# Decoding Word Semantics from Magnetoencephalography Time Series Transformations

Alona Fyshe

May 15, 2012

**Abstract**

In the brain, millions of neurons interact to represent thoughts and create knowledge. The way in which neurons represent thought is one of the major open questions of neuroscience. Advances is neuroimaging, including Magnetoencephalography (MEG), have allowed us to capture data on a finer time scale and record the neural signature of thoughts evolving in the human brain. In this report we explore several methods of analyzing the signal recorded by the MEG machine, and use that analyzed signal to predict the word a person is reading. The utility of different signal transformations is assessed by measuring their ability to encode the semantic properties of words. We hope the discoveries outlined in this report will help to explain how groups of neurons can encode concepts, and in turn will lead to a better understanding of the human brain.

## 1 Introduction

The human brain's system of knowledge representation has been pondered by scientists and philosophers alike. What actions do our brains perform when we recall a memory or retrieve a fact? In what way does brain activity change depending on the concept we hold in mind? This report attempts to answer these questions by exploring several transformations of Magnetoencephalography (MEG) recordings. These transformations of the MEG signal

provide different representations of the underlying neural activity. For example, some transformations capture the phase of the neural oscillations, and others capture the change in neuronal firing over time. Finding the relation of MEG signal transformations to thought patterns allows us to understand how information may be encoded in the brain, and thus elucidates the *neural code*.

Much research has dealt with brain images generated by functional magnetic resonance imaging (fMRI). fMRI measures the ratio of oxygenated and de-oxygenated hemoglobin in the blodd, which is affected by brain activity as well as other factors. Blood deoxygenation is an indirect measure of the rate of neuronal firing. A more recent imaging technique is magnetoencephalography (MEG), which uses sensors positioned in a helmet to measure the weak magnetic field caused by neurons firing in a coordinated fashion. Changes in magnetic field are a more direct measure of neuronal firing, and have much better time resolution than the slow-moving hemodynamic response that governs changes in blood oxygen levels. MEG can capture a sensor reading every millisecond (1000 Hz), whereas fMRI machines typically capture an image every 2 seconds (0.5 Hz). However, the spatial resolution of the MEG sensors is much coarser than the resolution of the fMRI image. Generally, one can think of the MEG image as a set of time series, whereas the fMRI image can be thought of as a 3D image. The MEG machine has three sensors in 102 different locations for a total of 306 sensors (Figure 1 illustrates the layout of the 102 sensor locations). In each location there is a magnetometer and two gradiometers. The magnetometer measures the strength of the magnetic field created by the neurons directly under the sensor. The MEG gradiometer measures the spatial gradient of this magnetic activity (i.e. the change in magnetic field strength over space) measured in Teslas per meter (T/m) [7]. Because the gradiometer measures a spatial gradient, two gradiometers are positioned at each location, and the direction of their gradients is perpendicular.

This report focuses on the task of predicting the word a person is reading based on the MEG signal. We collected MEG data while 9 subjects viewed 60 word/picture pairs, with 20 interleaved repetitions (single trials) per word. Each of the 60 words are concrete nouns from one of 12 categories (animals, tools, buildings, food, furniture, insects, transportation, clothing, body parts, building parts, utensils and objects). Each sensor time series is down sampled to 200 Hz and has 340 time samples, for a total of 1.7 seconds. Complete details of MEG data collection and preprocessing steps appear in

Appendix A.

For each of the 60 words we have 218 semantic features. These semantic features are rated answers $[1 \dots 5]$ to questions like "Do you hold it to use it?" and "Is it alive?". The answers to these 218 questions were generated via Mechanical Turk, an online crowd-sourced question answering service. This projection of the 60 words into a semantic space allows us to decompose the neural representation of a word into the parts related to each of the semantic features. Past work has shown that aspects of word semantics have differing effects on the MEG signal recorded from different brain locations and points in time after stimulus onset [16]. We wish to explore not only when and where these effects happen, but whether there are other representations of the MEG signal, such as frequency, that show differential effects.

Given infinite amounts of both data and computational power, this task would be conceptually simple. We could learn functions over features derived from transformations of the MEG data to predict each of the 218 semantic features. We could measure the predictive performance of functions learned on subsets of features from each MEG feature type, as well as all cross products of subsets between the MEG feature types. Obviously, the number of functions learned and the amount of data required quickly gets out of hand. Given that we have limited computational resources and time, how can we best use the data?

First, let us reduce the dimensionality of the problem by considering each pair of MEG feature type and semantic feature independently. This reduces the size of the computational problem, and it also allows us to directly connect a MEG feature type to the ability to predict a given semantic feature. Identifying such connections is a step on the way to uncovering the neural code. Then, for each MEG feature type, we can use regularized regression to choose an optimal subset of the features. We can examine the performance of the learned functions over time, brain locations, and frequencies (where applicable) to further relate the semantic features to MEG features.

## 1.1 Zero Shot Learning

While we have MEG data for only 60 words, there are tens of thousands of words in the English language. In addition, new words are continually being introduced into the lexicon. We would like to develop a system that can predict the word a person is reading even though we have not collected an MEG recording of that word. We accomplish this by decomposing our words

into their semantic features.

As a result of our projection the 60 words into 218-dimensional semantic space, we can perform "Zero Shot Learning" [12]. Zero Shot Learning allows us to predict the word for an MEG brain image for which we have never seen a training example. We train an independent function $f$ for each of the 218 semantic features, and so we can predict a new vector of semantic features that is unlike any combination of semantic features that we encountered during training. This is a particularly attractive characteristic for the task of predicting words from brain images, as there are tens of thousands of words in the English language, and we cannot hope to capture MEG recordings for all of them. Zero Shot learning allows us to break a word down into its semantic features and use that information to recognize new words without having to collect additional data.

