

Learning Methods for Thought Recognition

Mark Palatucci

October 2009

Robotics Institute
Carnegie Mellon University
Pittsburgh, Pennsylvania 15213

Thesis Committee:
Tom Mitchell, Chair
Dean Pomerleau
J. Andrew Bagnell
Andrew Ng, Stanford

Proposal for Doctoral Thesis

*Submitted in partial fulfillment of the requirements
for the degree of Doctor of Philosophy in Robotics*

© 2009 Mark Palatucci

Abstract

This thesis proposal considers the problem of training machine learning classifiers in domains where data are very high dimensional and training examples are extremely limited or impossible to collect for all classes of interest. As a case study, we focus on the application of *thought recognition*, where the objective is to classify a person's cognitive state from a recorded image of that person's neural activity. Machine learning and pattern recognition methods have already made a large impact on this field, but most prior work has focused on classification studies with small numbers of classes and moderate amounts of training data. In this thesis, we focus on thought recognition in a limited data setting, where there are few, if any, training examples for the classes we wish to discriminate, and the number of possible classes can be in the thousands.

Despite these constraints, *this thesis seeks to demonstrate that it is possible to classify noisy, high dimensional data with extremely few training examples by using spatial and temporal domain knowledge, intelligent feature selection, semantic side information, and large quantities of unlabeled data from related tasks.*

In our preliminary work, we showed that it possible that build a binary classifier that can accurately classify between cognitive states with more than 80,000 features, and only two training examples per class. We also showed how classification can be improved using principled feature selection, and derived a significance test using order statistics that is appropriate for very high-dimensional problems with small numbers of training examples.

We have also explored the most extreme case of limited data, the *zero-shot learning setting*, where we do not have any training examples for classes we wish to discriminate. We showed that by using a knowledge base of semantic side information to create intermediate features, we can build a classifier that can classify words that people are thinking about, even without training data for those words while the classifier is forced to choose between nearly 1,000 different candidate words.

Finally, we showed how *multi-task learning* can be used to learn useful semantic features directly from data. We formulated the semantic feature learning problem as a *Multi-task Lasso* and presented an extremely fast and highly scalable algorithm for solving the resulting optimization.

We propose work to extend our zero-shot learning setting by optimizing semantic feature sets and by using an *active learning* framework to choose the most informative training examples. We also propose to use latent feature models such as components analysis and sparse coding in a *self-taught learning* framework to improve decoding by leveraging data from additional neural imaging experiments.

Contents

1	Introduction	4
2	Thesis Contributions	5
3	Related Work	6
3.1	Thought Recognition and Cognitive State Classification	6
3.2	Zero-Shot and Active Learning	7
3.3	Feature Selection, Sparsity, and Multi-task Learning	8
3.4	Self-Taught Learning and Latent Variable Models	8
4	Preliminary Work	9
4.1	Classification and Feature Selection for Problems with High Dimensionality and Few Training Examples	9
4.2	Zero-Shot Learning with Semantic Output Codes	11
4.3	Selecting Semantic Features with the Multi-task Lasso	13
5	Proposed Work	14
5.1	Zero-Shot Learning	14
5.2	Self-Taught Learning and Latent Factor Models	15
5.3	Application: Word Decoding	18
6	Schedule of Work	20

1 Introduction

In the last few years, there has been much work applying machine learning and pattern recognition methods to the problem of *thought recognition*. The objective is to classify a person’s cognitive state from a recorded image of that person’s neural activity [25] [54]. Often the neural images are taken with *functional magnetic resonance imaging* (fMRI) but other brain scanners such as magnetoencephalography (MEG) and electroencephalography (EEG) have been used as well [4]. fMRI remains the most popular method for cognitive state classification, however, due to its high spatial resolution that can measure neural activity in regions of the brain only a few millimeters wide.

Cognitive state classification has many applications, and in the last five years a large and growing body of research has emerged across many fields, ranging from psychology and neuroscience to machine learning and statistics. Much of the work is concerned with studying brain function. For example, a researcher might want to determine from a brain scan if a person is viewing a picture or listening to an audio clip. Or a more difficult task might be to discriminate between a picture of a house or a tool. If a machine learning classifier can be trained to discriminate between cognitive states, then the classifier can be analyzed to yield insights into how the brain reacts to different stimuli.

Similarly, cognitive state classifiers are also used to study pathology of the brain. For example, classifiers have been trained to diagnose schizophrenia and Alzheimer’s diseases [10][24]. Interpretation of the learned classifiers has given insight into how various neural disorders affect the brain.

Another application of cognitive state classification is *brain-computer-interfaces* (BCI) [80] [69] [36]. The goal of BCI is to use the signals recorded by a brain scanner for control and communication. This application has huge potential benefits for disabled persons and could greatly enhance quality of life. For example, a paralyzed person might be able to control a wheelchair just by thinking of a desired direction of movement [13].

Cognitive state classification is also of interest to quantitative machine learning and statistics researchers because the domain pushes the limits of modern classification methods. The datasets produced by brain scanners such as fMRI and MEG are incredibly high dimensional, often with hundreds of thousands or even millions of raw features. The data are also noisy, meaning that if the same experiment is repeated, a different brain image is obtained (sometimes dramatically different). This “noise” is caused by both the physical sensing process as well as the contextual influences on the human research subject.

Typically, human subjects are kept in a fMRI scanner between 30-60 minutes to minimize discomfort as they must remain extremely still. Brain scanning also involves a financial cost, as both the scanner and subject’s time must be paid for. In addition, since fMRI measures the slow hemodynamic (blood) response, a sample is typically collected only once per 15-20 seconds. As a result, the number of collected samples for a given task is usually very small (i.e. typically less than 100 samples). Thus, the main quantitative problem that must be addressed is:

How can classification be performed accurately when data are very high dimensional, noisy,

and very few training samples are available?

This thesis will address this question by building machine learning methodology that *maximizes classification accuracy and the number of cognitive states that can be classified, while minimizing the amount of training data required.*

2 Thesis Contributions

The central thesis of this work is that it is possible to decode noisy, high dimensional data with extremely few training examples by using spatial and temporal domain knowledge, intelligent feature selection, semantic side information, and large quantities of unlabeled data from related tasks.

