FSL).
2. Perform brain extraction to the BOLD data using BET (FSL).
3. Perform first-level analysis, to generate statistical parametric maps (SPMs) (Friston et al., 1994, 1998) of contrasts for each experiment condition using FEAT (FSL). FSL design files are generated from the OpenfMRI onsets files using the custom code reimplemented from (Poldrack et al., 2013).

[3]Only 19 subjects were involved in run 2 for this task.

Table 2: Preprocessing pipeline for the selected OpenfMRI data.

Steps	Operation Description	Tools Used
1	Motion correction	FSL mcflirt
2	Brain extraction	FSL bet
3	Within-run statistical analysis	FSL feat
4	Alignment of Z statistic maps	FSL featregapply
5	Vectorization of Z statistic maps masked by MNI152 T1 2mm brain mask	Nibabel

4. Align the spatially normalized Z statistic maps among the SPMs obtained with the MNI152 standard image using featregapply (FSL). The data dimension is standardized to $109 \times 109 \times 91$ ($2mm^3$) voxels.
5. Vectorize the voxels from the Z statistic maps that fall within the standard MNI152 T1 $2mm$ brain mask (distributed with FSL) using the Python package Nibabel (Brett et al., 2017).

The contrasts associated with each task represent differences between the primary tasks and some baseline conditions. Rather than considering the influence of various baseline conditions, we conducted our experiments using the same single contrast per task as Poldrack et al. (2013). The contrasts used are also reported in Table 1.

2.3. Domain Adaptation for Whole-Brain fMRI

We formulated a domain adaptation framework for whole-brain fMRI (DawfMRI) as shown in Fig. 1. This framework consists of three steps: feature extraction, feature adaptation, and classifier adaptation.

- *Feature extraction* aims to distill informative and non-redundant features from high-dimensional data to accelerate computation, reduce overfitting, and facilitate interpretation. Whole-brain fMRI is very high-dimensional. However, there are not as many meaningful components as the number

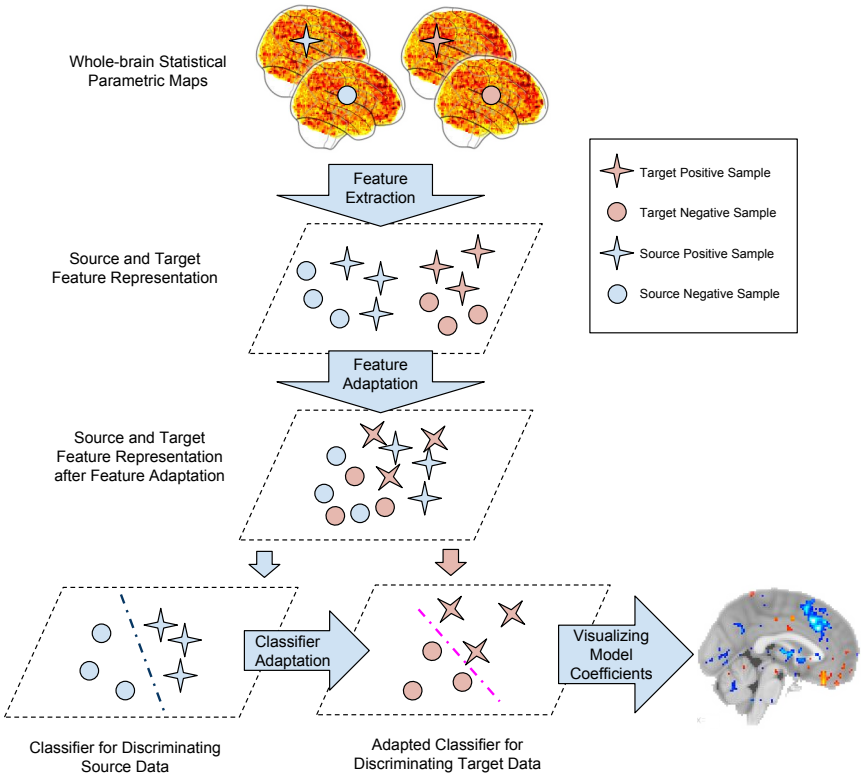


Figure 1: The proposed domain adaptation framework for whole-brain fMRI consists of three steps: feature extraction, feature adaptation, and classifier adaptation, e.g. via independent component analysis, transfer component analysis, and cross-domain SVM, respectively. Learned model can be visualized on a brain atlas for cognitive interpretation.

of voxels. Thus there is high redundancy and feature extraction can be applied to reduce the dimension by identifying a more concise set of informative features, e.g. with principal component analysis or independent component analysis.

- *Feature adaptation* is a domain adaptation scheme that utilizes the source domain samples for target model training. The motivation is to leverage the samples from a related (source) domain when the information provided by target domain samples is limited for training a good model. However, a classifier trained on source domain data typically can not perform well for

classifying target samples. This is due to domain feature distribution mismatch, which means that the features in target and source domain do not follow the same probability distribution. The objective of feature adaptation is to minimize this mismatch by feature mapping or re-weighting (Pan and Yang, 2010). After feature adaptation, the samples from source domain can be used as additional samples for training the target model.

- *Classifier adaptation* is another domain adaptation scheme that aims at improving the classifier performance in a target domain using the knowledge, such as coefficients or parameters, from a pre-trained classifier. The motivation is that when the information provided by target domain data is limited to train a good classifier, the discriminative information that a classifier learned from source data can be leveraged to train a better target classifier. It should be noted that classifier adaptation only emphasizes the performance on target domain. How the model will perform on the source domain after adaptation is not considered.

Each step of DawfMRI can be optional. If all the three steps are skipped, we train a classifier directly on the whole-brain data. To study DawfMRI systematically, we employ a state-of-the-art method for each step in DawfMRI: independent component analysis (ICA) (Hyvärinen and Oja, 2000) for feature extraction, transfer component analysis (TCA) (Pan et al., 2011) for feature adaptation, and cross-domain SVM (CDSVM) (Jiang et al., 2008) for classifier adaptation. Table 3 lists the key notations used for easy reference.

2.3.1. Feature Extraction by ICA

ICA is a popular method for fMRI feature extraction. In our experiments, we firstly performed spatial smoothing on the SPMs obtained from the fourth step in Table 2. Then we performed ICA on whole-brain smoothed SPMs using MELODIC (FSL) (Beckmann and Smith, 2004).

Remark: The *feature extraction* step does not differentiate source or target domains. It is unaware of the distribution difference between the two domains

Table 3: Notations and descriptions.

Notation	Description
D	Input feature dimension
$\mathbf{D}_l^t$	Labeled target data
d	Output (lower) feature dimension
$\mathbf{H}$	Centering matrix
$\mathbf{I}$	Identity matrix
$\mathbf{K}$	Kernel matrix
$k(\cdot,\cdot)$	Kernel function, e.g. linear, RBF, polynomial
n_T, n_S	Target/source sample size
$\mathbf{U}$	Transformation matrix
$\mathbf{V}^s$	Source support vectors
$\mathbf{w}$	Classifier coefficients
$\mathbf{X}_T$, $\mathbf{X}_S$	Target/source data
$\mathbf{Z}_T$, $\mathbf{Z}_S$	Learned target/source representation

and does not attempt to minimize such difference. In other words, there is no *adaptation* in the *feature extraction* step. Hence we can apply ICA to all fMRI datasets for better estimation quality.

