Using Machine Learning to Predict Human Brain Activity

Senior Thesis
Mahtiyar Bonakdarpour
Advisor: Tom Mitchell

1 Abstract

Brain imaging studies are geared towards decoding the way the human brain represents conceptual knowledge. It has been shown that different spatial patterns of neural activation correspond to thinking about different semantic categories of pictures and words. This research is aimed at developing a computational model that predicts functional magnetic resonance imaging (fMRI) neural activation associated with words. The current model has been trained with a combination of data from a text corpus and fMRI data associated with viewing several dozen concrete nouns. Once trained, the model predicts fMRI activation for other nouns in the text corpus with significant accuracy (for individual subjects).

In this thesis, we aim to assist in the development of a model which can accurately predict fMRI activation across subjects and studies. Through the failure of a naive solution to this problem, we explore both the differences in brain activation from study to study (in the same subject), and the accuracy of mapping brain coordinates to a common space. We also develop new methods of searching for informative and stable voxels. We compare models for the same subject across multiple studies, and multiple subjects in the same study thereby allowing us to understand the variability in brain activation from subject to subject, and study to study.

2 Introduction

In the last two decades there have been great advances in brain imaging technology. Since the early 20th centry, neuroscientists inferred a correlation between neural activity and blood oxygenation in the brain. Since neurons do not have an inherent source of energy, an active brain region requires the inflow of oxygen carried by hemoglobin from nearby capillaries. Thus, a region with increased neural activity requires an increase in blood flow (occurring approximately 1 to 5 seconds after the oxygen consumption). This *hemodynamic response* peaks for about 4 to 5 seconds before returning to equilibrium.

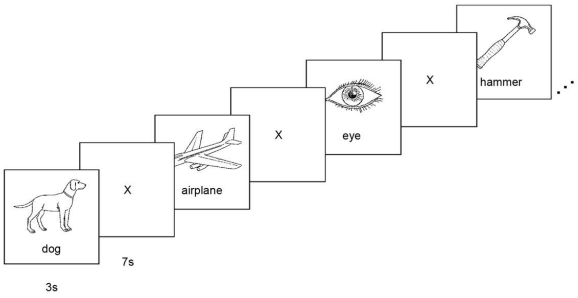
Most fMRI research depends on *blood-oxygen-level dependence* (BOLD); blood releases oxygen to active regions of the brain at a faster rate than to inactive regions – this disparity results in a

magnetic susceptibility between oxygenated regions and deoxygenated regions. fMR brain images provide researchers with a visual representation of brain activity. Data obtained through repeated experiments can be used in conjunction with statistical methods to determine which areas of the brain are related to certain thoughts (about food, animals, tools, etc.).

3 Experiment

Data from three experiments have been obtained. The difference between experiments lies only in the stimuli presented to the subjects. In the *60 word-picture* experiment, stimuli were line drawings and concrete noun labels (see Figure 1) of 60 concrete objects, from 12 semantic categories (e.g. tools and animals). The *60 word-only* experiment contained the same words as the *60 word-picture* experiment, without the line drawings. Finally, the *40 abstract-concrete* experiment contained 40 word-only stimuli consisting of both concrete and abstract (e.g. democracy, justice, etc.) nouns. Each word, in each experiment, was presented 6 times.

Figure 1: Stimuli



Prior to the experiment, participants were asked to create a list of properties for each word. When a stimulus was presented, the subject was asked to think about the previously-generated list of properties to promote consistency across presentations. There was no attempt to keep this list of properties consistent between subjects. Functional images were obtained for each presentation of each word. To account for the delay in the hemodynamic response, each image was the mean of the images collected at 4s, 5s, 6s, and 7s after stimulus onset. This thesis is mainly concerned with the analysis of the *60 word-picture* and *60 word-only* experiments. The *40 abstract-concrete* experiment is presented to illustrate the relationship between the stimulus words and the feature set selected in the model.

4 Per-Subject Model

4.1 Overview

In the interest of completeness, I will present the single-subject model before illustrating its extension to multiple subjects. The model operates under two crucial assumptions: (1) the neural basis

of semantic representation of *concrete* nouns is related to the statistical properties of those nouns in a text corpus (i.e. the way *celery* and *eat* co-occur in text gives us information about how they are represented in the brain) and (2) the brain activity for concrete nouns is a linear combination of contributions from each of its *semantic features* (here, the features are specifically chosen in advance as we will illustrate later). These two assumptions will become clear as the model is described. Using these two assumptions, the model uses a two-step approach in predicting the brain activity for a given noun.

To generate the fMR image for a noun, the model first maps the noun to a set of intermediate semantic features which are obtained from a Google text corpus. In the current model, these semantic features are statistical properties related to predefined verbs and the input noun. Each feature is defined as the number of co-occurences of that verb with the input noun in the text corpus (within 5 tokens of eachother). For example, one feature might be the frequency with which the input word, *celery*, co-occurs with the verb *taste*. The second step generates the fMR image as a weighted sum of brain images contributed by each of the semantic features. Specifically, the predicted activation A_v at voxel v in the brain image for word w is given by:

$$A_v = \sum_{i=1}^{n} c_{vi} f_i(w) \tag{1}$$

where $f_i(w)$ is the co-occurence value of the ith semantic feature with the input word w, n is the number of features (predefined for the model), and c_{vi} is a learned parameter that specifies the magnitude of activation the ith intermediate semantic feature contributes to voxel v. An advantage to using this model is the fact that, once trained, the model can be evaluated by giving it words outside of the training set and comparing the predicted image with the actual image.