This thesis seeks to make contributions to both core machine learning methodology, as well as applied neuro-imaging and brain-computer-interfaces. The following summarizes the contributions to date as well as the remaining contributions that will result after the thesis is completed:

1. **A classification methodology for problems with large numbers of features and small numbers of training examples** - We will show how to leverage spatial and temporal correlations in fMRI data to build a classifier that can accurately discriminate cognitive states with only a very small number of training examples per class, even with thousands of raw features. These results have already been obtained in our preliminary work.
2. **A feature selection methodology for high-dimensional problems** - We will show a significance test especially suited to very high dimensional problems with few training examples. We will use this test to choose the most relevant features during a discriminative feature selection and show how this improves classifier performance and helps model interpretation. These results have already been obtained in our preliminary work.
3. **A general framework for zero-shot learning with semantic side information** - We will develop a framework that allows a classifier to discriminate classes that did not appear in a training set. The framework uses *semantic output codes*, which is a knowledge base of side information that contains features (i.e. attributes) of the classes. By learning an intermediate layer of semantic features rather than the class labels directly, a classifier can extrapolate and discriminate novel classes that were omitted from a training set. This also enables accurate decoding of a much larger number of classes. Some preliminary results have been obtained, and additional work regarding optimization of codes and active learning is proposed.
4. **A methodology for multi-task feature learning useful for discovering semantic features** - We will show that useful semantic features can be learned directly from data using the multi-task Lasso. We present a blockwise coordinate descent algorithm to

solve the resulting optimization in a highly scalable manner. These results have already been obtained in our preliminary work.

5. **Exploration and discovery of semantic features useful for prediction and decoding of neural activity** - We will explore different sets of intermediate semantic features based on large corpus statistics as well as feature norming studies from human labeling. We will use sparse regression methods such as the Multi-task Lasso to discover which features are most useful for predicting different regions of neural activity the brain. This work is ongoing. Some results have been achieved to date, but we continue to evaluate new semantic feature sets.
6. **A methodology to leverage data sets from other imaging studies and human subjects** - We will show that latent variable models such as sparse coding and component analysis can be used to create useful features for decoding. This allows us to improve performance by leveraging unlabeled data from other imaging studies as well as data from other human subjects. This is proposed work.
7. **Word decoding application** - We show that it is often possible to classify a specific word that a person is thinking about, even without any training examples for that word, from a large set of nearly 1,000 possible words. We have achieved some preliminary results using fMRI and we will explore this application in other modalities such as MEG and EEG.

3 Related Work

3.1 Thought Recognition and Cognitive State Classification

Machine learning classifiers have made a large impact on the field of cognitive neuroscience by showing that a person’s cognitive or “conscious” state can be discriminated from an image of that person’s neural activity. Much of the early work in this area analyzed patterns of neural activity recorded using fMRI while human subjects perceived different objects [15] [45] [44]. Later work showed that other cognitive states such as political affiliation [28], drug addiction [83], and even truthfulness [17] could be classified. Excellent overviews of this line of research are available in Haynes and Rees [25] and Norman et al. [54].

Classification techniques have also been applied to the study of disease pathology. Studies have shown that it’s possible to discriminate between schizophrenics and healthy controls [10] [74]. Other work has investigated Alzheimer’s disease [24] as well as stroke recovery [71].

There has also been a large body of related work in the field of non-invasive brain computer interfaces (BCI), where the goal is to use neural signals for control and communication. Most of this work has focused on electroencephalography (EEG) because of its more practical form factor. Overviews of this field are available in [80] [69] [36]. Studies have also investigated BCI tasks in other neural scanners such as magnetoencephalography (MEG) [4] and even fMRI [82].

More recently, there has been work to build *generative models* of neural activity. Rather than focus on purely discriminative (i.e classification) tasks, this work tries to build models that can predict patterns of neural activity in response to novel stimuli. The work of Kay et al. [29] demonstrates a model that can predict neural activity in response to novel visual scenes, while the work of Mitchell et al. [48] shows that it is possible to predict neural activity for concrete nouns in English.

Much of the related work to date has focused on problems with moderate amounts of training data (e.g. forty examples per class) and small numbers of classes (e.g. binary). By contrast, this thesis focuses on a more difficult learning scenario: when there are potentially thousands of classes to discriminate and there are extremely few, and in some cases no training examples, for each class. This is an important problem setting and especially relevant to neural imaging, largely due to the difficulty of collecting large amounts of labeled training data from neural scanners.

3.2 Zero-Shot and Active Learning

The goal of zero-shot learning is to learn a classifier $f : X \rightarrow Y$ that must predict novel values of Y that were omitted from a training set. This problem has received little attention in the machine learning community. Some work by [31] on *zero-data learning* has shown the ability to predict novel classes of digits that were omitted from a training set. Some related theoretical work has been performed by [6].

In computer vision, techniques for sharing features across object classes have been investigated [77] [2] but relatively little work has focused on recognizing entirely novel classes, with the exception of [30] predicting visual properties of new objects and [19] using visual property predictions for object recognition.

Zero-shot learning is an important problem setting for neural decoding, computer vision as well as any domain where it may be impossible to collect training data for all classes that a classifier must discriminate. In this thesis, we develop this formalism and present a zero-shot learning algorithm called the *semantic output code classifier*. We also prove some of the first theoretical guarantees about when novel classes can be recovered without seeing them during training.

We also note that zero-shot learning is also closely related to the topic of *active-learning* because both deal with learning settings where limited training data are available. In zero-shot learning, only a small number of labeled classes are available in the training data (fewer than we wish to discriminate). Thus a natural question is, *if we would like to discriminate a large number of classes with only a small subset of these classes available in the training data, what is the optimal subset of classes to acquire?* This is a question of active-learning, which is concerned with predictive models that can choose what training data they want to be trained on [72] [73]. This thesis will address the important connection between zero-shot and active learning.

3.3 Feature Selection, Sparsity, and Multi-task Learning

The problem of selecting useful features for prediction is central to machine learning research. Feature selection is one of the oldest topics of research in this field, and a very comprehensive survey is available in [22]. One increasingly popular way to select features is to use regularized methods that induce sparsity, meaning that only a small number of available features are used to build a predictive model. This removes the burden of preselecting the most appropriate features, and allows the model to automatically ignore irrelevant features and choose the features most useful for the prediction task. The most popular sparse, linear model is the Lasso [76]. Other models that achieve sparsity through regularization include additive non-linear models [66] as well as those that perform structure learning of graphs [20].

Sparsity constraints have also been used in multi-task learning problems. In a multi-task learning setting, a learner is required to predict several tasks simultaneously, and large body of work has shown that learning *related* tasks together can improve performance over single-task learning, especially in situations with limited training data [11] [75]. In a similar fashion to single-task learning, sparse methods have been used to select predictive features across multi-tasks simultaneously. This question has been addressed in several fields. In machine learning the problem is known as multi-task feature selection [1], in statistics as the simultaneous or multi-task Lasso [79], and in signal processing as simultaneous sparse approximation [78].

In this thesis, we show how multi-task feature selection can be used to select features that 1) are useful for thought recognition tasks 2) lead to interpretable models. Unlike previous work that scales to only small numbers of features and tasks, we present a multi-task feature learning algorithm that uses a blockwise coordinate descent method that scales to both thousands of features and tasks.