2.3.2. Feature Adaptation by TCA

TCA aims to find a feature mapping to minimize the mismatch between target and source distributions. Maximum mean discrepancy (MMD) (Borgwardt et al., 2006) is employed as the distribution mismatch metric. Given source domain data $\mathbf{X}_S \in \mathbb{R}^{D\times n_S}$, target domain data $\mathbf{X}_T \in \mathbb{R}^{D\times n_T}$, where n_S and n_T denote the numbers of samples of $\mathbf{X}_S$ and $\mathbf{X}_T$ respectively, and D denotes

the input feature dimension. MMD between the two domains is

$$
\begin{aligned}
MMD(\mathbf{X}_S, \mathbf{X}_T) &= \left\| \frac{1}{n_S} \sum_{i=1}^{n_S} \mathbf{x}_{S_i} - \frac{1}{n_T} \sum_{i=1}^{n_T} \mathbf{x}_{T_i} \right\|_{\mathcal{H}}^2 \\
&= \Big[\frac{1}{n_S^2} \sum_{i,j=1}^{n_S} k(\mathbf{x}_{Si}, \mathbf{x}_{Sj}) - \frac{2}{n_S n_T} \sum_{i,j=1}^{n_S, n_T} k(\mathbf{x}_{Si}, \mathbf{x}_{Tj}) \\
&\quad + \frac{1}{n_T^2} \sum_{i,j=1}^{n_T} k(\mathbf{x}_{Ti}, \mathbf{x}_{Tj}) \Big],
\end{aligned} \tag{1}
$$

where $\mathcal{H}$ denotes a reproducing kernel Hilbert space (RKHS) (Berlinet and Thomas-Agnan, 2011), $k(\cdot, \cdot)$ denotes a kernel function, such as linear, RBF, and polynomial, $\mathbf{x}_{Si}$ and $\mathbf{x}_{Tj}$ are the ith and jth sample of $\mathbf{X}_S$ and $\mathbf{X}_T$, respectively. TCA aims at minimizing marginal distribution mismatch, i.e, $P(\mathbf{X}_S) \neq P(\mathbf{X}_T)$. The objective is to learn a feature representation $\mathbf{Z}_S$ and $\mathbf{Z}_T$ by mapping the input data to a feature space where the MMD between the two domains is minimized, i.e., $P(\mathbf{Z}_S) \approx P(\mathbf{Z}_T)$. Equation (1) can be rewritten as $MMD(\mathbf{X}_S, \mathbf{X}_T) = tr(\mathbf{KL})$, where

$$
\mathbf{K} \in \mathbb{R}^{n \times n} = k([\mathbf{X}_S, \mathbf{X}_T], [\mathbf{X}_S, \mathbf{X}_T]) = \begin{bmatrix} \mathbf{K}_{S,S} & \mathbf{K}_{S,T} \\ \mathbf{K}_{T,S} & \mathbf{K}_{T,T} \end{bmatrix}, \tag{2}
$$

$\mathbf{L} \in \mathbb{R}^{n \times n}$ is:

$$
L_{ij} \begin{cases} \frac{1}{{n_S}^2} & \text{if} \quad \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}_S, \\ \frac{1}{{n_T}^2} & \text{if} \quad \mathbf{x}_i, \mathbf{x}_j \in \mathbf{X}_T, \\ -\frac{1}{n_S n_T} & \text{otherwise}, \end{cases} \tag{3}
$$

and $n = n_S + n_T$. To minimize MMD, TCA employs a dimension reduction approach. A matrix $\tilde{\mathbf{U}} \in \mathbb{R}^{n \times d}$ is used to map the kernel features to a d-dimensional space ($d \ll n_S + n_T$), then we obtain a new kernel matrix

$$
\tilde{\mathbf{K}} = (\mathbf{K}\mathbf{K}^{\frac{1}{2}}\tilde{\mathbf{U}})(\tilde{\mathbf{U}}^\top \mathbf{K}^{\frac{1}{2}}\mathbf{K}). \tag{4}
$$

We can consider $\tilde{\mathbf{K}}$ as the inner product of the new representation $\mathbf{Z} \in \mathbb{R}^{d \times n} = [\mathbf{Z}_S, \mathbf{Z}_T]$, and therefore $\mathbf{Z} = \tilde{\mathbf{U}}^\top \mathbf{K}^{\frac{1}{2}}\mathbf{K}$. Let $\mathbf{U} \in \mathbb{R}^{n \times d} = \mathbf{K}^{\frac{1}{2}}\tilde{\mathbf{U}}$, we obtain

$$
tr((\mathbf{K}\mathbf{U}\mathbf{U}^\top \mathbf{K})\mathbf{L}) = tr(\mathbf{U}^\top \mathbf{K}\mathbf{L}\mathbf{K}\mathbf{U}), \tag{5}
$$

In addition, scatter matrix $\mathbf{U}^\top\mathbf{KHKU}$ (Fukunaga, 1990) is employed for preserving the properties (variance) of $\mathbf{X}_S$ and $\mathbf{X}_T$, where $\mathbf{H} \in \mathbb{R}^{n\times n}$ is a centering matrix (Marden, 2014). Consequently, the learning objective becomes

$$\begin{aligned} \min_{\mathbf{U}} \quad & tr(\mathbf{U}^\top\mathbf{KLKU}) + \lambda \cdot tr(\mathbf{U}^\top\mathbf{U}) \\ s.t. \quad & \mathbf{U}^\top\mathbf{KHKU} = \mathbf{I}_d, \quad \lambda > 0, \end{aligned} \tag{6}$$

where $\mathbf{I}_d \in \mathbb{R}^{d\times d}$ is an identity matrix, λ is a tradeoff parameter for regularization. Denoting $\Gamma = diag(\gamma_1, \ldots, \gamma_d)$ as Lagrange multipliers, we can derive the Lagrange function for Eq. (6) as

$$\mathcal{L} = tr(\mathbf{U}^\top(\mathbf{KLK} + \lambda I)\mathbf{U}) + tr((I - \mathbf{U}^\top\mathbf{KHKU})\Gamma). \tag{7}$$

Setting $\frac{\partial \mathcal{L}}{\partial \mathbf{U}} = 0$, we obtain generalized eigendecomposition problem

$$(\mathbf{KLK} + \lambda I)\mathbf{U} = \mathbf{KHKU}\Gamma. \tag{8}$$

Finally, $\mathbf{U}$ can be learned by solving Eq. (8) for the d smallest eigenvectors.