Formally, let us define a word $w$ as having a semantic decomposition into semantic features $\vec{s}_w = \{s_1 ... s_m\}$ where $m = 218$ is the dimension of our semantic space. Typically one might use machine learning to learn a function $f$:

$$f(X) \rightarrow w$$

So that we would predict the word $w$ based on the MEG data $X$. Zero Shot Learning utilizes a known mapping

$$w \rightarrow \{s_1 \dots s_m\}$$

and then trains $m$ independent functions

$$\begin{aligned} f_1(X) &\rightarrow s'_1 \\ &\vdots \\ f_j(X) &\rightarrow s'_j \\ &\vdots \\ f_m(X) &\rightarrow s'_m \end{aligned}$$

where $s'$ represents the value of a predicted semantic feature. The output of $f_1 \dots f_m$ are combined to create a predicted semantic vector

$$\vec{s'} = \{s'_1 \dots s'_m\}.$$

We then define a function $d(\{s'_1 \dots s'_m\}, \{s_1 \dots s_m\})$ that quantifies the dissimilarity between two semantic vectors. Any distance metric could be used here; we will use cosine distance:

$$d(\vec{s}, \vec{s'}) = 1 - \frac{\sum_i s_i s'_i}{\sqrt{(\sum_i {s_i}^2)\left(\sum_i {s'_i}^2\right)}}$$

We choose the word $w$ with the semantic vector $\vec{s}_w$that minimizes $d(\vec{s}_w, \vec{s'})$

$$w = \underset{w}{\operatorname{argmin}} \quad d(\vec{s}_w, \vec{s'})$$

as the final predicted word.

# 2 MEG Feature Transformations

Zero Shot Learning has provided us with a mechanism to move from a set of MEG time series $X$, to a vector of semantic features $\vec{s'}$ to arrive at a predicted word $w$.

$$f_i(X) \dots f_m(X) \rightarrow \{s'_1 \dots s'_m\}$$
$$w = \underset{w}{\operatorname{argmin}} \quad d(\vec{s}_w, \vec{s'})$$

Now we seek to define transformations $g$ on the MEG time series $X$ which may provide additional information to the learned functions $f$ so that the mapping

$$f_i(g(X)) \dots f_m(g(X)) \rightarrow \{s'_1 \dots s'_m\}$$

increases the chance that the word $w$ that minimizes $d(\vec{s}_w, \vec{s'})$ is the correct word label for the MEG recording $X$. The space of possible functions $g$ is infinite. In this section we select and define 7 functions from the infinite space of functions for further exploration. We call the output of these functions *MEG feature types*.

Examples for sensor 77 and each of the 7 MEG feature type are shown in Figures 2-5. The position of sensor 77 in the MEG helmet can be seen in Figure 1. A summary of the window size for each MEG feature type appears in Table 1.

Throughout this report, we will refer to the MEG signal as $X \in \Re^{306 \times T}$, where $T$ is the total number of time points. $X_{i,j}$ is the reading from sensor $i$ at time $j$. A superscript on $X$ or one of its computed features denotes that a variable is computed from or represents data from a single trial, of which there are $R$. A full listing of variables and their definitions appears in Table 6 of the Appendix.

Table 1: The window sizes and window overlaps (in milliseconds) used to compute the different MEG features. The window overlap dictates how much the windows of adjacent MEG features overlap. For example, two adjacent windowed mean MEG features will share 5 time points in common (25 ms). In the case of the Continuous Haar transform, there are multiple window sizes, with overlaps equal to their width minus one time point.

| Feature name | Window size (ms) | Window overlap (ms) |
|---|---|---|
| Raw | 5 | 0 |
| Windowed mean | 50 | 25 |
| Mean slope in window | 50 | 25 |
| Gradiometer norm | 5 | 0 |
| FFT Power | 100 | 50 |
| FFT Phase | 100 | 50 |
| Continuous Haar | - | - |

## 2.1 Raw Signal

To create the Raw MEG feature type, all $R = 20$ trials for a given word are averaged, and the averaged signal for each of the time points become the features.

$$g_{i,j}(X^1 \dots X^R) = \frac{1}{R} \sum_{\tau=1}^{R} X_{i,j}^{\tau}$$

and we define $g_{i,j}$ for $i = 1 \dots 306, j = 1 \dots T$

If the raw signal is the best MEG feature, then it is the magnitude of the magnetic field (or gradient of the field, in the case of gradiometers) that best encodes the semantic features of the words.

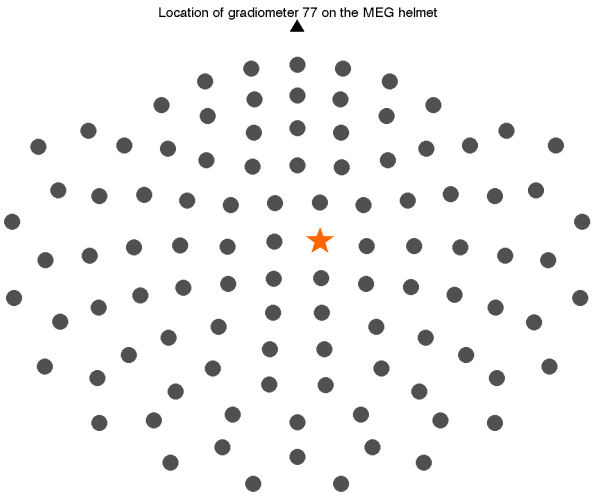


Figure 1: The position of the sensor whose signal was used to create example plots in Figures 2-5. Sensor 77 is shown as an orange star, while other sensors shown as grey circles. The black triangle represents the subject's nose, and the view is from above.

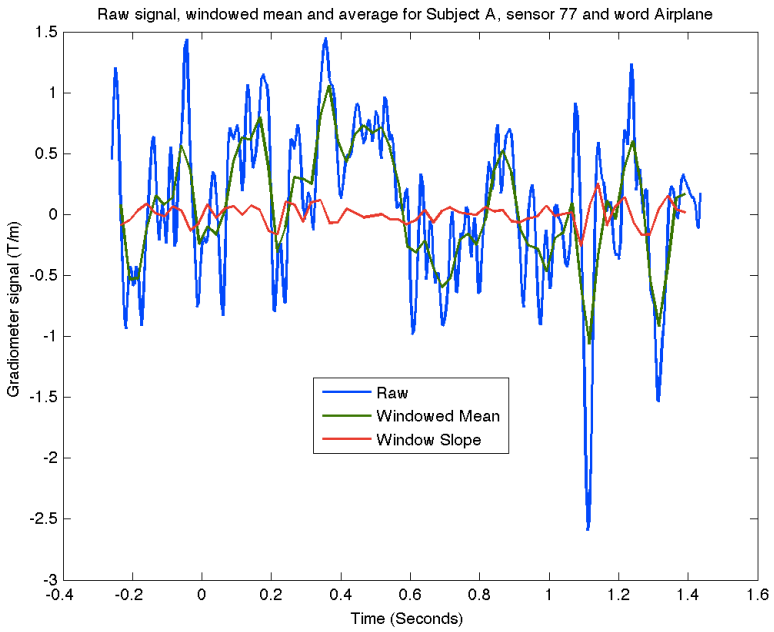


Figure 2: An example of the raw and windowed features for sensor 77 and the word Airplane. The windowed mean creates a smoother time series than the raw signal. The windowed slope is normalized so that it represents the average difference between adjacent time points within a given window.