4.2 Training

After defining the semantic features $f_i(w)$ for each feature i in the predefined set (i.e. computing the co-occurence scores), the parameters c_{vi}, which predict the neural signature contributed by the ith semantic feature to the vth voxel, are inferred from the data. These parameters are learned from a set of observed fMR images associated with known input words. The $N-2$ images (where N is the number of words used in the original study) chosen on each training iteration are normalized by subtracting the mean of all the images from each one. The algorithm is presented below:

1. A training stimulus (i.e. a concrete noun) w_t is mapped to a feature vector $< f_1(w_t), ..., f_n(w_t) >$ (recall that $f_1(w_t)$ is the co-occurence of feature 1 and word w_t in a corpus of data)

2. Use multiple regression to obtain MLEs of the c_{vi} values (i.e. the set of c_{vi} values that minimize the sum of squared errors between the predicted images and the actual images)

When the number of features exceeds the number of training examples, a regularization term is inserted into the solution to penalize error. Using the feature vector and these learned weights the model can now be used to predict words outside of the training set. The model will have a distinct feature vector for each word, and will predict an image for any word with a statistic in the text corpus. These feature vectors can be compared by exploring the differences in their predicted images with the actual images.

4.3 Evaluation

Different semantic features can be compared by training a model for each of them, calculating prediction accuracies for each, and comparing those accuracies. The model is evaluated using a leave-two-out cross validation approach. Given N stimulus items, the model is repeatedly trained using every possible $N-2$ stimulus subset. For each iteration, the trained model was tested by having it predict the brain images for the 2 left out words. Then, requiring it to *match* the two predicted images with the two actual images for the left out words. This is iterated $\binom{N}{2}$ times, and an accuracy score is calculated based on how often the model matches the predicted with the actual correctly.

4.3.1 Matching

The model matches the predicted images with the actual images using a comparison test. A comparison metric is computed for each (predicted image, actual image) pairing. The pairing with the highest similarity score is taken to be the predicted association. If the predicted image is matched with the correct actual image, then the model is said to have succeeded on that training iteration. The current metric used in the model is cosine similarity (also known as pearson's correlation coefficient) treating each FMR image as a vector. The similarity score is calculated on a subset of the voxels, since most are thought to be noisy and do not contain information about the stimulus.

4.3.2 Voxel Selection

On each training iteration, a subset of voxels are chosen to compare predicted and actual images (and also to infer the c_{vi} coefficients mentioned above). The theory underlying voxel selection is that those voxels which have consistent activity patterns for each presentation of a given word are those which contain the most information about the stimuli. The stability scores for each voxel are calculated by using data from the 6 presentations of the $N-2$ stimuli on each training iteration. Thus, each voxel is associated with a $(6 \times N-2)$ matrix, where entry (i, j) is the voxel activity on presentation i, word j. The stability score for each voxel is then calculated as the average pairwise correlation over all rows in the matrix. The validity of this voxel selection method is explored later in this thesis.

4.4 Per-Subject Model Results

The above model was applied to the three experiments referred to previously:

- 60 word-picture
- 60 word-only
- 40 abstract-concrete

The model was compared using two feature sets: 25 hand-selected verbs, and 486 commonly occuring verbs. It was evaluated on a each subject, and a mean accuracy score was calculated for each experiment, across all subjects.

4.4.1 25 Verbs

The 25 verbs used were chosen to reflect high-level sensor-motor activities. They include: see, hear, listen, taste, smell, eat, touch, rub, lift, manipulate, run, push, fill, move, ride, say fear, open, approach, near, enter, drive, wear, break and clean.

60 word-picture Results

Subject	Accuracy
03616B	0.8249
03839B	0.7644
03861B	0.7802
03921B	0.7237
03993B	0.7841
04008B	0.8542
04019B	0.7299
04124B	0.6785
04228B	0.8230

Average: .7736

60 word-only Results

Subject	Accuracy
04383B	0.7424
04480B	0.6017
04564B	0.5350
04605B	0.7565
04619B	0.6057
04647B	0.7186
04408B	0.6610
04550B	0.4780
04597B	0.7780
04617B	0.6825
04639B	0.7830

Average: .6680

40 abstract-concrete Results

Subject	Accuracy
05248B	0.5436
05300B	0.4731
05344B	0.5705
05344B	0.7565
05181B	0.6308
05230B	0.4705
05245B	0.3398
05258B	0.6013
05324B	0.5282
05236B	0.4910
05222B	0.6051
05176B	0.5090

Average: .5239

The model's chance accuracy is 50 percent, and a modified version of the permutation test shows us that significant accuracy is above 61 percent. Not surprisingly, this model performs poorly with the 40 abstract-concrete dataset. The 25 verbs were hand-selected to accurately describe concerete nouns (those related to sensory-motor activities). Abstract nouns, however, are not accuractly represented with these verbs and this is relfected in the accuracy score.