2.3.3. Classifier Adaptation by CDSVM

CDSVM is a kind of SVM classifier that can utilize the support vectors learned by a pre-trained standard SVM to find a better decision boundary for target samples. Each source support vectors will be re-weighted according to its average distance to the target (training) feature vectors, and then the SVM classifier will be trained with the target training samples and re-weighted source support vectors. We learn the decision function of CDSVM $f(\mathbf{x}) = \mathbf{w}^\top\mathbf{x}$ by optimizing the following objective:

$$\begin{aligned} & \min_{\mathbf{w}} \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{M}\xi_i + C\sum_{j=1}^{K}\sigma(\mathbf{v}_j^s, \mathbf{D}_l^t)\bar{\xi}_j \\ & s.t. \quad y_i f(\mathbf{x}_i) \geq 1 - \xi_i, \quad \xi_i \geq 0, \quad \forall(\mathbf{x}_i, y_i) \in \mathbf{D}_l^t, \\ & \quad y_j^s f(\mathbf{v}_j^s) \geq 1 - \bar{\xi}_j, \quad \bar{\xi}_j \geq 0, \quad \forall(\mathbf{v}_j^s, y_j^s) \in \mathbf{V}^s, \\ & \quad \sigma(\mathbf{v}_j^s, \mathbf{D}_l^t) = \frac{1}{M}\sum_{(\mathbf{x}_i, y_i)\in\mathbf{D}_l^t} \exp(-\beta\|\mathbf{v}_j^s - \mathbf{x}_i\|_2^2), \end{aligned} \tag{9}$$

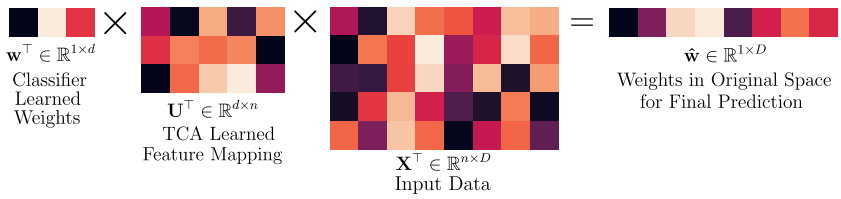


Figure 2: Map classifier coefficients $\mathbf{w}^\top \in \mathbb{R}^{1\times d}$ back to the original feature space $\mathbb{R}^{1\times D}$.

where $\mathbf{D}_l^t$ represents the labeled (training) target domain data, $\mathbf{V}^s$ denotes a matrix composed of all source support vectors, M is the number of samples in $\mathbf{D}_l^t$, K is the number of source support vectors in $\mathbf{V}^s$, $\mathbf{v}_j^s$ is the jth source support vector in $\mathbf{V}^s$, ξ_i and $\bar{\xi}_j$ are slack variables for the ith target feature vector and jth source support vector respectively, C is a hyperparameter controlling the trade-off between the slack variable penalty and the SVM soft margin, $\sigma(\mathbf{v}_j^s, \mathbf{D}_l^t)$ is a function that evaluates the distance between $\mathbf{v}_j^s$ and $\mathbf{D}_l^t$, and β is a hyperparameter controlling the influence of the source support vectors, larger value of β leads to the less influence of source support vectors, and vice versa.

2.4. Visualizing Decoding Model Coefficients for Interpretation

The final classifier coefficients (or weights) indicate the significance of the features for discriminating two classes. This can facilitate the interpretation, e.g. understand which brain areas are important for distinguishing two cognitive tasks. We developed a method to map the weights in lower dimension back to the original input (high-dimensional) space for interpretation. To achieve this, we chose a linear kernel in TCA, SVM and CDSVM. We consider the setting of applying TCA+SVM (or TCA+CDSVM) to whole-brain data. When feature extraction (ICA) is performed, one more step will be required to map the weights in the IC space to the whole-brain space, e.g. by GIFT (Calhoun et al., 2001). Using the same notations above, we have data $\mathbf{X} \in \mathbb{R}^{D\times n}$ composed of target and source data, TCA learned a feature transformation $\mathbf{U} \in \mathbb{R}^{n\times d}$, and SVM learned coefficients $\mathbf{w} \in \mathbb{R}^{d\times 1}$. According to Eq. (5), the features transformed by TCA can be represented as $\mathbf{Z} \in \mathbb{R}^{d\times n} = \mathbf{U}^\top \mathbf{X}^\top \mathbf{X}$. Thus for an arbitrary

sample $\mathbf{x} \in \mathbb{R}^{D\times 1}$, its transformed feature $\mathbf{z} \in \mathbb{R}^{d\times 1} = \mathbf{U}^\top \mathbf{X}^\top \mathbf{x}$. The prediction for its class made by the classifier is

$$\hat{y} = sgn(\mathbf{w}^\top \mathbf{z}) = sgn(\mathbf{w}^\top \mathbf{U}^\top \mathbf{X}^\top \mathbf{x}), \tag{10}$$

where sgn is the sign function extracts the sign of a real number (1 for positive, -1 for negative). Let $\hat{\mathbf{w}} \in \mathbb{R}^{1\times D} = \mathbf{w}^\top \mathbf{U}^\top \mathbf{X}^\top$, we obtain

$$\hat{y} = sgn(\hat{\mathbf{w}}\mathbf{x}). \tag{11}$$

Hence, $\hat{\mathbf{w}}$ contains the weights corresponding to each voxel in whole brain space for final prediction (shown in Fig. 2). This enable us to visualize the weights in a 3D brain atlas for interpretation, e.g. by Nilearn (Abraham et al., 2014).

3. Experimental Results

3.1. Experiment Settings and Evaluation Methods

Our experiments will focus on using domain adaptation to improve performance on challenging binary classification problems that require distinguishing brain states associated with different cognitive tasks. Thus, we only considered the most basic scenario, specifically, both target and source classification problems are binary. We studied DawfMRI in the setting of one-to-one domain adaptation only. This means that one target domain will be leverage only one single other set of source domain data.

3.1.1. Algorithm Setup

We evaluated eight possible variations of the DawfMRI framework, including two aforementioned non-adaptation algorithms: 1) SVM, 2) ICA+SVM, and six adaptation algorithms: 3) CDSVM (Jiang et al., 2008), 4) ICA+CDSVM, 5) TCA+SVM (Pan et al., 2011), 6) TCA+CDSVM, 7) ICA+TCA+SVM, 8) ICA+TCA+CDSVM.

For feature adaptation only algorithms (5 & 7), samples from the source were used with the target training samples together for SVM training after performing

TCA. When involving classifier adaptation (3, 4, 6, and 8), we trained a SVM on the source domain, and then used the learned source support vectors as additional input knowledge for training CDSVM on the target domain.

As mentioned in Sec. 2.4, we used a linear kernel in TCA, SVM, and CDSVM for interpretability. We optimized hyperparameters on regular grids of log scale for each algorithm. We searched for the best C and μ values within the range $[10^{-3}, 10^{3}]$ and $[10^{-5}, 10^{5}]$ for SVM and TCA, respectively. For CDSVM, we searched for best C and β values exhaustively in $[10^{-3}, 10^{3}]$. We also varied the feature dimension of ICA and TCA output from 2 to 100 (2, 10, 20, 50 and 100), and the algorithms were optimized with the best feature dimensions.

3.1.2. Target and Source Domain Setup

The experimental data are from the set of whole-brain SPMs associated with the 11 contrasts (one per run per task) for each subject obtained from the preprocessing pipeline in Sec. 2.2. These contrasts isolate task-related brain activity relative to a baseline condition. We expect the classification problems with lower prediction accuracy have higher potentiality of improvement. Hence, we focused on the target domain pairs that were the most difficult to classify. We determined these pairs by performing multi-class classification with two methods, SVM and ICA+SVM, and then chose the task pairs that were the most confusable.

Specifically, we first applied a linear SVM implemented in the Python package Scikit-learn (Pedregosa et al., 2011) to whole-brain SPMs of all eleven classes without feature extraction to do multi-class classification. Figure 3 shows the 10-fold cross-validation results as a confusion matrix. The classification results indicate that Tasks 1 and 6 were often confused with other tasks. Task 1 was misclassified as Task 22 more than half the time. Task 6 was the least accurately classified overall, of which the samples were often misclassified as Tasks 3, 21, or 22. We noted that samples from Task 3 can also be misclassified as Task 6.