3.4 Self-Taught Learning and Latent Variable Models

As mentioned earlier, one of the central problems of machine learning is selecting features from data that are useful for a predictive task. Often, the problem of feature selection is the most difficult part of building a prediction algorithm, and much research in fields from natural language processing to computer vision is dedicated to selecting features or constructing novel features from raw streams of data (e.g. text corpora or video streams). As a result, one common criticism of machine learning methods is that algorithms are useless without humans investing large amounts of time constructing useful features.

While this criticism is valid for some domains, it ignores recent progress in automated feature construction and *self-taught learning* [62]. Self-taught learning is an emerging body of research that deals with algorithms that can construct useful, novel features automatically from data (rather than just selecting the most relevant features like the previously described Lasso). Often learning is performed in a two-stage process, with the first stage dedicated to constructing novel features in an unsupervised manner, and the second stage dedicated to the supervised task of learning the relationship between the novel features and some output variable. These methods work particularly well when a large pool of data is available, and results have shown that useful features can be constructed even when data in the pool is from

tasks that are different from the current learning task. Thus, self-taught learning is one way to transfer knowledge from previously learned tasks to a new task [33] [35].

A similar idea is that of *deep-belief networks* which works by stacking models such as the Restricted Boltzmann Machine (RBM) to construct a hierarchy of novel features from a large pool of data [26][3]. This is related to components analysis, which tries to find lower-dimensional representations of data [5]. The common theme amongst all these methods is that useful features can be automatically constructed by learning latent, generative models of the data, where the latent factors become the novel features. A key benefit is that tasks that are weakly statistically related in the raw input space may share an underlying set of latent features. Thus, data from related tasks can be used to improve performance of a given learning task, especially when data is limited for the current task. One example of such a benefit applied to the fMRI domain was given in [68], which shows how *canonical correlation analysis* (CCA) can be used to combine data from multiple participants a fMRI study to improve a neural decoding task.

We believe that latent variable models will provide many benefits to performing classification in very high dimensional datasets with little training data. In this thesis, we will use latent variable models to discover hidden structure in neural imaging data, and will show how this structure can be used to 1) automatically construct useful features for neural classification tasks 2) ameliorate the limited data problem by sharing data through latent components across subjects, studies, and imaging modalities.

4 Preliminary Work

4.1 Classification and Feature Selection for Problems with High Dimensionality and Few Training Examples

Performing cognitive state classification using fMRI data is a challenging task due to the noisy, high-dimensional nature of the recorded neural images, and a typical dirt of labeled training examples. Often, problems may include hundreds of thousands of features, and less than ten labeled training examples per class. To deal with these challenges, we designed a classifier that utilizes spatial and temporal domain knowledge to improve parameter estimation and a feature selection method that automatically adapts to the dimensions of the problem.

4.1.1 Leveraging Spatial and Temporal Domain Knowledge

Most cognitive state classification problems exhibit sparsity, meaning that only a small number of the thousands of features are relevant to the task. For example, to discriminate when a person is thinking about a face vs. another object, a classifier might focus on only the features that are located in the fusiform gyrus region of the brain (activation in this region is correlated with observation of faces), while ignoring the remaining features located elsewhere, since the large majority of the features may not provide any useful signal that the classifier can use to discriminate between the two states.

Therefore, to perform accurate discrimination, the classifier must be *attribute efficient*, meaning that it will attempt to ignore features that are irrelevant to the task. One popular classifier that exhibits this property is Gaussian Naive Bayes (GNB). This classifier is popular for cognitive state discrimination tasks because it is quickly trained even with thousands of features, it is robust to noisy, irrelevant features, and being a parametric model, it is able to learn with only a small number of training examples. Often thirty to forty training examples may be sufficient to build an accurate classifier.

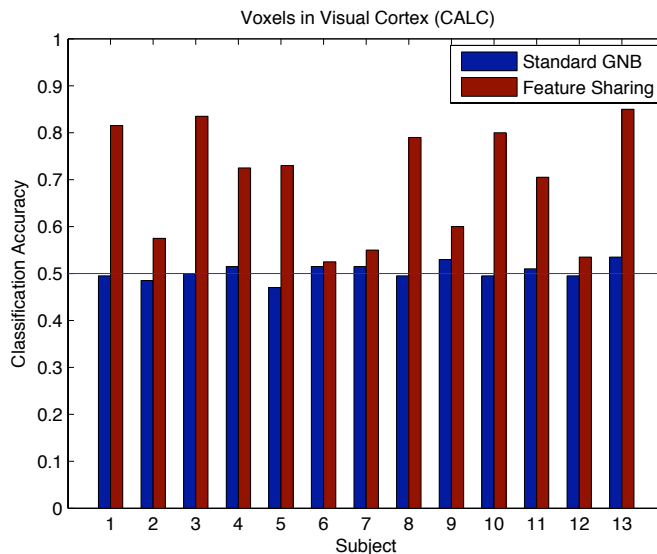


Figure 1: Accuracies of the standard Gaussian Naive Bayes classifier and the Feature Sharing classifier for 13 human subjects with two training examples per class. The classifier uses only voxels in the Visual Cortex (CALC).

Despite its popularity, the standard GNB model is insufficient when the number of training examples is extremely small (e.g. < 5 examples per class). However, we recognized that fMRI data exhibits a spatial and temporal smoothness, and the timeseries of adjacent features are often correlated. Using this domain knowledge, we constructed a classifier that uses variance pooling and hierarchical Bayesian estimators to learn parameters. This allows related features to be used as priors in the parameter learning. Using this technique, we were able to build a binary classifier that could accurately discriminate with only two training examples per class, even when the original data has over 80,000 features [58]. The normal GNB classifier fails completely on this task (See Figure 1).

4.1.2 Feature Selection Criteria

When training a classifier, the most relevant features are often chosen using *discriminative feature selection*. A classifier is trained individually for each feature and then evaluated on a validation set, where the best performing features are selected to train a final classifier. While the number of features can also be chosen using cross-validation, it is common practice to

avoid the computational burden and select an arbitrary number of features (e.g. 100) or all features that perform better than a specific accuracy (e.g. 50%).

Choosing an arbitrary accuracy, however, is particularly dangerous in classification problems with thousands of features and only a small number of examples in a validation set (e.g. < 50). With so many features, the risk that a random, noisy, feature may perform well (often with 80-90% accuracy) is quite high.

To address this issue, we developed a feature selection criterion [57] called the *multiplicity gap midpoint* that uses order statistics¹ to yields a significance threshold appropriate for the given problem. We found this method outperforms the *false-discovery-rate*, and is more intuitive than multiple-hypothesis testing because it avoids the *a priori* choice of a significance level. See Figure 2.