The window width is 50ms (10 samples) for both windowed features.

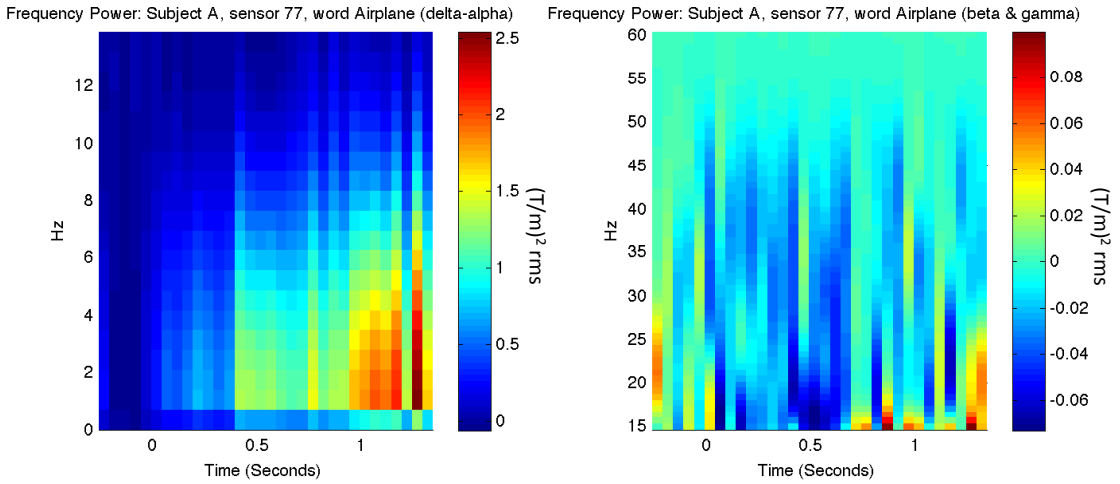


Figure 3: An example of the frequency power created from the short time Fourier transform (STFT) of sensor 77 recorded during the word Airplane. The window width for the STFT is 100ms (20 time samples), and they overlap by 50ms. Frequency bands delta through alpha are shown on the left, and gamma on the right. Note that the scale is different for the two plots.

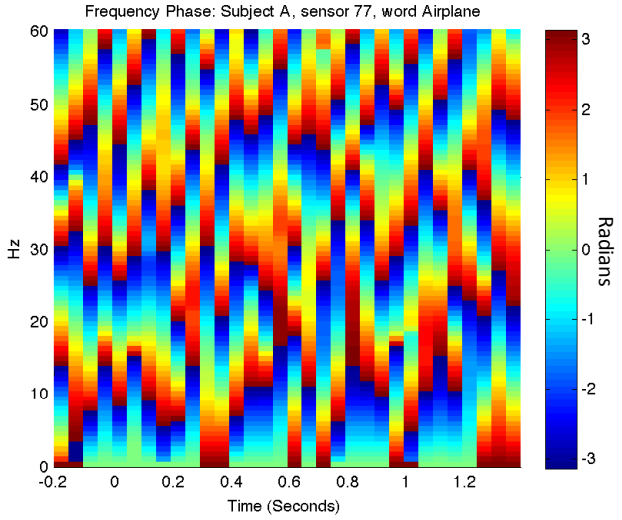


Figure 4: An example of the frequency phase feature created from the short time Fourier transform (STFT) of sensor 77 recorded during one trial of the word Airplane. The window width for the STFT is 100ms (20 time samples), and they overlap by 50ms.

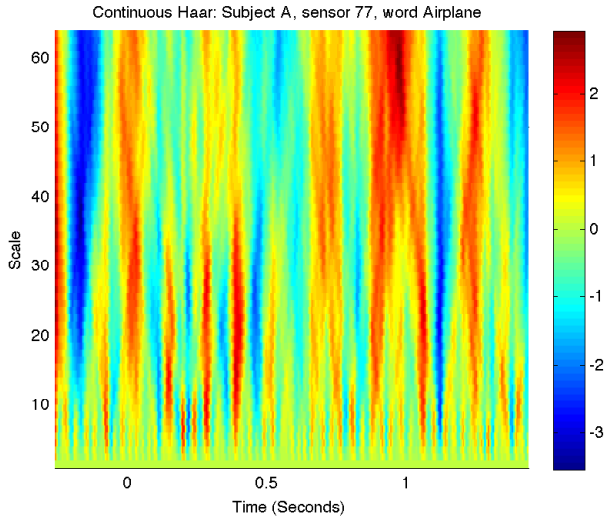


Figure 5: An example of the coefficients returned by the continuous Haar wavelet transform of sensor 77 recorded during the word Airplane. Smaller scales correspond to smaller wavelets and higher frequencies.

Though we refer to this feature type as Raw, it should be noted that some preprocessing has been performed on the MEG data. A full explanation of the preprocessing appears in Appendix A.

## 2.2 Average in a Window (Windowed Mean)

To create the Windowed Mean MEG feature, all trials for a given word are averaged, and then the time points within 50ms windows (with 25ms overlap) are averaged. The transformation $g$ is:

$$g_{i,j}(X^1 \dots X^R) = \frac{1}{RW} \sum_{t=0}^{W-1} \sum_{\tau=1}^{R} X^{\tau}_{i,j+t}$$

where $W$ is the width of the time window, $i$ is the sensor number and $j$ is the time index. Here $W = 10$ (50 ms) and the windows overlap for 25 ms. We define $g_{i,j}$ for

$$i = \{1 \dots 306\}$$
$$j = \{1 : 5 : (T - W)\}$$

where ": 5 :" denotes that j is incremented in steps of 5 (consistent with Matlab notation).

Like the Raw Signal, if the Windowed Average MEG signal is the best MEG feature, then it is the magnitude of the magnetic field (or gradient of the field) that best encodes the semantic features of the words. But, unlike the Raw Signal, the average is smoother and less noisy. If the Windowed Average outperforms the Raw Signal, this indicates that averaging reduces the noise more than it reduces the signal.