We then repeated the multi-class classification analysis in a low-dimensional ICA feature space, i.e., with ICA+SVM. The most confusable tasks were qual-

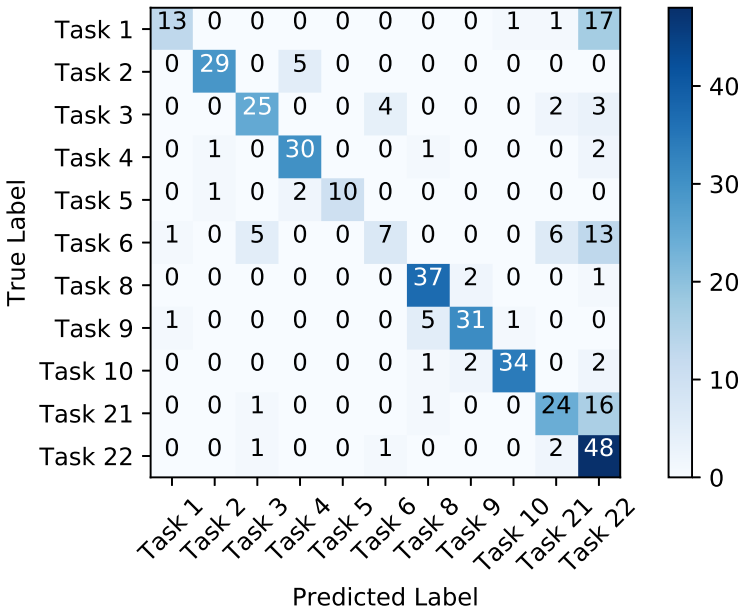


Figure 3: Multi-class classification confusion matrix for linear SVM performance on the whole-brain SPMs. Numbers in the matrix indicate the number of samples. The most challenging pairs of classification tasks (3 vs 6, 6 vs 22, and 1 vs 22) were selected as target domains to perform domain adaptation.

itatively similar to those in Figure 3.

Selected targets: Based on the results above, we selected the following three target domains (see Table 1 for details of the cognitive tasks) for subsequent experiments. Each domain is defined by as a pair of tasks to be classified:

1. Tasks 1 (32 samples) & 22 (56 samples),
2. Tasks 3 (34 samples) & 6 (32 samples),
3. Tasks 6 (32 samples) & 22 (56 samples).

These selected pairs are confirmed later in our adaptation effectiveness study (see Fig. 6) to be among those benefiting the most from domain adaptation.

Source selection and class labels: When using one pair of tasks as the target domain data, the remaining nine tasks are combined pairwise to give 36 unique pairs, each of which is a candidate source domain. For each pair, one

task is labeled 1 and the remaining task is labeled -1, also called positive and negative classes, respectively. We also studied an alternative setting with the labels for source domain tasks reversed in adaptation algorithms.

3.1.3. Evaluation Methods

We performed $10 \times k$-fold cross-validation (CV) evaluation, with $k = 2$, 5, and 10, corresponding to using 50%, 80%, and 90% of available target domain samples for training. All training samples were sampled uniformly at random for cross validation. CV was only applied to target domain samples. Source domain samples all used for training without holdout for testing when performing domain adaptation. We will report the mean classification accuracy with standard derivations for performance evaluation. To study the statistical significance of the results obtained by adaptation algorithms compared to those by non-adaptation algorithms, we will report the p-values of paired t-tests results.

Applying CDSVM directly to the whole-brain SPMs ran out of 50GB memory, so we were unable to complete the analysis. Experimental results for the *remaining seven algorithms* will be reported in the following section.

3.2. Classification Performance Comparisons

3.2.1. Performance on Different Target Domains

Figure 4 presents the 10-CV classification performance across the three different target domains. The adaptation algorithms are displayed with the corresponding source domains for obtaining the best results. The best results on the three target domain were obtained by TCA+CDSVM, ICA+TCA+SVM, and TCA+CDSVM respectively. The three algorithm outperformed the non-adaptation algorithms significantly (maximum p-value < 0.0001 in paired t-test). The largest accuracy improvement was obtained by TCA+CDSVM with source 22 vs 9. This improves over the best non-adaptation method (SVM) by **8.94%** (**from 78.64% to 87.58%**).

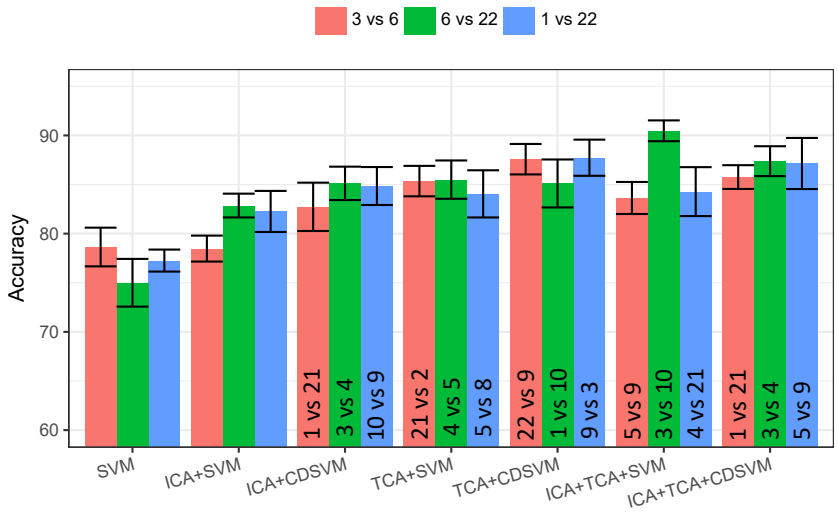


Figure 4: Classification accuracy (in %) obtained on the three target domains. Error bars indicate the standard derivations. Adaptation algorithms are displayed with the classification problems of source domains for obtaining the best results.

3.2.2. Performance on Different Training Sample Sizes

Figure 5 depicts the classification accuracy of seven algorithms. We fixed the target and source to observe how the algorithms perform on different sizes of training data (10/5/2-fold CV). The target domain is 3 vs 6, and the source domain is 10 vs 1. We presented the results with respect to this particular target and source domain because 3 vs 6 is a balanced classification problem, and it yielded the most consistent pattern of improvement over the non-adaptation algorithms for all of the adaptation algorithms. Note that this source domain was not necessarily optimal in all cases.

The best performance 85.60% was given by TCA+CDSVM, under the 10-fold CV (10-CV). Further paired t-tests between the best adaptation results and the best non-adaptation results were statistically significant for 10-CV and 5-CV (p-value < 0.0001). However, for 2-CV, the accuracy improved by TCA+SVM (77.37%) and ICA+TCA+CDSVM (77.27%) over ICA+SVM (76.36%) were not statistically significant, the corresponding p-values were 0.66 and 0.62 respectively, indicating very small training sample size is still challenging.