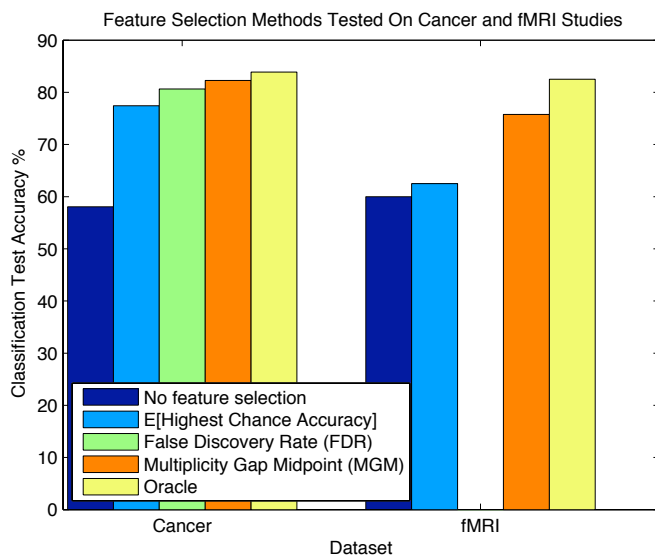


Figure 2: Accuracies for different feature selection methods for two classification tasks: Cancer (left) fMRI (right). The False Discovery Rate (FDR) method selected no features in the fMRI task.

4.2 Zero-Shot Learning with Semantic Output Codes

Machine learning classifiers are typically trained to approximate a target function $f : X \rightarrow Y$, given a set of labeled training data that includes all possible values for Y , and sometimes additional unlabeled training data. Little research has been performed on zero-shot learning, where the possible values for the class variable Y include values that have been omitted from the training examples. This is an important problem setting, especially in domains where Y can take on many values, and the cost of obtaining labeled examples for all values is high. One

¹Order statistics is concerned with the study of probability distributions over ranked or ordered variables. For example, order statistics can be used to define a probability distribution over the minimum of a collection of random variables.

obvious example is computer vision, where there are tens of thousands of objects which we might want a computer to recognize.

Another example is in thought recognition, where the goal is to determine the word or object a person is thinking about by observing an image of that person’s neural activity. It is intractable to collect neural training images for every possible word in English, so to build a practical neural decoder we must have a way to extrapolate to recognizing words beyond those in the training set.

To achieve this, we designed a *semantic output code* classifier (SOC) which utilizes a knowledge base of semantic properties of Y to extrapolate to novel classes [60]. Rather than predict the class labels directly, the classifier predicts properties or attributes of the objects. Formally,

Definition 1. *Semantic Output Code Classifier (SOC)*

A semantic output code classifier $\mathcal{H} : X^d \rightarrow Y$ maps points from some d dimensional raw-input space X^d to a label from a set Y such that \mathcal{H} is the composition of two other functions, \mathcal{S} and \mathcal{L} , such that:

$$\begin{aligned}\mathcal{H} &= \mathcal{L}(\mathcal{S}(\cdot)) \\ \mathcal{S} &: X^d \rightarrow F^p \\ \mathcal{L} &: F^p \rightarrow Y\end{aligned}$$

The semantic output code classifier first maps from a d dimensional raw-input space X^d into a semantic space of p dimensions F^p , and then maps this semantic encoding to a class label. For example, we may imagine some raw-input features from a digital image of a dog first mapped into the semantic encoding of a dog (i.e. a feature vector describing attributes of a dog), which is then mapped to the class label *dog*.

As part of its training input, this classifier is given a set of N examples \mathcal{D} that consists of pairs $\{x, y\}_{1:N}$ such that $x \in X^d$ and $y \in Y$. The classifier is also given a knowledge base \mathcal{K} of M examples that is a collection of pairs $\{f, y\}_{1:M}$ such that $f \in F^p$ and $y \in Y$. Typically, $M \gg N$, meaning that data in semantic space is available for many more class labels than in the raw-input space. Thus,

A semantic output code classifier is useful when the knowledge base \mathcal{K} covers more of the possible values for Y than are covered by the input data \mathcal{D} .

We demonstrated the semantic output code classifier on a thought recognition task using semantic knowledge bases derived from both human labeling and corpus statistics. We showed a SOC classifier can predict the word a person is thinking about from a recorded fMRI image of that person’s neural activity with accuracy much higher than chance, even when training examples for that particular word were omitted from the training set and the classifier was forced to pick the word from among nearly 1,000 alternatives [60]. See Table (1).

We also studied this formalism in a PAC framework. We proved the first formal guarantees that show conditions under which this classifier will predict novel classes. Specifically, given a distribution of semantic features, we can determine the number of examples necessary to predict a novel class with a given probability.

Table 1: The top five most likely words predicted for a held-out fMRI image collected for the word in bold (all fMRI images taken from participant P1). The number in the parentheses contains the rank of the correct word selected from 941 concrete nouns in English. Note that no training images for the held-out word appeared in the training set.

Bear	Foot	Screwdriver	Train	Truck	Celery	House	Pants
(1)	(1)	(1)	(1)	(2)	(5)	(6)	(21)
<i>bear</i>	<i>foot</i>	<i>screwdriver</i>	<i>train</i>	jeep	beet	supermarket	clothing
fox	feet	pin	jet	<i>truck</i>	artichoke	hotel	vest
wolf	ankle	nail	jail	minivan	grape	theater	t-shirt
yak	knee	wrench	factory	bus	cabbage	school	clothes
gorilla	face	dagger	bus	sedan	<i>celery</i>	factory	panties

4.3 Selecting Semantic Features with the Multi-task Lasso

In the previous section, we showed how semantic knowledge bases can be used to perform zero-shot learning. An obvious question then is how to choose an appropriate encoding of semantic features for classes we wish to predict. Previously, we experimented with semantic knowledge bases derived from large text corpora, as well as using human labeled features. Generally, we found that human labeled features performed better than the text based features.

The variation in performance with different sets of features leads naturally to the question: *what makes the best set of semantic features?* To address this question, we investigated using sparse learning methods such as the Lasso that could potentially select useful features directly from data. The hope was that we could pass a large set of potential semantic features directly into the model, and the model would automatically select which features were most useful.

To achieve this, we built a model to predict an image of neural activities from a large set of semantic features. One complication was that this image contained many small regions (i.e. voxels), and the neural activity had to be predicted for each one. Thus, the problem was that of *multi-task feature selection*, where each task was the prediction of neural activity at a specific region. The goal then was to select a single, common set of semantic features that were useful for predicting neural activity across all these regions in the brain.

We formulated this problem as a *Multi-task Lasso*. Although the formulation led to a convex optimization problem, a large technical challenge emerged because we wanted the algorithm to scale to thousands of features and tasks. To solve this, we developed a *blockwise coordinate descent* algorithm to solve the resulting optimization. This algorithm is now the fastest known method for solving the Multi-task Lasso.