## 2.3 Average Slope in a Window (Windowed Slope)

To create the Windowed Slope MEG feature, all trials for a given word are averaged, and then the difference between adjacent time points within 50ms windows (with 25ms overlap) are averaged. The transformation $g$ is:

$$g_{i,j}(X^1 \dots X^R) = \frac{1}{RW} \sum_{t=0}^{W-2} (\sum_{\tau=1}^{R} X^{\tau}_{i,j+t} - \sum_{\tau=1}^{R} X^{\tau}_{i,j+t+1})$$

where $W$ is the width of the time window. Again, $W = 10$ (50 ms) and the windows overlap for 25 ms. $g_{i,j}$ is defined for

$$i = \{1 \dots 306\}$$
$$j = \{1 : 5 : (T - W)\}$$

If the Windowed Slope outperforms other features, then it is the trend of the signal within a window that encodes a semantic feature. That is, the actual level of the MEG signal is not important, rather its movement up or down is what carries information.

## 2.4 Euclidean Norm of Gradiometers

To create the Euclidean Norm of Gradiometers MEG feature, all trials for a given word are averaged, and the euclidean norm of the two gradiometer signals is calculated:

$$g_{i,j}(X^1 \dots X^R) = \sqrt{\left(\frac{1}{R} \sum_{\tau=1}^{R} X^{\tau}_{i,j}\right)^2 + \left(\frac{1}{R} \sum_{\tau=1}^{R} X^{\tau}_{i+1,j}\right)^2}$$

$$i = \{1 : 3 : 306\}$$
$$j = \{1 \dots T)\}$$

Each triplet of magnetometer, first and second gradiometers appear in consecutive rows in our MEG signal matrix. Thus, the signals from the first and second gradiometers are found in rows $\{1, 2\} + 3n$ for $n = 0 \dots 101$. This transformation $g$ creates one time series per helmet location. Magnetometers are discarded.

If the Norm of Gradiometers performs well, it shows that the sign of the gradiometers does not carry information, but rather the combined activity that encodes semantic information.

## 2.5 Frequency Power and Phase

To create Phase and Power features, a short time Fourier transform (STFT) is applied to single trials, and the resulting Fourier coefficients $S$ are used to calculate the power and phase in each frequency band. Fourier coefficients are calculated with the Fourier transform:

$$S_k = \int_{-\infty}^{\infty} f(t) e^{-2\sqrt{-1}\pi t k} dt$$

where $k$ is the frequency of interest, $t$ is time, and $f(t)$ is the continuous and differentiable function that defines the value of the signal at time t. Of course our MEG time series is neither continuous nor infinite, so we use the discrete approximation:

$$S_{i,k}^{\tau} = \frac{1}{T} \sum_{t=1}^{T} X_{i,t}^{\tau} e^{\frac{-2\sqrt{-1}\pi k(t-1)}{T}}$$

$S_{i,k}^{\tau}$ is the Fourier coefficient for frequency $k$ and sensor $i$ over all time $1 \dots T$ and $\tau$ is the trial number.

Non-stationary signals are signals which have frequency components that change over time. The Fourier Transform is formulated for stationary data, where the frequency components are identical within a time window. When applied to non-stationary data, the results can be ambiguous - that is two signals with contributions from the same frequency bands but at different times will produce the same frequency power patterns. To compensate for

this, the Short Time Fourier Transform (STFT) was invented. The Short Time Fourier Transform computes the Fourier coefficients for (possibly overlapping) windows of time, with the hope that the signal will be stationary or near-stationary within a time window. For the STFT we define

$$S^{\tau}_{i,j,k} = \frac{1}{W} \sum_{t=0}^{W-1} X^{\tau}_{i,t+j} e^{\frac{-2\sqrt{-1}\pi kt}{W}}$$

where j is the first point in a time window and W is the window width. Here, $W = 20$, or 100 ms and the windows overlap by 50 ms. We compute $S^{\tau}_{i,j,k}$ for

$$i = \{1 \dots 306\}$$
$$j = \{1 : 10 : (T - W)\}$$

Frequencies between 0 and 60 are computed.

The power for trial $\tau$, sensor $i$ at time $j$ and frequency $k$ is calculated as

$$\text{power}(i, j, k, \tau) = c|S^{\tau}_{i,j,k}|^2$$

where $c$ is a constant that depends on the window function used, and the sampling frequency. Phase is calculated as:

$$\text{phase}(i, j, k, \tau) = imag(\log(S^{\tau}_{i,j,k}))$$

where *imag* returns the imaginary portion of the complex coefficient. Power and phase are calculated for each single trial independently, and then the average is taken over the single trials. Our final functions $g$ operate on the Fourier coefficients.

$$g^{\text{power}}_{i,j,k}(S^{1}_{i,j,k} \dots S^{R}_{i,j,k}) = \frac{1}{R} \sum_{\tau=1}^{R} \text{power}(i, j, k, \tau)$$

$$g^{\text{phase}}_{i,j,k}(S^{1}_{i,j,k} \dots S^{R}_{i,j,k}) = \frac{1}{R} \sum_{\tau=1}^{R} \text{phase}(i, j, k, \tau)$$

The valid indices for $i$ and $j$ are the same as for the Fourier coefficients:

$$i = \{1 \dots 306\}$$
$$j = \{1 : 10 : (T - W)\}$$

The time windows of the STFT have width 100ms with 50 ms overlap. Due to the low pass filter on the original data, frequencies above 60 Hz are eliminated.

The rhythmic coordination of many neurons to form oscillations is a topic of great interest amongst MEG researchers (e.g. [15], [11], [5]). If the STFT Power feature performs best, then the activity of neurons firing at a particular rate is what encodes information. If the STFT Phase feature performs best then it is not the strength of the oscillations in a frequency band, but their synchronization to the stimulus that is important.

## 2.6 Continuous Haar Wavelet Decomposition

To understand the Continuous wavelet decomposition, it is easiest to start with the discrete wavelet decomposition. During a discrete wavelet decomposition, the mother wavelet template is used to create daughter wavelets at various scales. At first the width of the daughter wavelet is exactly 2 time samples. It is convolved with the signal for adjacent time points to create *detail coefficients*. This process is equivalent to laying many copies of the wavelet end to end and convolving it with the signal. For the Haar wavelet, the detail coefficients represent the difference of adjacent time points. The signal is then down sampled by averaging adjacent time points to create *approximation coefficients*. We again convolve the Haar wavelet with these new down sampled approximation coefficients, effectively stretching the wavelet's scale to twice its previous width. This process continues for some fixed number of iterations. After $N$ iterations we are left with $N$ sets of detail coefficients and one set of approximation coefficients (the intermediate approximation coefficients having been used to create subsequent detail coefficients).