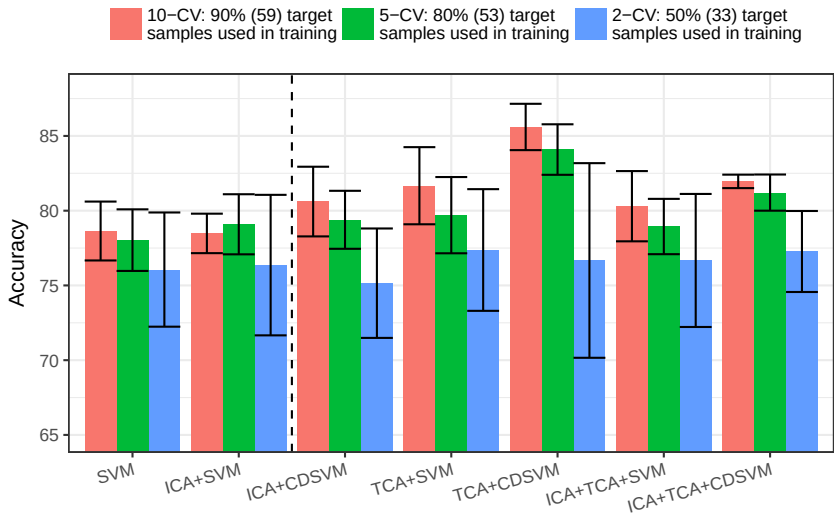


Figure 5: Classification accuracy (in %) of seven methods for three cross validation settings on target classification problem task 3 vs task 6 (positive vs negative), with source task 10 vs task 1. Error bars indicate the standard derivations. Non-adaptation (left) and adaptation (right) methods are separated by the dashed line.

3.2.3. Sensitivity to Source Domain

For studying the *adaptation effectiveness* of different source domains over different target domains, we applied TCA+CDSVM and SVM to all possible combinations of the 11 tasks as target domain. Figure 6 shows the obtained accuracy. The target domains are sorted by their largest improvements (TCA+CDSVM over SVM). A psychological interpretation will be given in Sec. 4.1.

We can observe the effectiveness of domain adaptation was affected by the source domain, and TCA+CDSVM did not consistently outperform SVM. This is called "negative transfer" (Pan and Yang, 2010).

We can also observe that the maximum improvements show a negative correlation with the baseline SVM accuracy. For high SVM accuracy, TCA+CDSVM can hardly perform better, instead worse in many cases (e.g. 8 vs 22, 2 vs 6). This is consistent with our expectation, that domain adaptation is effective for

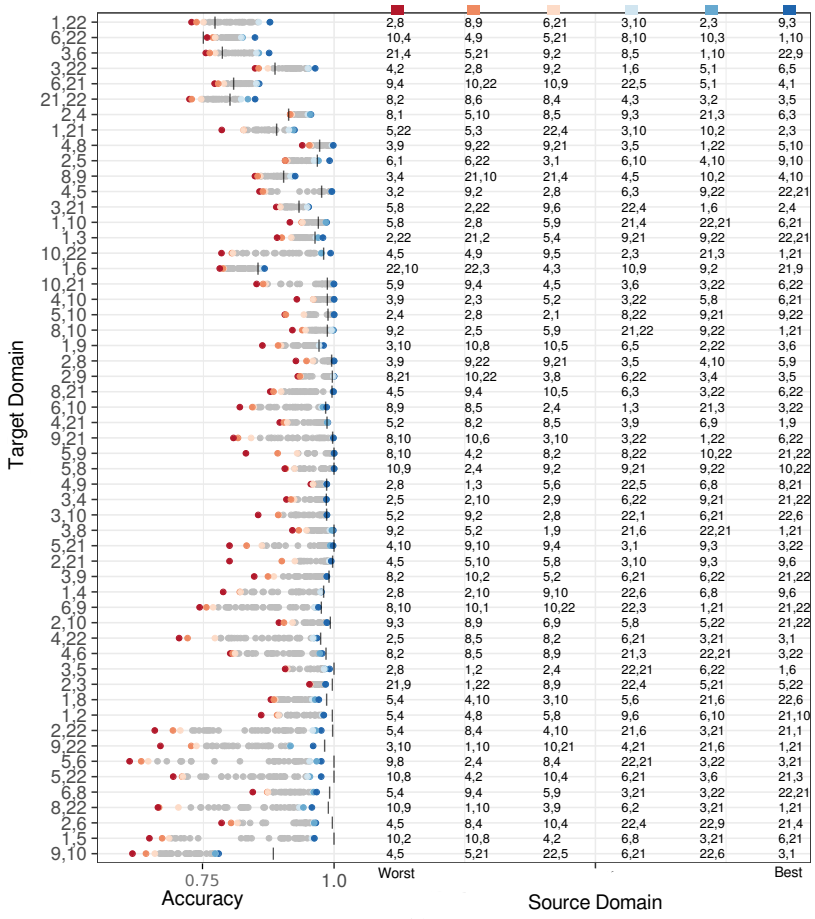


Figure 6: Adaptation effectiveness of TCA+CDSVM (colored dots) over SVM (black vertical bars) across all target domains, with 10-fold cross-validation. Target domains are sorted with respect to the maximum improvement. The top three and bottom three source domains are listed on the right half, in the order of the worst, second worst, third worst, third best, second best, and best, from left to right.

more challenging problems. This is the method behind our target selection strategy in Sec. 3.1.2.

3.2.4. Effectiveness of Each DawfMRI Step

In feature extraction, ICA can extract informative features from high-dimensional whole brain data. By comparing the results in Fig. 4, we can observe that

ICA+SVM outperform SVM for the targets 1 vs 22 and 6 vs 22. For target 3 vs 6, the accuracy obtained by ICA+SVM is slightly lower than the one obtained by SVM. Considering the much lower dimension of ICs compared with the dimension of whole-brain data, ICA is an effective feature extractor.

In feature adaptation, TCA can extract common and useful features across source and target domains. TCA can also extract features with lower dimension from whole-brain SPMs without performing ICA. Moreover, as shown in Figures 4 and 5, both TCA+SVM and ICA+TCA+SVM outperformed non-adaptation algorithms with the appropriate sources. This indicates that by performing TCA, samples from appropriate source domains can be used as additional training data for target domain, and SVM performed better with more data for training.

In classifier adaptation, CDSVM can improve the accuracy when combined with ICA or TCA (or both). By comparing the results in Figures 4 and 5, ICA+CDSVM did improve over ICA+SVM consistently, though the amount of improvement was not as large. By contrast, TCA+CDSVM did not outperform TCA+SVM consistently. This indicates that the interaction between the data and each component of DawfMRI can influence the adaptation effectiveness.

4. Analysis and Discussion

This section will further analyze DawfMRI with two objectives: 1) exploring whether adaptation effectiveness is related to psychological similarities between tasks, 2) understanding how domain adaptation improves neural decoding by visualizing the model coefficients.

4.1. Psychological Interpretation of Source Domain Effectiveness

Domain adaptation effectiveness is closely related to meaningful relationships between the target and source domains. On the other hand, psychological experiments are intrinsically related by the cognitive mechanisms that support

Table 4: Statistics of the four psychological similarity features used for logistic model training, which are target domain similarity (TDSim), source domain similarity (SDSim), cross-domain similarity (CDSim) and target domain SVM accuracy (TDSVM_Acc).