We evaluated this method and found that we could select semantic features directly from data that outperformed the hand-crafted text corpora features previously described in Mitchell et al. [48]. We’re also using this method to identify useful sets of semantic properties for specific regions of the brain (e.g. visual cortex).

5 Proposed Work

5.1 Zero-Shot Learning

5.1.1 Optimizing Semantic Output Codes

In our previous work, we considered semantic feature sets that were derived from both text corpora and human labeling. Given a novel neural image, our model would predict the semantic features for this image, and then would look in a semantic knowledge base for words that had a semantic encoding close to this prediction. If the prediction was close to the word’s encoding in the knowledge base, it would predict the word correctly.

Despite some early successes with this approach, the method is sub-optimal because it ignores the error in the prediction of the semantic features. The model also does not consider the relationships between the semantic encodings in the given knowledge base and how those encodings relate to words that we would like to predict.

By leveraging knowledge of which semantic features can be predicted well, along with the distribution of word encodings in a knowledge base, it should be possible to design more optimal semantic codes that would maximize prediction accuracy in a zero-shot setting. For example, given a subset of classes that we would like to predict well, we could choose the semantic features that discriminate these classes most accurately.

For example, suppose we know that we’ll be decoding objects related to foods. Our goal would be to take our existing semantic encoding of the classes, and produce a new semantic encoding that would allow us to maximize the error in the prediction of the semantic features that discriminate foods (e.g. color) while ignoring features that do not discriminate (e.g. size). If we define a distribution over the classes we expect to see, as well as a distribution over the probability of predicting a specific feature well, then we could choose an encoding that maximizes the expected decoding accuracy of the given objects.

The simplest way to achieve this would be to define a distance metric in the second stage $\mathcal{L}(\cdot)$ of the semantic output code classifier. A Mahalanobis distance could be learned to ignore features that are predicted poorly, as well as those that do not discriminate the desired classes. Scale adjustments could also be made to account for different variances in the semantic features.

5.1.2 Optimizing Collection of Training Data with Active Learning

In a zero-shot learning setting, the learning algorithm does not have training examples for every class that it must predict. Instead, from a small set of training examples, the algorithm must learn to predict a set of semantic features that are common to all the classes. As a result, performance of a zero-shot learning algorithm depends heavily on the initial set of training examples.

If the learning algorithm could choose which examples appear in its training set, then the training set could be optimized so that the classifier could extrapolate to novel classes with

higher performance. For example, the classifier should choose training examples that utilize semantic features other than those of previous training examples, or it could choose examples to help it learn to predict existing features more accurately.

This is closely related to *active learning*, where learning algorithms can choose examples from a large pool of unlabeled data. We believe there is a large opportunity to apply similar techniques to optimize the choice of training data for a zero-shot learner. To develop an active learner for the zero-shot learning setting, three important questions must be addressed:

1. *Is the semantic code correct?* In other words, given enough data in the raw input space, could we learn to predict each of the semantic features perfectly, so that any class could be recovered? If not, then the active learning algorithm could be used to explore different semantic feature sets, as opposed to just optimizing which examples are best for a specific semantic code. These feature sets could be explored by using large text corpora (e.g. the web) or by human computation using a service like Mechanical Turk.
2. *What do we want to predict?* If there is a particular set of classes we care more about, then we should optimize data collection to maximize decoding accuracy of this set.
3. *How do we measure the contribution of an example to the prediction of a feature?* Each example may or not contribute to the improved prediction of a feature. We need to develop a way to measure the reduction of error for a given feature (e.g. reduction of variance).

As a simple experiment, we could score each possible example according to some measure. For example, we might select the word that has the highest combined inner product in semantic space with the all words we hope to predict. We could test the effectiveness of this measure by training on only a subset of our previously collected data. We would hope the intelligently selected examples would lead to high decoding performance compared with a randomly selected set of the same size.

We could also evaluate an active learning measure in a real-time, adaptive experimental paradigm. It should be possible to alter the collection of data by presenting different stimuli depending on how well the model predicts the data already collected. Recent improvements in the data acquisition software at the UPMC MEG Center make such an experiment possible.

5.2 Self-Taught Learning and Latent Factor Models

The goal of sparse coding is to find a small number of latent or hidden factors that can be used to explain some observed pattern of data. If this observable pattern has structure, it may be possible to represent (i.e. explain) the data in a lower dimensional space. This is similar to lossy data compression, but also may be used to recover structure from a highly noisy, observable signal.

Recent empirical results show that performance in many learning tasks can be improved by first learning latent factors using sparse codes, and then using this latent lower dimensional

representation of the data as features instead of the raw observable signal [34] [62].

This idea is similar to preprocessing data using various component analysis techniques like PCA, ICA, and CCA as well as methods like deep belief networks. In fact, all of these methods can be viewed as a type of latent variable model, and the differences depend on the prior assumptions about how the observable data relates to the latent factors, and also how the latent factors relate to each other. For example, PCA assumes factors are uncorrelated while ICA makes a stronger assumption that the factors are statistically independent.

Recent work in self-taught learning shows that latent factor models based on sparse coding can improve performance in data limited settings by first learning a lower-dimensional representation of data from other tasks *related* to the task of interest. This is useful when a large pool of data may be available for related tasks, but few data are available for a specific task. The assumption is that all these tasks share similar lower dimensional factors, and by casting a new task into this representation, it provides constraints that can be useful for learning, particularly when the original data is noisy, high dimensional, and few data are available.

This is exactly the scenario that we encounter in thought recognition problems. Given some of these early successes in self-taught learning, we believe that there is a rich opportunity to apply similar latent feature models to neural imaging.

5.2.1 Code Learning: Combining Data from Multiple Human Subjects and Studies

A neural imaging experiment is typically conducted on multiple human subjects. An open question is how to best combine data from multiple human subjects for a given neural decoding task. Clearly, brains exhibit some common patterns of activity across people in a given experiment, but there are often large differences in observed data. Is it possible to use data from other brains to improve decoding performance for a single human subject?

We propose to apply self-taught learning to the problem of combining data from multiple subjects. We believe that data from multiple subjects could be used to automatically learn a useful set of features in an unsupervised way. For example, sparse coding could be used to find a basis to reconstruct the neural activities across several different subjects. This basis would become a feature space for subsequent supervised learning tasks. Given a particular subject, the subject's raw fMRI data could be transformed into the basis that was learned for the other subjects. For a given neural decoding task, a classifier would be trained in a supervised fashion using the learned features, rather than the raw fMRI data of the given subject. The hope is that decoding from the learned basis would lead to improved accuracy compared with training from the raw fMRI features of the single subject (ignoring data from the other subjects).