4.4.2 486 Verbs

The 486 verbs were selected from a list of commonly occuring verbs in a large text corpus.

60 word-picture Results

Subject	Accuracy
03616B	0.8723
03839B	0.7893
03861B	0.7859
03921B	0.7096
03993B	0.8497
04008B	0.7362
04019B	0.7949
04124B	0.7949
04228B	0.7966

Average: .7922

60 word-only Results

Subject	Accuracy
04383B	0.7333
04480B	0.6266
04564B	0.5441
04605B	0.7232
04619B	0.6040
04647B	0.7672
04408B	0.7243
04550B	0.6633
04597B	0.8079
04617B	0.5876
04639B	0.8537

Average: .6941

40 abstract-concrete Results

Subject	Accuracy
05258B	0.5962
05300B	0.3308
05344B	0.7308
05181B	0.7372
05230B	0.6115
05245B	0.6077
05258B	0.6731
05324B	0.6218
05256B	0.6731
05222B	0.7154
05176B	0.7371

Average: .5900

The 486 verbs improve the 40 abstract-concrete model by about 7 percent relative to the 25 verbs. This supports the hypothesis that the semantic features do indeed encode the meaning of the nouns. However, it seems that the first assumption in the model (that the neural basis of semantic representation is related to statistical properties of words in a text corpus) is better supported by the concrete nouns.

5 (Naively) Pooled Model

The current across-subject-across-experiment model is a basic extension of the per-subject model. Some steps have been altered to accomodate the ”pooled” data, and they are detailed below.

5.1 Preprocessing

Pooling data has only been tried with two experiments. Given experiments A and B with N and M stimulus words respectively, we treat the experiments individually until subtracting the mean of the $N+M$ images from all of the images. Then, since voxel i in experiment A does not correpond to voxel i in experiment B (due to the separate preprocessing of data), the two brain regions are mapped into a common space, and intersected, thereby keeping only the voxels the two experiments shared. It is important to note that the common space mapping is not found to be perfectly accurate - therefore, voxels are thought to be mapped into similar regions instead of exactly on top of one another. Since brain activation for a given word is usually found to be clustered, this mapping is thought to suffice. This interesection shrinks the image from around $20,000$ voxels to between $13,000$ and $19,000$. Finally, the data are merged and training proceeds in the following manner.

5.2 Training

Instead of training on $\binom{N+M}{2}$ training iterations as in the Per-Subject Model, we limit the training iterations to those which leave-two-out within the same experiment. We exclude training iterations when one word is left out in experiment A and the other is left out in experiment B. This was used intially to speed up computation. Preliminary analysis of allowing the left-out words to be across experiments was not found to yield a considerable difference in model accuracy.

5.3 Voxel Selection

We have attempted two different voxel selection methods in the pooled model.

5.3.1 Method 1

First, the voxels are intersected, leaving only those voxels which are present in both studies. After intersection, the stability score for voxel i (as calculated in 4.3.2) is averaged from the two studies. The top 500 voxels (after averaging) are then chosen to train and test the model.

5.3.2 Method 2

Since the training iterations are contrained to leaving two words out within the same study, we attempted a voxel selection method which is more compatible with the single-study results. When leaving two words out of experiment A, we obtain the 500 most stable voxels in experiment A and the corresponding voxels (after intersection), in experiment B. Note that the voxels chosen in experiment B may not be the most stable in that experiment (and generally are not).

5.4 Results

Accuracy scores have been found for two distinct sets of pooled models. The first set deals with pooling data from the *same* subject across different experiments. The second set deals with pooling

data from *different* subjects in the same experiment.

5.4.1 Same Subject Across Experiments

Data from subject **05236B**, in the *40 abstract-concrete* experiment, and subject **03921B**, in the *60 word-picture* experiment were pooled. This was the same subject, in two different experiments. The model was tested with 25 verbs. Here are the Per-Subject model results:

Per-Subject Results

Subject	Accuracy (25 verbs)	Accuracy (486 verbs)
05236B	0.4910	0.4154
03921B	0.7237	0.7027

The accuracy of the pooled model was **0.5529** using Method 1 and **.5467** using Method 2.

Data from subject **04384B**, in the *60 word-only* experiment, and subject **03616B**, in the *60 word-picture* experiment were pooled. The model was tested with 25 verbs. Here are the Per-Subject model results:

Per-Subject Results

Subject	Accuracy (25 verbs)	Accuracy (486 verbs)
04383B	0.7424	0.7333
03616B	0.8248	0.8733

The accuracy of the pooled model was **0.5715** using Method 1 and **.6071** using Method 2. This pooling was also tested with 486 verbs. The accuracy of the pooled model was **0.5850** using Method 1 with 486 verbs.