	CDSim	TDSim	SDSim	TDSVM_Acc
Min	-1.78	-1.02	-1.21	-1.60
Median	-0.05	-0.59	-0.36	-0.026
Mean	0.00	0.00	0.00	0.00
Max	3.47	1.93	2.73	1.33

the ability to perform the task. Hence, we expect the cognitive similarity between a set of tasks to be predictive of whether or not domain adaptation will be effective.

4.1.1. Psychological Similarity Study

We explored this by modeling the probability of domain adaptation with TCA+CDSVM being effective as a function of the psychological similarity between tasks within and across domains. In order to estimate the psychological similarity, we began by associating each task with a set of cognitive functions that it relies on. The associations were defined by referring to the cognitive concepts in Cognitive Atlas (Poldrack et al., 2011). Out of the 11 tasks in Table 1, 10 were associated with a number of cognitive functions (min = 5, max = 13, median = 7); the mixed event related probe (task 4) had an incomplete entry and so it was excluded from these and further studies. There were 42 functions in total, and each task was represented as a binary feature vector. The psychological similarity among each pair of tasks was taken to be the cosine similarity between their feature vectors.

Because each domain is composed of a pair of tasks, the similarity between the target and source domains in each model is associated with four pairwise task similarities. The overall psychological similarity between each pair of domains was estimated by averaging the four relevant pairwise similarities. We will refer to this estimate as *Cross-Domain Similarity* (CDSim), the similarity

Table 5: Logistic model learning outcome. Top half demonstrates a model regressed on three variables: TDSim, SDSim, and CDSim. Bottom half demonstrates a model regressed on the three variables and one additional variable: TDSVM_Acc.

var	beta	se	tval	pval
(intercept)	-0.07	0.09	0.78	0.43
TDSim	-0.32	0.099	-3.26	0.0011
SDSim	-0.21	0.10	-2.10	0.035
CDSim	0.29	0.11	2.74	0.0062
(intercept)	-0.085	0.097	-0.88	0.38
TDSim	-0.10	0.11	-0.98	0.32
SDSim	-0.20	0.11	-1.89	0.057
CDSim	0.26	0.11	2.38	0.017
TDSVM_Acc	-0.80	0.10	-7.75	9.154e-15

between the two tasks that define the target domain as *Target-Domain Similarity* (TDSim), and the similarity between the two tasks define the source domain as *Source-Domain Similarity* (SDSim).

This study focused on 504 target and source combinations of the classification problems with no more than 90% in accuracy obtained by whole-brain SVM. Of these 504 models, 261 was improved. We labeled them as improved or not improved to train a logistic model. This binary outcome was firstly regressed on the three aforementioned variables, CDSim, TDSim, and SDSim. We also ran a version of this model to include the accuracy of the basic SVM in the target domain (TDSVM_Acc) as a covariate, to control for overall difficulty and statistically remove this from the other variables. The variables were standardized to have a mean of zero and standard deviation of one before fitting the model. Table 4 lists the statistics of the four variables.

Table 5 reports the learning outcome of the two logistic models. In both cases, increasing CDSim increased the probability of improved accuracy. Moreover, when TDSVM_Acc is included, the significance of TDSim decreased, which

indicates the baseline SVM accuracy can be correlated with psychological similarity. This relationship between psychological similarity and effectiveness is important, not only because it is consistent with the how these adaptive methods are meant to work, but because it suggests it may be possible to anticipate the adaptation effectiveness in advance, without resorting to a post-hoc selection of the source domain through trial and error.

4.1.2. Source Selection Validation

We also adopted leave one target domain out strategy to learn to "select" an appropriate source domain. For the hold out target domain, we selected the domain with the highest likelihood of accuracy improvement given by the logistic model as source. Then we compared the real improvement of using the selected source data with random source selection (mean improvement) for TCA+SVM, TCA+CDSVM, and ICA+TCA+SVM. Results showed that psychological similarity based source selection are 0.0068, 0.0065, and 0.0371 higher than random selection respectively, and it can help source selection.

In addition, we did the same analysis to TCA+SVM to compare MMD-based source selection with random source selection. We computed the MMDs for all possible target and source combination using Eq. (5) after applying TCA, as well as the accuracy of respective TCA+SVM. Results showed that selecting source domains with smallest MMD can achieve 0.064 higher on average in accuracy than random source selection.

4.2. Neural Decoding Visualization and Cognitive Interpretation

Beyond practical utility in building better predictive models, advances in cognitive neuroscience and understanding the physiology of neural disorder require understanding why a model performs well and which brain networks are particularly important. Considering which kind of model tends to emerge through domain adaptation will provide insight into how these methods work, computationally, and the kind of cognitive similarity that is being leveraged through domain adaptation. If we consider the set of voxels with high weights

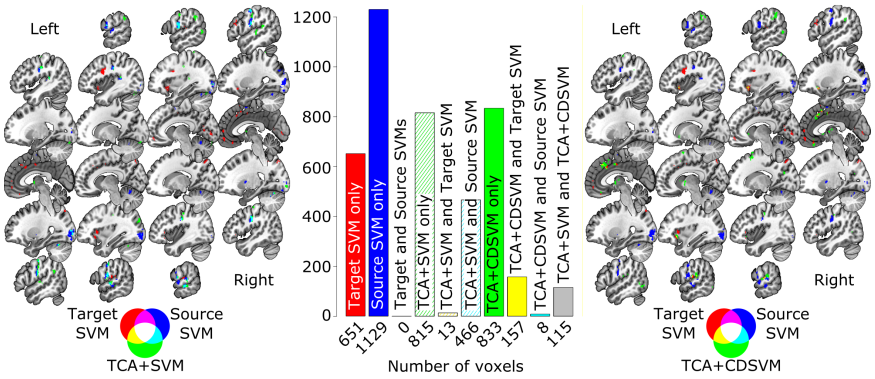
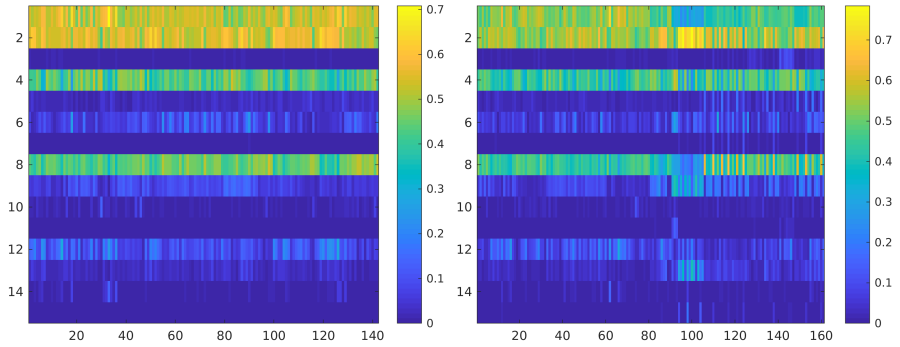


Figure 7: The visualization for four models: target SVM, source SVM, TCA+SVM, and TCA+CDSVM. The model coefficients are filtered by absolute value (top 1%) and cluster size (≥ 20). Number of distinct and overlapped voxels identified by the models are shown in the bar chart in the middle.

in magnitude and occur in clusters in each model, where do these clusters exist in the brain, and how much do the voxel sets in the target domain, source domain, and adapted models overlap?