Similarly, neural activity patterns often show spatial and temporal smoothness, regardless of the neural imaging experiment. Also, brain regions are also functionally connected, and strong correlations in activity often occur between different regions. Therefore, one might conjecture that all neural imaging experiments might share some common latent factors that govern how patterns are generated. As a result, the neural activity pattern in response to a stimulus for a given experiment might provide information that can be used for a completely different neural decoding task.

We believe that self-taught learning may be useful to improve performance in a neural decoding task by leveraging data from a completely different neural imaging experiment. In a similar fashion to the multiple subject experiment described earlier, we would like to learn a set of features using data from a different experimental paradigm (either for the same subject or another). We hope that using features learned from an entirely different paradigm could improve performance in a given supervised decoding task.

5.2.2 Single Trial Decoding

In a given neural imaging experiment, there is a large variation in the observed pattern of neural activity in response to a particular stimulus. As result, experiment paradigms usually collect data for the same stimulus multiple times, and combine these multiple trials to improve the signal-to-noise ratio of the observed data. It is common to see decreased performance in a neural decoding task if only a single trial is collected for a given stimulus.

As a result, an obvious goal is to improve decoding performance when only a single trial for a given stimulus is available. This is an important goal for brain-computer-interfaces, where the objective is to immediately decode a neural pattern in real-time without repeatedly collecting multiple trials.

We believe that a self-taught learning approach with latent variable models could be used to improve performance in single trial neural decoding by using the latent representation of the data to perform image de-noising.

5.2.3 Cross Modality Prediction

The three most popular neural scanning devices are fMRI, MEG, and EEG. These three devices each measure neural activity in a different way, and have different properties in terms of their spatial and temporal resolution as well as their form factor and cost. An open question is whether combing data from different modalities can improve performance in a neural decoding task. For example, could experimental data collected from a MEG scanner be used to improve performance in a subsequent EEG decoding task? Although each device relies on a different type of sensor, they are all measuring effects caused by the same physical process of neuron firing. As a result, a reasonable conjecture is that the information provided by one scanner may be used to augment data collected from another scanning modality, especially considering that scanning is a highly noisy sensing process, regardless of the modality.

We believe that latent variable models could be used to learn domain specific factors from large bodies of experimental data from each paradigm. As a result, an interesting opportunity exists to learn *higher-order latent factors*, which is a latent representation built by combining the latent representations from completely different sensory modalities. A higher order code may be appropriate when the statistical assumptions about the raw, observable data are different (e.g. one modality uses a Gaussian observable assumption while another uses a Poisson assumption). If the same assumption is shared between modalities, than a higher order code may not be necessary. In either case, this would enable the prediction of one modality from the

other, and provide a principled method for combining data from a variety of scanner modalities. Interpretation of the lower and higher order learned factors could yield many new insights into neuroscience.

5.3 Application: Word Decoding

5.3.1 Word Decoding as a Ranking Problem

In our previous work, we showed that it was possible to build a classifier to decode words that people were thinking about far above the chance level, even when the classifier had to choose from nearly 1,000 possible words. One way to measure performance of a word decoder beyond just simple classification accuracy is to use *rank accuracy*. Rank accuracy is useful when the classifier does not predict the correct class as the most likely. For example, a classifier that ranks the correct class as the fifth most likely out of 1,000 choices is preferable to one that ranks the correct word as the one-hundredth most likely.

Similarly, when a word is predicted incorrectly, we would like the words that ranked higher than the true word to be as similar as possible to the correct word. By similar we mean that a human would believe the two words are synonymous or closely related. For example, assume the classifier tried to predict the word *eagle*. We believe a classifier that incorrectly predicted the word *hawk* would be preferable to a classifier that predicted the word *airplane*, despite the fact that both hawks and airplanes fly. There are certain semantic features that are more important than others for determining similarity.

Our existing models ignore this fact and predict the semantic features of a given neural image by weighting each feature equally. The problem with this approach is that given a semantic knowledge base, two words that appear equally close to another word given some metric (e.g. such as \mathcal{L}_2), may not be ranked similarly close to the true word according to a human. Leveraging this idea, we believe that it would be possible to learn a ranking or distance metric so that incorrectly predicted words are closer to a human notion of similarity.

To achieve this, we propose an experiment where humans are given a word and then asked to pick the most similar word from a group of words. By repeating this process for many groups of words, we could discover which semantic features are most salient for a given set of words. This may provide interesting insights to the field of cognitive neuroscience, and could be used to learn a useful distance metric for word decoding.

5.3.2 Word Decoding with MEG and EEG

Our previous work focused exclusively on neural decoding tasks with data collected from fMRI scanners. An interesting question is whether the results can be replicated in other imaging modalities such as MEG and EEG. Both of these imaging modalities have substantially different spatial and temporal characteristics than fMRI. We have recently acquired data in these modalities using the same experimental paradigms that we used in our previous fMRI studies.

We propose to investigate word decoding in these modalities using our existing models as

well as the latent variable models we proposed in previous sections. We believe that the latent variable models, in particular, could be useful to assist the problem of feature engineering, which becomes much more difficult in the MEG and EEG domains due to the extremely high temporal resolution of the data.

6 Schedule of Work

A timeline of proposed work is given in Table 2. The timeline proposes fifteen months of work leading up to the thesis defense in Winter 2010.

Table 2: Timeline of proposed work

Task	Date of Completion
Classification and Feature Selection	
Classification with spatial and temporal domain knowledge	Completed
Significance measure for feature selection	Completed
Zero-shot Learning	
Basic formulation and theoretical analysis	Completed
Word decoding with fMRI	Completed
Learning of semantic features using Multi-task Lasso	Completed
Optimizing Zero-Shot Learning	
Optimized semantic codes	Fall 2009
Active learning metrics	Fall 2009
Self-Taught Learning and Latent Variable Models	
Combining data from multiple subjects and studies	Winter/Spring 2010
Single trial decoding	Winter/Spring 2010
Cross modality prediction	Spring/Summer 2010
Word Decoding Application	
Word decoding in MEG and EEG modalities	Fall 2009-Summer 2010
Learning rank/distance metrics	Fall 2010
Write thesis and defend	Winter 2010