5.4.2 Same Experiment, Different Subjects

Data from subject **03616B**, in the *60 word-picture* experiment, and subject **04008B**, in the *60 word-picture* experiment were pooled. This is two different subjects in the same experiment. The model was tested with 25 verbs. Here are the Per-Subject model results:

Per-Subject Results

Subject	Accuracy (25 verbs)	Accuracy (486 verbs)
03616B	0.8248	0.8733
04008B	0.8542	0.7362

The accuracy of the pooled model was **0.5927** using Method 2.

Data from subject **04228B**, in the *60 word-picture* experiment, and subject **03993B**, in the *60 word-picture* experiment were pooled.

Per-Subject Results

Subject	Accuracy (25 verbs)	Accuracy (486 verbs)
04228B	0.8230	0.7966
03993B	0.7841	0.8497

The accuracy of the pooled model was **0.5696** using Method 2.

Data from subject **03861B**, in the *60 word-picture* experiment, and subject **03839B**, in the *60 word-picture* experiment were pooled.

Per-Subject Results

Subject	Accuracy (25 verbs)	Accuracy (486 verbs)
03861B	0.7802	0.7859
03839B	0.7644	0.7893

The accuracy of the pooled model was **0.5539** using Method 2.

5.5 Analysis of Naive Pooling

As shown above, the accuracy for the pooled model is far worse than that of the single-experiment model. A crucial assumption of the naively pooled model is that, for a given subject, the brain activation for word *X* is the same across experiments and studies. Since the subject was asked to generate a list of properties before the experiment, it seems reasonable to assume that the brain activation for the same word across studies would be similar. The poor accuracy points to two possible conclusions:

1. The accuracy of brain mapping to a common space is poor. That is, voxel (x,y,z) in experiment *A* is not the same as voxel (x,y,z) in experiment *B* due to inaccurate across-experiment mapping.

2. Brain activation across experiments is not the same. That is, the activation for word *X* in experiment *A* is not the same as word *X* in experiment *B* (either because of the different experimental paradigms, or because of experiment-to-experiment variance)

5.5.1 Qualitative Image Comparison

To explore these possibilities, we first qualitatively looked at the images across experiments. The images below are from the same subject in two different experiments.

Figure 1 and 2 above show a slice of the mean image, across all words, for the two experiments (i.e the average activation of each experiment). While most activity seems to take place in the occipital lobes, there does seem to be a difference between the two activation patterns (on average). To further investigate the possibility of divergent activation patterns across experiments, we looked closer at the images for specific words. The images below present the difference in activation patterns (specifically, the words-only image minus the word-picture image). The more red and orange areas, the higher the divergence of activation.

Figures 3,4 and 5 support the hypothesis that the activation patterns diverge from experiment to experiment.

Figure 2: Mean Image of Word-Picture Experiment

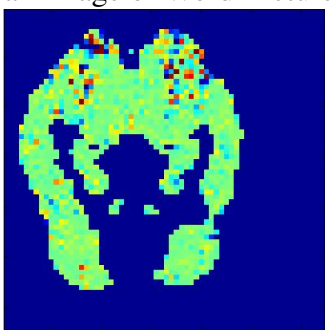


Figure 3: Mean Image of Words-Only Experiment

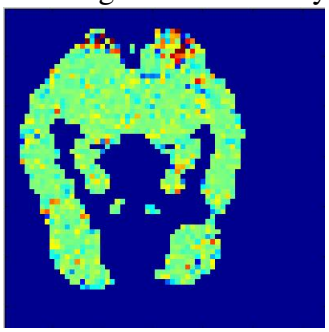
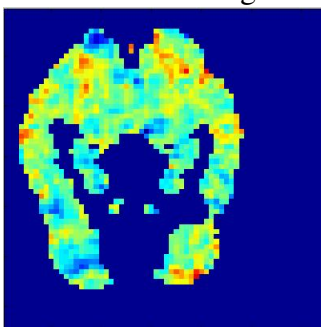


Figure 4: Difference Image: Butterfly



5.5.2 Stable Voxel Analysis

To investigate the differences in activation from experiment to experiment, we next explored the voxel selection method. Recall that voxels were selected on each training iteration, calculating a stability score for each leave-two-out pairing. In general, most of these voxels were maintained from iteration to iteration. Figure 6 and 7 below show the stable voxels with no words left out for the two studies.

Figure 5: Difference Image: Spoon

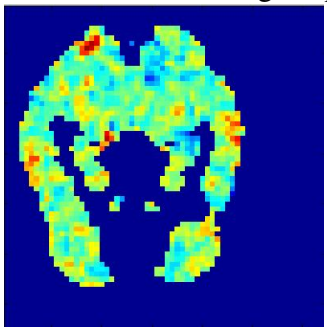


Figure 6: Difference Image: Foot

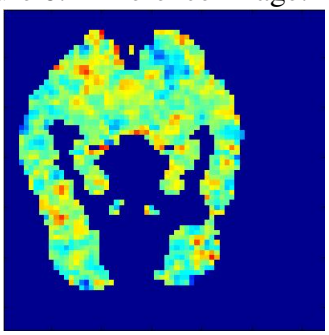
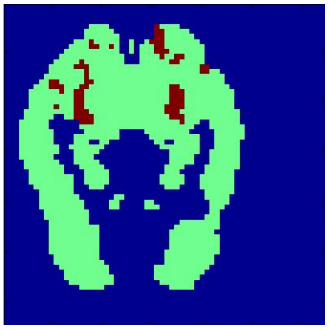


Figure 7: Stable voxels from Word Picture



Though these images are before intersection, it is evident that even the stable voxel locations diverge from experiment to experiment. Even more interesting is that after intersection, the stable voxels become sparse and sporadic (shown below). This result may be a strong case against intersecting the images in the pooled model (and perhaps taking a hierarchical approach to pooling instead).