The function for the Haar wavelet is a step function:

$$\psi(t) = \begin{cases} -1 & \text{if } 0 \leq t < 0.5 \\ 1 & 0.5 \leq t < 1 \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $t$ is time. In a continuous wavelet transform, instead of aligning the wavelets end to end (as in the discrete case), we shift the wavelet by one time tick and recompute the coefficients. This results in a coefficient matrix with dimensions $S \times T$ where $S$ are the scales of the wavelet, and $T$ is the number of time ticks. Due to the slow time shifting of the underlying wavelet, the continuous wavelet transform is also more robust to noise and time shifts of frequency components[1].

The wavelet decomposition is windowed at multiple time scales, and so, unlike the Fourier Transform, it can handle non-stationary data with no alterations to the method. This makes it a particularly attractive candidate for analyzing MEG data. A pictorial representation of the coefficients from a continuous Haar wavelet transformation can be seen in Figure 5. For this study, scales $[1 \ldots 64]$ were used.

Let $C^{\tau}_{i,j,k}$ represent the coefficient resulting from convolving the signal from sensor $i$ with a Haar wavelet with scale $k$, centered at time point $j$ for single trial $\tau$. Then we define the transformation function $g$ as:

$$g_{i,j,k}(C^1_{i,j,k} \ldots C^R_{i,j,k}) = \frac{1}{R} \sum_{\tau=1}^{R} C^{\tau}_{i,j,k}$$

The Haar wavelet decomposition represents many different types of information. The continuous wavelet transform is an over-complete representation of the signal, so like the discrete case, one can fully recreate the original signal with a linear combination of the wavelet coefficients. The wavelet coefficients can also be used to extract frequency and phase information from the signal. Wavelets of a particular scale correspond to different frequency bands; when coefficients of a particular scale are high, the power in the corresponding frequency band is also high. Wavelets also encode phase information. When a wavelet coefficient in a particular band is high at a particular position in time it signals that the corresponding frequency band is in phase with the wavelet at that time point. If the Continuous Wavelet features perform the best, then it may be a combination of phase, power and the raw signal that contribute to the decoding of semantics. In addition, the robustness of the Continuous Wavelet Transform to noise may contribute to its performance.

[1] Alona Fyshe's previous (unpublished) study for a class project explored the usefulness of several wavelet types using both the discrete and continuous transform. The conclusion was that the continuous Haar wavelet transform produced the best features for semantic feature prediction. For that reason no other wavelet types are explored here.

It is important to remember that the Haar wavelet is not smooth. Due to this fact the extracted features are not a perfect replicate of the STFT phase and frequency features.

# 3 Related Work

Past work has found useful signal amongst most of the MEG features proposed here. For example, the power in gamma frequency bands (>30 Hz) has been shown to be an indicator of attention and memory, as well as the coordination of brain areas [9]. While the task analyzed in this report does not involve memory, we consider phase features because of their possible role in top-down influence and coordination.

A common strategy when dealing with low SNR (signal to noise ratio) is to take a windowed average of the signal. In [4], the average amplitude in several time windows were used as input to an SVM classifier. The classifier was trained to distinguish living vs. non-living things (Mean accuracy of 76%) and individual words (83% accuracy). Though this study distinguished between only 10 words, the stimuli were presented both as visual and auditory stimulus. The study showed that the words could be decoded across presentation modalities.

The continuous wavelet transform has been used to decode the movements made by a subject [13] or to detect networks of neuronal activity related to movement [1]. Though these studies did not involve language, it has been proposed that language processing may be a distributed task that, for example, involves the motor cortex when the language being perceived is movement related. The idea of a distributed system of semantic representation has stirred up controversy (see [14]). Still, we would like to leave open the possibility of involvement of motor cortex and visual cortex (and their associated rhythmic activity) when learning our semantic prediction functions $f$.

A full examination of the usefulness of wavelet transforms for EEG data is given in [17], where wavelets were used to classify between three different cognitive tasks (multiplication, mental rotation of 3D object, silent letter composition) and 6 motor tasks (imagined movements). This study found that the information in several different wavelet types could be used to successfully differentiate between cognitive and imagined motor tasks. They also found improved performance using wavelet packets rather than the raw

coefficients. Due to time constraints, wavelet packets are not explored here.

# 4 Prediction Framework

We turn now to the methods we use to evaluate the utility of each MEG feature. To learn a function that predicts each of the 218 semantic features we employ $L_2$ regularized regression:

$$\hat{\beta} = \underset{\beta}{\operatorname{argmin}} \left\{ \sum_{i=1}^{N} (y_i - \beta_0 - \sum_{j=1}^{P} \beta_j x_{i,j})^2 + \lambda \sum_{j=1}^{P} \beta_j^2 \right\} \tag{2}$$

where $N$ is the total number of training instances and $P$ is the total number of features, $y_i$ is the label for training instance $i$ and $x_{i,j}$ is the $j$th feature of the $i$th training instance and $\beta$ represents the weights optimized.

$L_2$ regularized regression, or Ridge Regression, has several nice properties. Firstly, the regularization automatically down weights less important features. Secondly, it has a closed form and can be solved without gradient descent methods. Thirdly, because $L_2$ produces a linear predictor, we can employ Generalized Cross Validation (GCV) to choose our Lambda parameter [6]. GCV allows us to calculate the leave one out cross validation (LOOCV) performance of a regressor without having to train N separate functions. Let $\hat{f}$ be a function learned using all instances $x_i$ and $\hat{f}^{-i}$ be a function learned using training instances $\{x_j : j \neq i\}$. The LOOCV is calculated as

$$\frac{1}{N} \sum_{i=1}^{N} (y_i - \hat{f}^{-i}(x_i))^2 \tag{3}$$

$$\tag{4}$$

which can be shown [8] to be equal to

$$= \frac{1}{N} \sum_{i=1}^{N} \left( \frac{y_i - \hat{f}(x_i)}{1 - S_{ii}} \right)^2 \tag{5}$$

where $S_{ii}$ is the $i$th diagonal of the matrix S, and S is the portion of the learned weight matrix that depends only on the data $x$:

$$\hat{y} = Sy$$

where $\hat{y}$ is the value predicted by a function trained with labels $y$. This formulation is convenient because the matrix $S$ depends only on the training data. This allows us to remove the effect of the training data from the prediction via the denominator in Equation 5. The final GCV approximation is:

$$\frac{1}{N}\sum_{i=1}^{N}\left(\frac{y_i - \hat{f}(x_i)}{1 - trace(S)/N}\right)^2. \tag{6}$$