References

- [1] A Argyriou, T Evgeniou, and M Pontil. Multi-task feature learning. *Advances in neural information processing systems*, Jan 2007.
- [2] E Bart and S Ullman. Cross-generalization: learning novel classes from a single example by feature replacement. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Jan 2005.
- [3] Y Bengio, P Lamblin, and D Popovici. Greedy layer-wise training of deep networks. *Advances in Neural Information Processing Systems 19*, Jan 2007.
- [4] N Birbaumer and LG Cohen. Brain-computer interfaces: communication and restoration of movement in paralysis. *The Journal of Physiology*, 579(3):621, 2007.
- [5] Chris Bishop. *Pattern Recognition and Machine Learning*. Springer, 2006.
- [6] J Blitzer, DP Foster, and SM Kakade. Zero-shot domain adaptation: A multi-view approach. *TTI-TR-2009-1*.
- [7] T Blumensath and M Davies. Sparse and shift-invariant representations of music. *IEEE Transactions on Audio*, Jan 2006.
- [8] D Bradley and J Bagnell. Differentiable sparse coding. *cs.cmu.edu*. Notes xxcxx.
- [9] D Bradley and J.A Bagnell. Convex coding. *UAI09*, pages 1–8, May 2009.
- [10] A Caprihan, G Pearlson, and V Calhoun. Application of principal component analysis to distinguish patients with schizophrenia from . . . *Neuroimage*, Jan 2008.
- [11] R Caruana. Multitask learning. *Machine Learning*, Jan 1997.
- [12] Y Cho and L Saul. Learning dictionaries of stable autoregressive models for audio scene analysis. *Proceedings of the 26th Annual International Conference on . . .*, Jan 2009.
- [13] Kyuwan Choi and Andrzej Cichocki. Control of a wheelchair by motor imagery in real time. pages 1–8, Sep 2008.
- [14] Paolo Ciaccia and Marco Patella. Pac nearest neighbor queries: Approximate and controlled search in high-dimensional and metric spaces. Jul 2001.
- [15] D Cox and R Savoy. Functional magnetic resonance imaging (fmri)“brain reading”: detecting and classifying . . . *Neuroimage*, Jan 2003.
- [16] Koby Crammer and Yoram Singer. On the learnability and design of output codes for multiclass problems. *Machine Learning*, Jun 2001.
- [17] C Davatzikos, K Ruparel, Y Fan, and D Shen. Classifying spatial patterns of brain activity with machine learning methods: application to lie . . . *Neuroimage*, Jan 2005.

- [18] Thomas G Dietterich and Ghulum Bakiri. Solving multiclass learning problems via error-correcting output codes. *JAIR*, Jul 1995.
- [19] A Farhadi, I Endres, D Hoiem, and D Forsyth. Describing objects by their attributes. *CVPR*, Jan 2009.
- [20] J Friedman, T Hastie, and R Tibshirani. Sparse inverse covariance estimation with the graphical lasso. *Biostatistics*, 9(3):432–441, Dec 2007.
- [21] R Grosse, R Raina, H Kwong, and AY Ng. Shift-invariant sparse coding for audio classification. *UAI07*, 9:8.
- [22] I Guyon and A Elisseeff. An introduction to variable and feature selection. *The Journal of Machine Learning Research*, Jan 2003.
- [23] D Hardoon, J Shawe-Taylor, and O Friman. Kcca for fmri analysis. *eprints.ecs.soton.ac.uk*, Jan 2004.
- [24] S Hayasaka, A Du, A Duarte, J Kornak, G Jahng, M Weiner, and N Schuff. A non-parametric approach for co-analysis of multi-modal brain imaging data: Application to alzheimer’s disease. *Neuroimage*, Jan 2006.
- [25] J Haynes and G Rees. Decoding mental states from brain activity in humans. *Nature Reviews Neuroscience*, Jan 2006.
- [26] G Hinton, S Osindero, and Y Teh. A fast learning algorithm for deep belief nets. *Neural Computation*, Jan 2006.
- [27] GE Hinton and RR Salakhutdinov. Reducing the dimensionality of data with neural networks. *science*, 313(5786):504–507, 2006.
- [28] J Kaplan, J Freedman, and M Iacoboni. Us versus them: Political attitudes and party affiliation influence neural response to faces of presidential candidates. *Neuropsychologia*, Jan 2007.
- [29] K Kay, T Naselaris, R Prenger, and J Gallant. Identifying natural images from human brain activity. *Nature*, Jan 2008.
- [30] C Lampert, H Nickisch, and S Harmeling. Learning to detect unseen object classes by between-class attribute transfer. *CVPR*, 2009.
- [31] H Larochelle, D Erhan, and Y Bengio. Zero-data learning of new tasks. *AAAI*, 2008.
- [32] H Lee, A Battle, R Raina, and A Ng. Efficient sparse coding algorithms. *Advances in neural information processing systems*, Jan 2007.
- [33] H Lee, C Ekanadham, and A Ng. Sparse deep belief net model for visual area v2. *Advances in neural information processing systems*, 20, 2008.

- [34] H Lee, R Raina, A Teichman, and A Ng. Exponential family sparse coding with applications to self-taught learning. *www-cs.stanford.edu*.
- [35] Honglak Lee, Roger Grosse, Rajesh Ranganath, and Andrew Ng. Convolutional deep belief networks for scalable unsupervised learning of hierarchical representations. *ICML '09: Proceedings of the 26th Annual International Conference on Machine Learning*, Jun 2009.
- [36] CT Lin, LW Ko, JC Chiou, JR Duann, RS Huang, SF Liang, TW Chiu, and TP Jung. Noninvasive neural prostheses using mobile and wireless eeg. *Proceedings of the IEEE*, 96(7):1167–1183, 2008.
- [37] H Liu, M Palatucci, and J Zhang. Blockwise coordinate descent procedures for the multi-task lasso, with applications to neural semantic basis discovery. *Proceedings of the 26th Annual International Conference on Machine Learning*, 2009.
- [38] R Lomasky, CE Brodley, M Aernecke, D Walt, and M Friedl. Active class selection. *Lecture Notes in Computer Science*, 4701:640, 2007.
- [39] J Mairal, F Bach, J Ponce, and G Sapiro. Online dictionary learning for sparse coding. *Proceedings of the 26th Annual International Conference on . . .*, Jan 2009.
- [40] J Mairal, F Bach, J Ponce, and G Sapiro. Supervised dictionary learning. *NIPS2008*, 2009.
- [41] J Mairal, G Sapiro, and M Elad. Learning multiscale sparse representations for image and video restoration. *SIAM Multiscale Modeling and Simulation*, Jan 2008.
- [42] B Marlin and K Murphy. Sparse gaussian graphical models with unknown block structure. *Proceedings of the 26th Annual International Conference on . . .*, Jan 2009.
- [43] T Mitchell. Computational models of neural representations in the human brain. *Proceedings of the 19th international conference on . . .*, Jan 2008.
- [44] T Mitchell, R Hutchinson, M Just, and R Niculescu. Classifying instantaneous cognitive states from fmri data. *AMIA... Annual Symposium proceedings [electronic resource]*, Jan 2003.
- [45] T Mitchell, R Hutchinson, R Niculescu, and F Pereira Learning to decode cognitive states from brain images. *Machine Learning*, Jan 2004.
- [46] T Mitchell and M Just. Using machine learning and cognitive modeling to understand the fmri measured brain activation *Proceedings of collaborative research computational . . .*, Jan 2005.
- [47] T Mitchell, S Shinkareva, A Carlson, and K Chang. Predicting human brain activity associated with noun meanings. *Citeseer*.