Figure 8: Stable voxels from Words Only

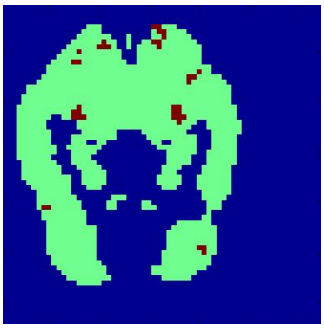


Figure 9: Stable voxels from Word Picture (after intersection)

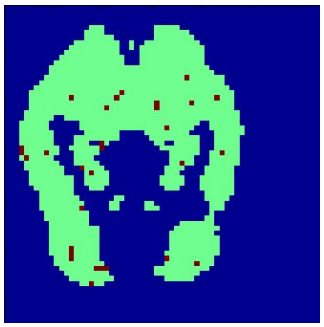
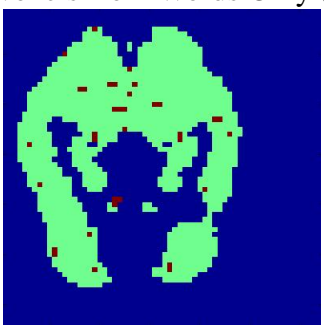


Figure 10: Stable voxels from Words Only (after intersection)



5.6 Voxel Selection Analysis

5.6.1 Introduction

A key component to the training of the model is the voxel selection method. Using the method described in section 4.3.2, we choose the top 500 voxels we think exhibit the most information about the input stimulus. We then learn the regression coefficients for only these voxels, and consequently use these voxels for the image comparison on each training iteration (as detailed in section 4.3). Because of this, the voxel selection method is a fundamental step in the preprocessing of the data. The current voxel selection method chooses those voxels which show consistent activation over each presentation of a given word. For example, a voxel which exhibits the same exact

activation for each presentation of each word would be granted the highest stability score.

The cognitive neuroscience literature seems to agree that the neural representation of objects is inherently distributed across the brain. That is, information is exhibited in the activation across many voxels instead of individual voxels. This has been illustrated by creating classification models which, when certain active regions of the brains are removed, can still classify the category of the input stimulus using the other, distributed, activation areas. If this is indeed the case, the method detailed in 4.3.2 may not be the optimal method for choosing voxels which are part of the distributed pattern associated with a stimulus word. Additionally, the voxels chosen with the 4.3.2 method may be voxels which are perceptually stable (those which are constantly active due to visual stimuli, instead of semantic reasons). While perceptually stable voxels for each word will help with classification, it will not provide insights into the way the brain represents conceptual knowledge. We explore these two possibilities in the following sections.

5.6.2 Semantically Stable Voxels

We have attempted to create a new voxel selection method. While the method described in section 4.3.2 chose voxels which were consistent across all words, this new method chooses voxels which are consistent across semantic categories. For each of the 12 semantic categories, the voxel's activation is averaged over each word's activation in that category. Then, each voxel has a 6×12 matrix where entry (i, j) is the average activation of the voxel on the ith presentation of the jth semantic category. The stability score is calculated as the average pairwise correlation of the rows. The top 500 stable voxels are presented below for the corresponding experiments shown in the images above. An interesting fact is that there are apparently less semantically stable voxels in the occipital lobes than the stable voxels chosen in section 4.3.2 – perhaps indicating that the method in 4.3.2 chooses perceptually-stable voxels, but not semantically stable.

Figure 11: Semantically Stable Voxels from Word Picture

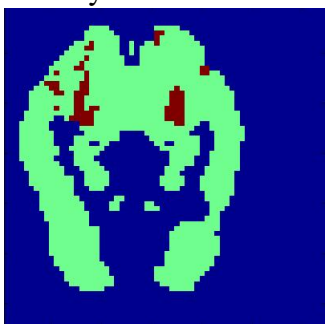
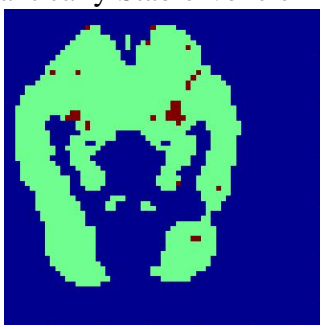


Figure 12: Semantically Stable Voxels from Words Only



5.6.3 Classifying Individual Voxel Activation

In this section we attempt to make two different classifiers for each voxel. Both classifiers attempt to classify voxel activation. In the first classifier, the classes are the word categories, in the second classifier the classes are the words themselves. We use a Gaussian Naive Bayes (GNB) classifier, using the voxel's activation as the feature and the category or word as the class. To evaluate the accuracy of the classifier, we use 6-fold cross validation, leaving out all the words from the ith presentation on the ith fold and calculating the rank accuracy of the correct prediction. The rank accuracy is defined as L/N where L is the number of classes ranked below the correct prediction by the classifier, and N is the total number of possible classes. To get a sense of the difference in accuracy from voxel to voxel, we also present the variance of the average rank accuracy across the voxels we have evaluated.