To visualize such models, we studied the case that showed the most dramatic improvement in classification accuracy through domain adaptation. Figure 7 shows the voxels where the corresponding coefficients of four models (target SVM, source SVM, TCA+SVM, and TCA+CDSVM) are weighted in the 99th percentile by magnitude and occur in clusters of at least 20 voxels. In this case, the target and source domain SVMs place their important voxels in completely different areas. Not only is there no overlap, but the supra-threshold voxels in each model are sampled from different lobes of the brain: the target domain SVM is associated primarily with supra-threshold voxels in the frontal lobes, while the source domain SVM is associated primarily with supra-threshold voxels in the occipital lobe and sensory motor cortex. The distribution of weight is so different between source and target models, but adaptation can be nevertheless very effective.

We then overlaid the adapted models from TCA+SVM and TCA+CDSVM.



(a) Overlap in Improved Models (b) Overlap in not Improved Models

Figure 8: The number of overlapped voxels between four models: target SVM, source SVM, TCA+SVM, and TCA+CDSVM. The model coefficients are filtered by absolute value (top 1%). Improved (left) and not improved (right) models are presented separately. The x-axis denotes different target source combinations, and y-axis denotes 15 relationship between four models: 1) Target SVM only, 2) Source SVM only, 3) Target and Source, 4) TCA+SVM only, 5) TCA+SVM and Target SVM, 6) TCA+SVM and Source SVM, 7) TCA+SVM and Target SVM and Source SVM, 8) TCA+CDSVM only, 9) TCA+CDSVM and Target SVM, 10) TCA+CDSVM and Source SVM, 11) TCA+CDSVM and Target SVM and Source SVM, 12) TCA+SVM and TCA+CDSVM, 13) TCA+SVM and TCA+CDSVM and Target SVM, 14) TCA+SVM and TCA+CDSVM and Source SVM, 15) TCA+SVM and TCA+CDSVM and Target SVM and Source SVM.

The TCA+SVM has substantial overlap with the source domain, and nearly no overlap with the target domain model. This substantial overlap, however, is only about 1/3 of the supra-threshold voxels in the TCA+SVM model. The remaining 2/3 are completely distinct from either the target or source model, indicating that information from the source domain has revealed a different dimension along to which to dissociate the tasks in the target domain than was apparent in the target data in isolation.

This general pattern is echoed in the model adapted with TCA+CDSVM, except that in this case there is virtually no overlap with the source domain and there is instead modest overlap with the target domain. Again, the adapted model is largely associated with supra-threshold voxels that do not overlap with either the target or source SVMs. Thus, the adaptation procedure has provided

additional insight into the classification problem, exploiting information that is more than the simple sum of the target and source domain models.

We further extended this study to analyze the overlapped voxels in the four aforementioned models for all of the of target and source combinations under the threshold of TDSVM_Acc ≤ 90. For each target and source pair, there is a 15 element vector, which encodes all possible relationship between the four models. Figure 8 depicts the values in the vectors, including the frequency of each kind of relationship. The results are consistent with the findings of the above case study On one hand, there is a clear effect that each model identifies a fairly distinct set of voxels (row 4 and 8 in Fig. 8a), which are larger than the number overlapped voxels between non-adaptation and adaptation models. On the other hand, by comparing the results shown in row of 5, 6, 9 and 10 of Fig. 8a, we can find that TCA+SVM overlaps with the source domain SVM much more than the target domain SVM, and the opposite is true for the TCA+CDSVM.

4.3. Technical Challenges

Through this work, we have identified two technical challenges in DawfMRI. One is *how to select a good source domain automatically (without exhaustive testing/validation) to reduce or even avoid "negative transfer"*, as observed in our experiments. The plausible relationship between psychological similarity and adaptation effectiveness in Sec 4.1 can lead to a better than random solution. However, it is not the optimal selection of sources. The other challenge is *how to make use of multiple source domains to further improve the classification performance*. We can see such a need when observing that domain adaptation is not effective when the number of training samples is very small, from the 2-CV results in Fig. 5. We need to carefully leverage the positive effects from each source domain while minimizing the potential negative impacts, which needs a smart, adaptive procedure to be introduced. We consider both challenges are important directions to explore in the future.

5. Conclusion

In this paper, we proposed a domain adaptation framework for whole-brain fMRI (DawfMRI). DawfMRI consists of three key steps: feature extraction, feature adaptation and classifier adaptation. We employed three state-of-the-art algorithms, ICA, TCA and CDSVM, for DawfMRI. We studied two non-adaptation algorithms and six possible adaptation algorithms on task-based whole-brain fMRI from seven OpenfMRI datasets. Results show that DawfMRI can improve the classification performance for challenging binary classification tasks. We also observed "negative transfer" in the experiments, which indicates domain adaptation does not always give better performance and should be used with care. Furthermore, we made a preliminary attempt in interpreting our how the models work, and pointed out two important research directions to pursue in future work. We will further enhance the DawfMRI framework and tackle the two technical challenges.

Acknowledgment

The authors would like to thank the OpenfMRI (now OpenNeuro) project for providing the public fMRI datasets for this research work, and Russell Poldrack for providing help in reimplementing the OpenfMRI preprocessing pipeline.

References

Abraham, A., Pedregosa, F., Eickenberg, M., Gervais, P., Mueller, A., Kossaifi, J., Gramfort, A., Thirion, B., Varoquaux, G., 2014. Machine learning for neuroimaging with scikit-learn. Frontiers in Neuroinformatics 8, 14.

Allen, E.A., Damaraju, E., Plis, S.M., Erhardt, E.B., Eichele, T., Calhoun, V.D., 2014. Tracking whole-brain connectivity dynamics in the resting state. Cerebral Cortex 24, 663–676.

Beckmann, C.F., Smith, S.M., 2004. Probabilistic independent component analysis for functional magnetic resonance imaging. IEEE Transactions on Medical Imaging 23, 137–152.

Berlinet, A., Thomas-Agnan, C., 2011. Reproducing kernel Hilbert spaces in probability and statistics. Springer Science & Business Media.

Borgwardt, K.M., Gretton, A., Rasch, M.J., Kriegel, H.P., Schölkopf, B., Smola, A.J., 2006. Integrating structured biological data by kernel maximum mean discrepancy. Bioinformatics 22, e49–e57.

Brett, M., Hanke, M., Ct, M.A., Markiewicz, C., Ghosh, S., Wassermann, D., Gerhard, S., Larson, E., Lee, G.R., Halchenko, Y., Kastman, E., M, C., Morency, F.C., moloney, Rokem, A., Cottaar, M., Millman, J., jaeilepp, Gramfort, A., Vincent, R.D., McCarthy, P., van den Bosch, J.J., Subramaniam, K., Nichols, N., embaker, markhymers, chaselgrove, Basile, Oosterhof, N.N., Nimmo-Smith, I., 2017. nipy/nibabel: 2.2.0. URL: `https://doi.org/10.5281/zenodo.1011207`, doi:`10.5281/zenodo.1011207`.

Bullmore, E., Brammer, M., Rabe-Hesketh, S., Curtis, V., Morris, R., Williams, S., Sharma, T., McGuire, P., 1999. Methods for diagnosis and treatment of stimulus-correlated motion in generic brain activation studies using fMRI. Human Brain Mapping 7, 38–48.

Calhoun, V.D., Adali, T., Pearlson, G.D., Pekar, J., 2001. A method for making group inferences from functional mri data using independent component analysis. Human brain mapping 14, 140–151.