- [48] T Mitchell, S Shinkareva, A Carlson, and K Chang. Predicting human brain activity associated with the meanings of nouns. *science*, Jan 2008.
- [49] A Mnih and G Hinton. A scalable hierarchical distributed language model. *Neural Information Processing Systems*, 22, 2009.
- [50] H Mobahi, R Collobert, and J Weston. Deep learning from temporal coherence in video. *Proceedings of the 26th Annual International Conference on . . .*, Jan 2009.
- [51] R Niculescu. Exploiting parameter domain knowledge for learning in bayesian networks. *reports-archive.adm.cs.cmu.edu*, Jan 2005.
- [52] R Niculescu, T Mitchell, and R Rao. Exploiting parameter related domain knowledge for learning in graphical models. *Proceedings of the Fifth SIAM International Conference*, Jan 2005.
- [53] R Niculescu, T Mitchell, and R Rao. Bayesian network learning with parameter constraints. *The Journal of Machine Learning Research*, Jan 2006.
- [54] K Norman, S Polyn, G Detre, and J Haxby. Beyond mind-reading: multi-voxel pattern analysis of fmri data. *Trends in Cognitive Sciences*, Jan 2006.
- [55] Guillaume Obozinski, Ben Taskar, and Michael I Jordan. Joint covariate selection and joint subspace selection for multiple classification problems. *Stat Comput*, pages 1–22, Jan 2009.
- [56] M Palatucci. Temporal feature selection for fmri analysis. *Technical Notes*, 2006.
- [57] M Palatucci and A Carlson. On the chance accuracies of large collections of classifiers. *Proceedings of the 25th international conference on Machine Learning*, 2008.
- [58] M Palatucci and T Mitchell. Classification in very high dimensional problems with handfuls of examples. *Lecture Notes in Computer Science: ECML/PKDD*, Jan 2007.
- [59] M Palatucci, T Mitchell, and H Liu. Discovering a semantic basis of neural activity using simultaneous sparse approximation. *Sparse Optimization and Variable Selection Workshop, International Conference on Machine Learning*, Jan 2008.
- [60] M Palatucci, D Pomerleau, G Hinton, and T Mitchell. Zero-shot learning with semantic output codes. *Neural Information Processing Systems (NIPS)*, 2009.
- [61] T PhD, R Hutchinson, M PhD, and R Niculescu. Classifying instantaneous cognitive states from fmri data. *Proceedings of the 2003 American Medical Informatics . . .*, Jan 2003.
- [62] R Raina, A Battle, H Lee, B Packer, and AY Ng. Self-taught learning: Transfer learning from unlabeled data. *Proceedings of the 24th international conference on Machine learning*, pages 759–766, 2007.

- [63] R Raina, A Madhavan, and A Ng. Large-scale deep unsupervised learning using graphics processors. *Proceedings of the 26th Annual International Conference on . . .*, Jan 2009.
- [64] CP Marc’ Aurelio Ranzato, S Chopra, and Y LeCun. Efficient learning of sparse representations with an energy-based model. *B. Schölkopf, J. Platt et T. Hoffman, éditeurs, Advances in Neural Information Processing Systems*, 19.
- [65] F Marc’ Aurelio Ranzato, Y Boureau, and Y LeCun. Unsupervised learning of invariant feature hierarchies with applications to object recognition. *Proceedings of the International Conference on Computer . . .*, Jan 2007.
- [66] P Ravikumar, H Liu, J Lafferty, and L Wasserman. Spam: sparse additive models. *Advances in neural information processing systems*, Jan 2008.
- [67] I Rish, G Grabarnik, G Cecchi, F Pereira, and GJ Gordon. Closed-form supervised dimensionality reduction with generalized linear models. *Proceedings of the 25th international conference on Machine learning*, pages 832–839, 2008.
- [68] I Rustandi, Marcel Adam Just, and Tom M Mitchell. Integrating multiple-study multiple-sub ject fmri datasets using canonical correlation analysis. *MICCAI*, pages 1–8, Jun 2009.
- [69] P Sajda, KR Müller, and KV Shenoy. Brain-computer interfaces. *IEEE Signal Processing Magazine*, 2008.
- [70] Riitta Salmelin, Sylvain Baillet, Mia Liljeström, Annika Hultén, Lauri Parkkonen, and Riitta Salmelin. Comparing meg and fmri views to naming actions and objects. *Hum. Brain Mapp.*, 30(6):1845–1856, Jun 2009.
- [71] T Schmah, GE Hinton, RS Zemel, SL Small, and S Strother. Generative versus discriminative training of rbms for classification of fmri images. *NIPS*, 2008.
- [72] B Settles, M Craven, and S Ray. Multiple-instance active learning. *Advances in neural information processing systems*, 2007.
- [73] Burr Settles. Active learning literature survey. *Computer Science Technical Report*, (1648):1–46, Jan 2009.
- [74] L Skelly, V Calhoun, S Meda, and J Kim. Diffusion tensor imaging in schizophrenia: Relationship to symptoms. *Schizophrenia research*, Jan 2007.
- [75] S Thrun. Is learning the n-th thing any easier than learning the first? *Advances in neural information processing systems*, Jan 1996.
- [76] R Tibshirani. Regression shrinkage and selection via the lasso. *Journal of the Royal Statistical Society. Series B (. . .*, Jan 1996.
- [77] A Torralba, K Murphy, and W Freeman. Sharing visual features for multiclass and multi-view object detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Jan 2007.

-
- [78] JA Tropp. Algorithms for simultaneous sparse approximation. part ii: Convex relaxation. *Signal Processing*, 86(3):589–602, 2006.
- [79] Berwin A Turlach, William N Venables, and Stephen J Wright. Simultaneous variable selection. *Technometrics*, (27):349–363, May 2005.
- [80] E Vaadia and N Birbaumer. Grand challenges of brain computer interfaces in the years to come. *Frontiers in Neuroscience*.
- [81] X Wang, R Hutchinson, and T Mitchell. Training fmri classifiers to detect cognitive states across multiple human subjects. *Proceedings of the 2003 Conference on Neural Information . . .*, Jan 2003.
- [82] N Weiskopf, F Scharnowski, R Veit, R Goebel, N Birbaumer, and K Mathiak. Self-regulation of local brain activity using real-time functional magnetic resonance imaging (fmri). *Journal of Physiology-Paris*, 98(4-6):357–373, 2004.
- [83] L Zhang, D Samaras, D Tomasi, and N Alia-Klein. Exploiting temporal information in functional magnetic resonance imaging brain data. *Lecture Notes in Computer Science*, Jan 2005.