Category Classifier

To begin with, we create and evaluate classifers for 40 voxels - the top 20 voxels from the 4.3.2 stability score calculation, and the top 20 semantically stable voxels (detailed above). This is done for the same subject in two studies: subject **03616B** in the *60 word-picture* experiment and subject **04383B** in the *60 word-only* experiment.

Category Classifier Results

Subject	Voxels	Accuracy	Variance
03686B	4.3.2 Voxels	.5887	.001
	Semantically Stable	.5262	$\ll$.001
04383B	4.3.2 Voxels	.5326	$\ll$ .001
	Semantically Stable	.5365	$\ll$.001

Word Classifier
Similarly, we create and evaluate word classifiers for the same 40 voxels on the same subjects.

Word Classifier Results

Subject	Voxels	Accuracy	Variance
03686B	4.3.2 Voxels	.5262	$\ll$.001
	Semantically Stable	.5255	$\ll$.001
04383B	4.3.2 Voxels	.5084	$\ll$.001
	Semantically Stable	.5093	$\ll$.001

The motivation behind these classifiers is the following: we believe that the better the word (category) classifier does on a given voxel, the more information that voxel holds about which stimulus word (category) is being presented. Surprisingly, the classifiers perform poorly on both word and category classification. The difference between the two classifiers is also negligible given that we were testing the rank accuracy on 60 words compared to only 12 categories. Interestingly, both classifiers seem to perform slightly better on the subject in the *60 words-picture* experiment hinting that the stable voxels chosen in that experiment hold more information about the stimulus word. One possibility for this is that the stable voxels in the *60 word-picture* experiment could be chosen mostly from the occipital lobe (dealing with vision), and therefore yielding more perceptually-stable voxels. To test this hypothesis, we evaluate these classifiers on voxels specifically chosen from the occipital lobe.

5.6.4 Voxel Classifiers on Occipital Voxels

For both of the subjects above, we intersect the occipital voxels with the top 500 4.3.2-stable voxels. Of those stable occipital voxels, we choose 20 at random and train classifiers for them. The occipital regions of interest are left-inferior extrastriate, left-superior extrastriate, right-inferior extrastriate, and right-superior extrastriate.

Category Classifier Results

Subject	Voxels	Accuracy	Variance
03686B	Occipital	.5309	$\ll$.001
04383B	Occipital	.5023	$\ll$.001

Word Classifier Results

Subject	Voxels	Accuracy	Variance
03686B	Occipital	.5066	$\ll$.001
04383B	Occipital	.5010	$\ll$.001

Overall, the accuracy of the classifiers in the occipital lobe are either equal or slightly worse than the overall voxels. The low accuracy scores suggest that individual voxels do not contain much information about stimulus word or category – instead, as suggested by the litereature, information is likely exhibited in a distributed nature. The high accuracy of the Per-Subject model

could be a consequence of the top 500 stable voxels containing those which appear in most of the clustered activation patterns. However, classification accuracy may be increased with a voxel selection method which is more distributed in nature.

6 Feature Contribution

In the framework of this regression model, the only variable, given the data, is the feature set (the intermediate word-vector). Analyzing the feature set could yield insights into how they affect the model, and could provide a better way of choosing features. We attempt to analyze the importance of each feature by creating a ranking metric over all features.

6.1 Ranking Metric

To begin exploring the influence of each feature, we created a naive ranking metric over all subjects in an experiment.

1. For each subject, we create a matrix of dimension $n \times k$ where n is the number of training iterations and k is the number of features. Element (i, j) is the sum of all the regression coefficients for training iteration i and feature j.

2. Average the sum of coefficients for feature i over all training iterations. We do this for each subject, resulting in an $n \times k$ matrix where n is the number of subjects and k is the number of features. Element (i, j) is therefore the average sum of regression coefficients for subject i and feature j.

3. For each subject, multiply the sum of coefficients for each feature in the matrix in step 2 by the corresponding co-occurence scalar for each input word. This results in an $n \times k$ matrix for each subject where n is the number of stimulus words and k is the number of features. Element (i, j) yields some insight into how much feature j influences the predicted image for word i.

4. Sort each row of the matrix in step 3, and keep track of the ranking of each feature. The idea here is that the higher the number in element (i, j), the more influence the feature has in predicting the image.

5. Sort the average ranking of the features over all words, giving a $1 \times n$ vector for each subject, where n is the number of features. This final vector is the ranking of the features from most influential to least

6.2 Results

First, the ranking metric was calculated on the 25 verbs, on each of the experiments. The results, along with the metric's standard deviation from subject to subject are reported below.

As a sanity check, we first tested the metric on individual input words to see if the metric made sense. Below we present results on four words.