Recall that for each of our 60 words we have human-curated ratings for 218 semantic features, and that we will use this decomposition to perform Zero Shot Learning. To test our Zero Shot learning performance, we could hold out one training instance, predict a new semantic feature vector and then use that new vector to rank the full set of 60 training words. However, a naive predictor could perform very well on this task - as only one of the 60 words was not seen during training, an algorithm that always ranked the unseen vector first would perform with 100% accuracy. For this reason we leave out a pair of words and their associated MEG feature vectors. Then the task is to correctly assign the two held out words to the two held out MEG feature vectors. To assign words to MEG feature vectors we use the cosine distance between the predicted semantic feature vectors and the true semantic feature vectors and choose the assignment that minimizes the sum of the two distances. Though this procedure is technically leave two out cross validation at the word level it is leave one out cross validation at the *word pair* level. For simplicity, we will refer to this cross validation procedure as leave two out cross validation (LTOCV) and the test as 2 vs 2. For each fold of the LTOCV we can use GCV on the training set to choose an optimal $\lambda$ with which to compute the final weight vector $\beta$.

Within our 60 words there are 12 word categories, with 5 words per category. Because they are close in semantic space, words within category (e.g. screwdriver and hammer or lettuce and celery) may be more difficult to distinguish between that words across category. For this reason, a cross validation partition may be particularly difficult if it happens to have a lot of same-category pairs. To address this problem we perform 5 rounds of LTOCV to minimize the chance that we choose a difficult partitioning (or

an easy one). If there is no relationship between the MEG data and word semantics, the expected performance on the 2 vs. 2 test is 50%.

To train the regressors we use 750ms of MEG signal beginning immediately after the onset of the stimulus. 750ms is the generally agreed upon time at which semantic processing has finished. For features created from MEG signals using a window we used those windows with midpoints between 0 and 750ms after stimulus onset. We standardize the semantic features so that each has mean 0 and standard deviation 1.

## 4.1 Percent of Variance Explained

In addition to the 2 vs. 2 task described in Section 4, we would also like to evaluate the performance of each MEG feature and each semantic feature individually. To evaluate individual semantic feature performance for a given MEG feature, we train a set of regressors using the framework described in Section 4. We then calculate the Percent of Variance Explained (POVE) for that MEG feature and semantic feature combination. POVE is:

$$\text{POVE} = 1 - \frac{\sum_i (f_i - y_i)^2}{\sum_i (y_i - \bar{y})^2}$$

where $y_i$ is the true value of the semantic feature for test instance $i$, $f_i$ is the predicted semantic feature value, and $\bar{y}$ is the mean of all $y_i$.

Some will be familiar with POVE by its other name: $R^2$, or coefficient of determination. In the context of regression, one often computes $R^2$ on the training data to measure the amount of variation *in the training data* that is explained by the model. Under those circumstances, some nice properties hold:

$$\sum_i (y_i - f_i)^2 + \sum_i (f_i - \bar{y})^2 = \sum_i (y_i - \bar{y})^2 \tag{7}$$

And thus

$$\text{POVE} = R^2 = \frac{\sum_i (f_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} \tag{8}$$

Under these constraints, $R^2$ cannot be negative. However, in this study, we calculate the POVE *on the test data*. When test data is used to calculate POVE, Equation 7 does not hold; $f_i$ is formulated to minimize the error on

the training data, not the unseen test data. Under this formation, the POVE can actually be negative.

For example, consider the case of POVE for a naive of predictor, where the learned function $f$ just predicts the mean of the training labels. Assume we use leave one out cross validation (LOOCV). Our naive predictor always predicts:

$$\bar{y}_{-i} = \frac{1}{N-1}\sum_{j \neq i} y_j$$

Note that

$$\begin{aligned}\bar{y}_{-i} &= \frac{1}{N-1}\sum_{j \neq i} y_j \\ &= \frac{(N\bar{y} - y_i)}{N-1}\end{aligned}$$

where $N$ is the full size of the data set. For our naive predictor, POVE is

$$\begin{aligned}1 - \frac{\sum_i (y_i - \bar{y}_{-i})^2}{\sum_i (y_i - \bar{y})^2} &= 1 - \frac{\sum_i (y_i - \frac{(N\bar{y} - y_i)}{N-1})^2}{\sum_i (y_i - \bar{y})^2} \\ &= 1 - \frac{\sum_i (\frac{(N-1)y_i - N\bar{y} + y_i}{N-1})^2}{\sum_i (y_i - \bar{y})^2} \\ &= 1 - \frac{\sum_i (\frac{N(y_i - \bar{y})}{N-1})^2}{\sum_i (y_i - \bar{y})^2} \\ &= 1 - \frac{N^2}{(N-1)^2}\frac{\sum_i (y_i - \bar{y})^2}{\sum_i (y_i - \bar{y})^2} \\ &= 1 - \frac{N^2}{(N-1)^2}\end{aligned}$$

Two surprising points have emerged. Not only is the POVE negative, it is completely independent of the distribution of the data, and depends only on the size of the data set! In the experiments outlined here we perform leave two out cross validation. There is no such simplified solution for POVE and the naive predictor under LTOCV, but as N grows, the performance quickly converges to that of LOOCV. Figure 6 shows POVE of the naive predictor under LOOCV and the estimated POVE for LTOCV.