Chen, P.H.C., Chen, J., Yeshurun, Y., Hasson, U., Haxby, J., Ramadge, P.J., 2015. A reduced-dimension fMRI shared response model, in: Advances in Neural Information Processing Systems 28, pp. 460–468.

Cheng, B., Liu, M., Shen, D., Li, Z., Zhang, D., Initiative, A.D.N., et al., 2017. Multi-domain transfer learning for early diagnosis of alzheimers disease. Neuroinformatics , 1–18.

Cheng, B., Liu, M., Suk, H.I., Shen, D., Zhang, D., Initiative, A.D.N., et al., 2015a. Multimodal manifold-regularized transfer learning for mci conversion prediction. Brain Imaging and Behavior 9, 913–926.

Cheng, B., Liu, M., Zhang, D., Munsell, B.C., Shen, D., 2015b. Domain transfer learning for mci conversion prediction. IEEE Transactions on Biomedical Engineering 62, 1805–1817.

Cheng, B., Zhang, D., Shen, D., 2012. Domain transfer learning for MCI conversion prediction, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2012, Springer. pp. 82–90.

Friston, K.J., Fletcher, P., Josephs, O., Holmes, A., Rugg, M., Turner, R., 1998. Event-related fMRI: characterizing differential responses. NeuroImage 7, 30–40.

Friston, K.J., Holmes, A.P., Worsley, K.J., Poline, J.P., Frith, C.D., Frackowiak, R.S., 1994. Statistical parametric maps in functional imaging: a general linear approach. Human Brain Mapping 2, 189–210.

Fukunaga, K., 1990. Introduction to statistical pattern recognition. Academic Press.

Ghafoorian, M., Mehrtash, A., Kapur, T., Karssemeijer, N., Marchiori, E., Pesteie, M., Guttmann, C.R., de Leeuw, F.E., Tempany, C.M., van Ginneken, B., et al., 2017. Transfer learning for domain adaptation in MRI: Application in brain lesion segmentation, in: Medical Image Computing and Computer-Assisted Intervention – MICCAI 2017, Springer. pp. 516–524.

Gonzalez-Castillo, J., Hoy, C.W., Handwerker, D.A., Robinson, M.E., Buchanan, L.C., Saad, Z.S., Bandettini, P.A., 2015. Tracking ongoing cognition in individuals using brief, whole-brain functional connectivity patterns. Proceedings of the National Academy of Sciences 112, 8762–8767.

Gorgolewski, K., Esteban, O., Schaefer, G., Wandell, B., Poldrack, R., 2017. Openneuro: a free online platform for sharing and analysis of neuroimaging

data, in: Organization for Human Brain Mapping. Vancouver, Canada, p. 1677.

Haynes, J.D., Rees, G., 2006. Neuroimaging: decoding mental states from brain activity in humans. Nature Reviews Neuroscience 7, 523.

Hyvärinen, A., Oja, E., 2000. Independent component analysis: algorithms and applications. Neural Networks 13, 411–430.

Jenkinson, M., Beckmann, C.F., Behrens, T.E., Woolrich, M.W., Smith, S.M., 2012. FSL. NeuroImage 62, 782–790.

Jiang, W., Zavesky, E., Chang, S.F., Loui, A., 2008. Cross-domain learning methods for high-level visual concept classification, in: Proceedings of 15th IEEE International Conference on Image Processing, ICIP., IEEE. pp. 161–164.

Kriegeskorte, N., Goebel, R., Bandettini, P., 2006. Information-based functional brain mapping. Proceedings of the National Academy of Sciences 103, 3863–3868.

Marden, J.I., 2014. Analyzing and modeling rank data. Chapman and Hall/CRC.

Ogawa, S., Lee, T.M., Kay, A.R., Tank, D.W., 1990. Brain magnetic resonance imaging with contrast dependent on blood oxygenation. Proceedings of the National Academy of Sciences 87, 9868–9872.

Pan, S.J., Tsang, I.W., Kwok, J.T., Yang, Q., 2011. Domain adaptation via transfer component analysis. IEEE Transactions on Neural Networks 22, 199–210.

Pan, S.J., Yang, Q., 2010. A survey on transfer learning. IEEE Transactions on Knowledge and Data Engineering 22, 1345–1359.

Patel, V.M., Gopalan, R., Li, R., Chellappa, R., 2015. Visual domain adaptation: A survey of recent advances. IEEE Signal Processing Magazine 32, 53–69.

Pedregosa, F., Varoquaux, G., Gramfort, A., Michel, V., Thirion, B., Grisel, O., Blondel, M., Prettenhofer, P., Weiss, R., Dubourg, V., Vanderplas, J., Passos, A., Cournapeau, D., Brucher, M., Perrot, M., Duchesnay, E., 2011. Scikit-learn: Machine learning in Python. Journal of Machine Learning Research 12, 2825–2830.

Poldrack, R., Kittur, A., Kalar, D., Miller, E., Seppa, C., Gil, Y., Parker, D., Sabb, F., Bilder, R., 2011. The cognitive atlas: Toward a knowledge foundation for cognitive neuroscience. Frontiers in Neuroinformatics 5, 17.

Poldrack, R.A., 2007. Region of interest analysis for fMRI. Social Cognitive and Affective Neuroscience 2, 67–70.

Poldrack, R.A., Barch, D.M., Mitchell, J., Wager, T., Wagner, A.D., Devlin, J.T., Cumba, C., Koyejo, O., Milham, M., 2013. Toward open sharing of task-based fMRI data: the OpenfMRI project. Frontiers in Neuroinformatics 7, 12.

Rao, N., Cox, C., Nowak, R., Rogers, T.T., 2013. Sparse overlapping sets lasso for multitask learning and its application to fMRI analysis, in: Advances in Neural Information Processing Systems 26, pp. 2202–2210.

Van Essen, D.C., Ugurbil, K., Auerbach, E., Barch, D., Behrens, T., Bucholz, R., Chang, A., Chen, L., Corbetta, M., Curtiss, S.W., et al., 2012. The human connectome project: a data acquisition perspective. NeuroImage 62, 2222–2231.

Varoquaux, G., Gramfort, A., Pedregosa, F., Michel, V., Thirion, B., 2011. Multi-subject dictionary learning to segment an atlas of brain spontaneous activity, in: Biennial International Conference on Information Processing in Medical Imaging, Springer. pp. 562–573.

Vu, A.T., Auerbach, E., Lenglet, C., Moeller, S., Sotiropoulos, S.N., Jbabdi, S., Andersson, J., Yacoub, E., Ugurbil, K., 2015. High resolution whole brain diffusion imaging at 7 t for the human connectome project. NeuroImage 122, 318–331.

Wachinger, C., Reuter, M., Initiative, A.D.N., et al., 2016. Domain adaptation for alzheimer's disease diagnostics. NeuroImage 139, 470–479.

Weiss, K., Khoshgoftaar, T.M., Wang, D., 2016. A survey of transfer learning. Journal of Big Data 3, 9.

Yuan, M., Lin, Y., 2006. Model selection and estimation in regression with grouped variables. Journal of the Royal Statistical Society: Series B (Statistical Methodology) 68, 49–67.

Zhang, H., Chen, P.H., Ramadge, P., 2018. Transfer learning on fMRI datasets, in: Proceedings of 21st International Conference on Artificial Intelligence and Statistics (AISTATS), pp. 595–603.