6.2.1 Features Ranked for 'Celery'

Celery
60 word-picture

Word	Mean	Std
'fill filled fills'	-55.335	48.1307
'taste tasted tastes'	-16.583	35.7863
'smell smells smelled'	-15.748	27.9572
'eat ate eats'	-10.363	35.778
'touch touched touches'	-2.0744	1.9043
'hear hears heard '	0	0
'listen listens listened'	0	0
'rub rubbed rubs'	0	0
'manipulate manipulates manipulated'	0	0
'run ran runs'	0	0
'push pushed pushes'	0	0
'move moved moves'	0	0
'fear fears feared'	0	0
'approach approaches approached '	0	0
'neared nears near '	0	0
'enter entered enters'	0	0
'drive drove drives'	0	0
'wear wore wears'	0	0
'lift lifted lifts'	0	0
'break broke breaks'	0	0
'ride rides rode'	0	0
'say said says'	0.4127	0.8158
'open opens opened'	0.9621	1.8796
'see sees'	6.494	11.9459
'clean cleaned cleans'	15.0677	8.8355

Looking at the absolute value average ranking metric, we see that celery is influenced, in order, by: fill, taste, smell, clean, eat. For the most part, these results make sense.

6.2.2 Features Ranked for 'Horse'

Horse
60 word-picture

Word	Mean	std
'run ran runs'	-12.812	8.7062
'smell smells smelled'	-4.653	8.2603
'hear hears heard '	-3.3658	6.4027
'fill filled fills'	-3.0061	2.6147
'push pushed pushes'	-2.8489	2.4849
'wear wore wears'	-2.0529	1.9394
'approach approaches approached '	-1.4918	0.7245
'drive drove drives'	-1.4036	2.0269
'break broke breaks'	-1.3652	1.9306
'eat ate eats'	-1.0193	3.519
'fear fears feared'	-0.9566	3.5247
'touch touched touches'	-0.9401	0.863
'enter entered enters'	-0.7823	1.9305
'taste tasted tastes'	-0.255	0.5503
'rub rubbed rubs'	-0.148	0.7781
'lift lifted lifts'	-0.14	0.7713
'manipulate manipulates manipulated'	0	0
'listen listens listened'	0.1206	0.223
'clean cleaned cleans'	1.3379	0.7845
'open opens opened'	1.466	2.8639
'say said says'	3.0028	5.9352
'see sees'	3.2908	6.0535
'move moved moves'	5.9705	6.9188
'neared nears near '	6.9209	5.2433
'ride rides rode'	64.481	33.9999

From the ranking metric, we deduce that the following features influence the predicted images for horse (in order): ride, run, near, smell. Again, these results make sense. Doing the same for coat, we find that the most influential features are: wear, clean, open, fill. And for car they are: near, drive, run, ride. From these results, we conclude that the ranking metric does indeed capture some information about the influence of each feature.

6.2.3 Average Feature Ranking for All Words

Using the absolute value of the average ranking score as our metric, we sort over the features in each experiment. The sorted features are shown below:

25 Verbs - Ranking Metric (Sorted)
60 word-picture

Word	Mean	Std
'fill filled fills'	-12.023	19.9632
'run ran runs'	-11.314	14.0694
'push pushed pushes'	-8.1831	14.1379
'touch touched touches'	-5.2838	11.9124
'approach approaches approached '	-4.2061	14.2629
'wear wore wears'	-3.9037	8.8312
'hear hears heard '	-3.8704	9.7702
'break broke breaks'	-3.2442	7.1885
'smell smells smelled'	-3.1958	10.925
'drive drove drives'	-3.1032	6.3041
'eat ate eats'	-1.639	8.3621
'lift lifted lifts'	-1.4027	11.7508
'enter entered enters'	-1.3977	5.4878
'taste tasted tastes'	-1.2991	6.1221
'rub rubbed rubs'	-0.8674	6.7226
'fear fears feared'	-0.6914	5.7035
'listen listens listened'	1.3832	9.0117
'manipulate manipulates manipulated'	2.5588	8.887
'open opens opened'	5.9717	8.069
'ride rides rode'	6.1357	15.5891
'say said says'	6.6715	10.6709
'see sees'	6.7401	8.3419
'move moved moves'	9.5533	15.7871
'clean cleaned cleans'	11.458	18.5581
'neared nears near '	14.3497	19.2366

25 Verbs - Ranking Metric (Sorted)
60 word-only

Word	Mean	std
’run ran runs’	-6.1278	8.4039
’wear wore wears’	-4.607	10.5456
’manipulate manipulates manipulated’	-3.5168	9.1308
’see sees’	-3.1066	5.4066
’neared nears near ’	-2.1966	5.2867
’smell smells smelled’	-1.7755	8.7858
’fear fears feared’	-1.7472	6.5352
’listen listens listened’	-1.5212	6.0582
’hear hears heard ’	-1.4661	8.5312
’rub rubbed rubs’	-1.1786	6.2618
’eat ate eats’	-1.132	4.3903
’drive drove drives’	-0.5756	5.3778
’break broke breaks’	-0.4536	7.1481
’lift lifted lifts’	-0.4335	3.9764
’approach approaches approached ’	-0.2085	3.83
’fill filled fills’	-0.0622	4.3569
’taste tasted tastes’	0.2161	6.269
’push pushed pushes’	1.2654	7.5568
’say said says’	1.6045	5.9476
’enter entered enters’	1.849	6.0186
’ride rides rode’	2.4562	6.1562
’touch touched touches’	4.0487	8.3434
’clean cleaned cleans’	4.6238	7.8603
’move moved moves’	4.9628	8.3683
’open opens opened’	8.0956	7.3042