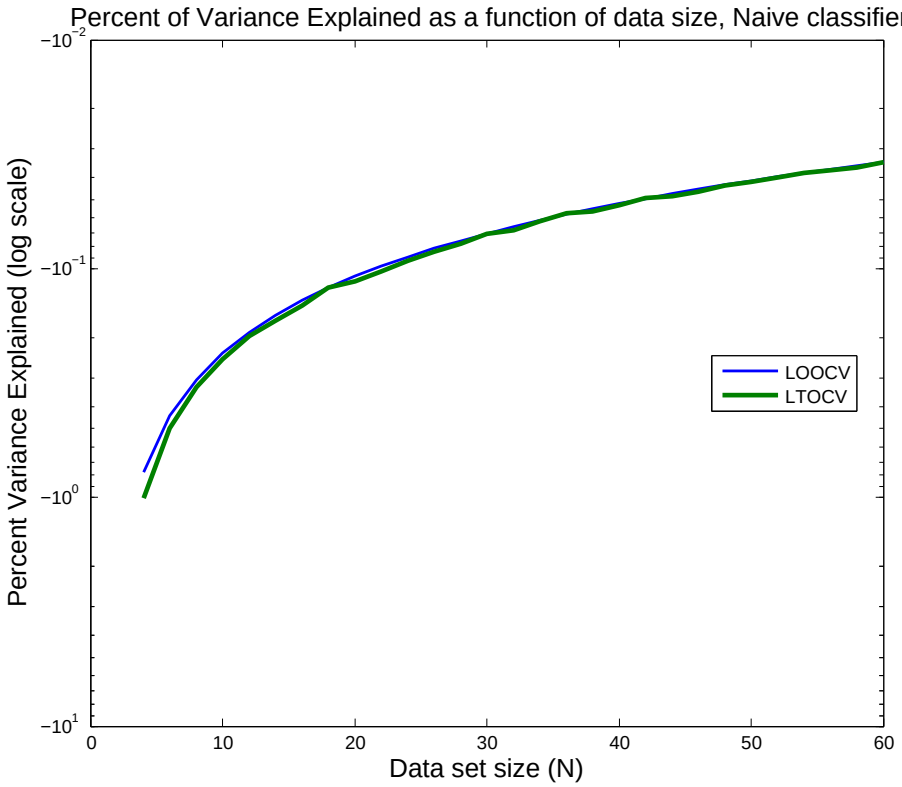


Figure 6: Percent of Variance Explained (POVE) for the naive predictor that simply predicts the mean of the training labels. POVE was evaluated under leave one out cross validation (LOOCV) and leave two out cross validation (LTOCV). Each point of the POVE for LTOCV was estimated using 500 samples drawn from a standard Normal and randomly selected cross validation folds.

## 4.2 False Discovery Rate

Since we are measuring the performance of many MEG features for many semantic features across many subjects, we must correct our performance results for multiple comparisons. For this we use a combination of Fisher's Method and the False Discovery Rate (FDR) method of Benjamini and Hochberg [2] (see also [22]).

Fisher's Method takes a group of p-values and combines them to create one statistical test.

$$X^2 = -2\sum_{i=1}^{s} \ln(p_i) \tag{9}$$

Where $s$ is the number of p-values combined to create the test statistic and the $p_i$ are the p-values. The resulting test statistic has a chi-squared distribution with $2 * s$ degrees of freedom. For our purposes, we will be combining $s = 9$ p-values from the 9 subjects to create one p-value per semantic feature/MEG feature pair.

Now we sort the p-values obtained from Fisher's Method:

$$P_{(1)} < P_{(2)} < \ldots P_{(m)}$$

where $m$ is the total number of p-values to be evaluated (one per semantic feature type). Then calculate

$$\ell_i = \frac{i\alpha}{C_m m}$$

where $i$ is the index $\{1 \ldots m\}$. If the p-values being tested are independent, $C_m = 1$. In our case, however, our p-values are not independent; many of the semantic features are correlated with each other, and some have no correlation. For this reason we must use $C_m = \sum_{i=1}^{m} 1/i$ to strengthen our bounds to cover arbitrary dependence amongst the $m$ tests. This is the strictest of corrections to the FDR procedure. See Theorem 1.3 of [3] for a complete derivation of this correction factor. Now define

$$R = max\left\{i : P_{(i)} < \ell_i\right\}$$

and identify $P_{(R)}$ as the BH threshold. We will reject the null hypothesis for all $P_{(i)}$ such that $P_{(i)} \leq P_{(R)}$.

### 4.3 Estimating the Null Distribution

In order to employ the method outlined above, we must be able to evaluate the probability of seeing some POVE value, given that the value was drawn from the null distribution. The null distribution should represent the case that there is no connection between the semantic features and the MEG data. To estimate the null distribution we perform 108 permutation tests in which we shuffle the labels of each of the 1200 MEG single trials, and use that relabeled data to create the MEG features. We then learn regressors with the same procedure outlined in Section 4, and calculate the POVE for the functions trained on permuted data. On average, the mean POVE for the null distributions is $-0.0311$. This is very close to the value calculated in Section 4.1 for the naive predictor and LOOCV: $N^2/(N-1)^2 = -0.0342$. The POVE values created from permuted data form an empirical PDF (EPDF).

Using the null EPDF, we can estimate the probability of a POVE value (that is, create a p-value) by interpolating between the observed points of the EPDF. We do this for all POVE values calculated from regressors trained on the non-permuted data. We sort the resulting p-values and correct them with the Benjamini-Hochberg method. We choose the standard cutoff of $\alpha = 0.05$. Because we are separately testing seven MEG feature types we use Bonferroni correction to adjust the level of the test. Bonferroni correction requires that the desired level be divided by the total number of tests performed, so we adjust our level to $\alpha = 0.05/7$.

## 5 Results

Table 2 shows the performance of all 9 subjects and 7 MEG features on the 2 vs 2 classification task. For permuted data, the performance on the 2 vs 2 task was no more than 51% across all subjects and all feature types. The Continuous Haar Wavelet Transform performs the best overall, though the Raw MEG feature performs better for subject 1. The Raw and Windowed Mean MEG features give are similar to each other in performance, with the Raw feature performing slightly better for some, but not all, subjects. The norm of the gradiometer is the next best for 2 vs. 2 performance. STFT Phase and the Windowed Slope rank fifth and sixth respectively, but have a performance difference of only 0.3%. The worst feature is STFT Power, giving 2 vs 2 performance of only 78.9%.