25 Verbs - Ranking Metric (Sorted)
40 abstract-concrete

Word	Mean	Std
’fill filled fills’	-9.2802	13.7431
’break broke breaks’	-5.3055	10.2558
’taste tasted tastes’	-3.8028	9.3615
’neared nears near ’	-3.6529	12.9686
’eat ate eats’	-3.2085	10.7114
’listen listens listened’	-1.9642	11.6393
’see sees’	-1.6258	5.3182
’lift lifted lifts’	-1.0183	5.3378
’push pushed pushes’	-0.9993	10.0374
’wear wore wears’	-0.8388	14.9565
’manipulate manipulates manipulated’	-0.7005	9.5194
’rub rubbed rubs’	-0.6819	7.8198
’enter entered enters’	0.0205	6.896
’drive drove drives’	0.3292	4.8827
’hear hears heard ’	0.3588	4.5038
’fear fears feared’	0.3606	6.3995
’move moved moves’	1.1024	5.5986
’open opens opened’	1.6251	5.4175
’clean cleaned cleans’	2.4309	13.5127
’smell smells smelled’	3.1572	11.6229
’say said says’	3.1923	4.4763
’run ran runs’	4.0331	10.3749
’touch touched touches’	4.1268	8.6991
’approach approaches approached ’	4.3017	9.9243
’ride rides rode’	7.6661	16.5263

6.2.4 Using Ranking Metric to Choose Feature

Using this ranking metric, we ranked the 486 commonly co-occurring verbs. Then, we ran the model using the top 25 verbs (using the metric). The top 25 verbs in the 486 verb set were: walk, chop, ring, cut, plan, open, see, stand, tour, close, repair, travel, bite, unite, build, press, order, pull, tip, live, ride, dance, say, dry, move. The results are below, with the accuracies from the original 25 verb set listed for comparison.

60 word-picture Results

Subject	Accuracy	Accuracy (old 25 verbs)
03616B	0.7853	0.8249
03839B	0.7146	0.7644
03861B	0.7068	0.7802
03921B	0.6322	0.7237
03993B	0.7519	0.7841
04008B	0.6864	0.8542
04019B	0.6932	0.7299
04124B	0.7525	0.6785
04228B	0.6796	0.8230

Average: 0.71, Old Average: .77

40 abstract-concrete Results

Subject	Accuracy	Accuracy (old 25 verbs)
05248B	0.6397	0.5436
05300B	0.4833	0.4731
05344B	0.4615	0.5705
05344B	0.5910	0.7565
05181B	0.4821	0.6308
05230B	0.5910	0.4705
05245B	0.5885	0.3398
05258B	0.5192	0.6013
05324B	0.4705	0.5282
05236B	0.3846	0.4910
05222B	0.6244	0.6051
05176B	0.6090	0.5090

Average: 0.53, Old Average: .52

60 word-only Results

Subject	Accuracy	Accuracy (old 25 verbs)
04383B	0.7333	0.7424
04480B	0.5350	0.6017
04564B	0.6040	0.5350
04605B	0.6322	0.7565
04619B	0.5458	0.6057
04647B	0.6718	0.7186
04408B	0.6198	0.6610
04550B	0.6090	0.4780
04597B	0.8062	0.7780
04617B	0.6215	0.6825
04639B	0.7684	0.7830

Average: .65, Old Average: .67

On average, using this ranking metric to choose the feature set seems to do worse. However, some subject accuracies were improved. Future work may elucidate what causes the higher regression coefficients for certain features.

7 Conclusion

In this thesis, we attempted to create a naive model which predicts brain activity across subjects and studies. Our naive solution was to merge the data from multiple subjects or studies and treat them as if they came from one subject, in one experiment. The poor accuracy of this solution suggested two distinct possibilities. First, that brain activation diverges for the same word across studies (e.g. the brain activation for the word *celery* in the *60 word-picture* experiment is different (for the same person) from the brain activation for the word *celery* in the *60 words-only* experiment). The second possibility is that the process of mapping brains to a common space is inaccurate which therefore restricts us from treating the data as if it came from the same subject. Our analysis showed that brain activation is indeed different from study to study, implying that we cannot simply merge the data.

In addition to the exploring a pooled model, we analyzed the process of voxel selection. We found that the current process may be choosing voxels which are perceptually stable, but not semantically stable. That is, they exhibit information about the visual stimuli and not the semantic properties of the stimuli. However, after creating a process for choosing semantically stable voxels, we find that accuracy is (for the most part) the same. Interestingly, we find that less voxels are chosen in the occipital lobe using the semantically stable procedure. Finally, we found that choosing features solely based on the magnitude of their influence on the predicted image is not an ideal feature selection method.