Table 2: Two vs Two accuracy (in percent) in for all 9 subjects and 7 feature types, as well as the average over all 9 subjects. Higher scores are better. All MEG features produced 2 vs 2 results that are significantly better than the null with $p = 0.05/7$ and p-values combined using Fisher's Method.

| MEG Feature | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | Mean |
|---|---|---|---|---|---|---|---|---|---|---|
| Raw | 90.7 | 92.7 | 89.3 | 96.7 | 91.3 | 92.0 | 88.7 | 84.0 | 93.3 | **91.0** |
| Grad Norm | 88.7 | 92.0 | 82.7 | 95.3 | 90.7 | 84.7 | 84.0 | 84.0 | 89.3 | **87.9** |
| W Mean | 88.0 | 91.3 | 86.7 | 97.3 | 89.3 | 93.3 | 90.0 | 82.0 | 93.3 | **90.1** |
| W Slope | 84.0 | 87.3 | 70.7 | 93.3 | 81.3 | 84.0 | 79.3 | 78.0 | 86.0 | **82.7** |
| Power | 78.0 | 80.7 | 71.3 | 86.7 | 88.0 | 72.0 | 84.7 | 70.7 | 78.0 | **78.9** |
| Phase | 79.3 | 87.3 | 73.3 | 92.0 | 88.7 | 79.3 | 85.3 | 76.0 | 86.0 | **83.0** |
| Haar | 89.3 | 97.3 | 90.0 | 98.7 | 94.0 | 94.0 | 91.3 | 89.3 | 95.3 | **93.3** |

Table 3: The percent of semantic features significantly decodable in 5 feature categories and across all 218 semantic features, as calculated across 9 subjects for each of the 7 MEG feature types. The number of semantic features in the feature categories are: Alive 44, Eating 5, Manipulable 6, Shelter 4, Size 8. Column All is the percentage of all the semantic features that are significant for the given MEG feature type.

| MEG feature | Alive | Eating | Manipulable | Shelter | Size | All |
|---|---|---|---|---|---|---|
| Raw | 81.8 % | 0.0 % | 100.0 % | 100.0 % | 100.0 % | 52.1 % |
| Grad Norm | 77.3 % | 0.0 % | 100.0 % | 75.0 % | 87.5 % | 44.7 % |
| W Mean | 79.5 % | 0.0 % | 100.0 % | 100.0 % | 87.5 % | 49.3 % |
| W Slope | 54.5 % | 0.0 % | 100.0 % | 25.0 % | 75.0 % | 28.6 % |
| Power | 47.7 % | 0.0 % | 66.7 % | 75.0 % | 62.5 % | 23.5 % |
| Phase | 72.7 % | 0.0 % | 100.0 % | 100.0 % | 87.5 % | 47.9 % |
| Haar | 88.6 % | 60.0 % | 100.0 % | 100.0 % | 100.0 % | 72.4 % |

Because there are 218 semantic features, it can be difficult to analyze the performance of a given MEG feature on the full set of semantic features. For this reason we have chosen a number of subsets of the semantic features and analyzed them in aggregate. Table 3 shows the percent of semantic features that are significantly decodable for several groups of features with a common theme. The groups (with number of semantic features per group in parenthesis) are Alive (44), Eating (5), Manipulable (6), Shelter (4), Size (8). Perhaps most striking is that semantic features related to eating can be reliably decoded only with the Continuous Haar MEG feature. This is surprising because past fMRI studies [10] have shown eating to be amongst the most decodable of features. However, that same study showed that several of the brain areas where eating was best decoded were in the inferior temporal gyrus, a brain area that is difficult to record from with MEG. Perhaps it is the robustness to noise of the Continuous wavelet transform that allows for this improved performance. Alternatively, it may be the combination of information from phase and signal magnitude that allows for the significant decoding of semantic features related to eating.

Overall, the performance of the Raw and Windowed Mean MEG feature are very similar, but for a few semantic feature categories (Alive and Size) the Raw MEG feature performs better. This implies that information within the 50ms window used to create the windowed mean is important for some semantic distinctions. The Euclidean Norm of Gradiometers and Windowed Slope MEG features perform below the Raw MEG feature in all cases except for the Manipulability feature category, where they are equal.

Across the board, the Phase MEG feature outperforms the Power MEG feature. Thus, the information about the semantics of an object lies not in the total energy in a frequency band, but in the "locking" of neurons firing at a particular rate in response to stimulus onset.

In all cases the Continuous Haar wavelet outperforms all other MEG features. In hindsight, this is not surprising as most of the other MEG can be recreated from the Continuous Haar Wavelet coefficients. For example, the Raw signal can be recreated from a linear combination the wavelet coefficients, and frequency and phase information can also be extracted. We will explore the significance of the Continuous Haar Wavelet's performance at the word level in the next section.

## 5.1 Rank Accuracy

The 2 vs 2 test results in high accuracy for most of the MEG features considered, possibly because the task is fairly easy. Even if the predictions for each of the two held out words are far from their true values, each predicted vector need only to be closer to the true vector than the alternate held out vector for the resulting assignment to be correct. Evaluating the predictions using a more difficult task could help us to better differentiate the performance of the MEG feature types. For this we turn to a new data set.

In addition to the 60 words for which we have MEG data and semantic features, we have semantic features (but no MEG data) for 940 other words. Given a predicted semantic feature vector, we can measure the cosine distance between the predicted vector and each of the 940 additional feature vectors, plus the true semantic feature vector. Now we are ranking 941 semantic feature vectors by their distance to the predicted semantic feature vector. We sort these distances and find the position of the true semantic feature vector for the $i$th held out word. Rank accuracy is then

$$\text{rank accuracy} = (1 - \frac{r_i}{W}) * 100 \tag{10}$$

where $r_i$ is the position of the true semantic feature vector in the list of sorted distances and $W = 941$ is the total length of the sorted list. Rank accuracy is equal to the percentage of semantic feature vectors that are further from the predicted vector than the true semantic feature vector. Under this schema, higher scores are better and a perfect score would be 100. Table 4 shows the median rank accuracy for each of the 9 subjects and 7 MEG feature types. On permuted data, the rank accuracy was never above 61% for any of the 7 MEG feature types. The mean rank across the 7 MEG feature types was 38.5%.

The Raw MEG feature and the Continuous Haar Wavelet feature give the best rank accuracy, but the Haar Wavelet has the maximum rank accuracy for all subjects. We can test wether the Continuous Haar Wavelet is indeed the better MEG feature type by recording the distance of each predicted vector to the true semantic feature vector. If there is no difference between the Raw and the Haar Wavelet MEG feature type, then we would expect the difference in the distance between the two predicted semantic feature vectors (one each from regressors trained on the Raw and Haar wavelet MEG features) and the actual semantic feature vectors to be normally distributed with mean 0. This analysis of difference in distance lends itself well to the paired t-test,

which confirms that the Haar wavelet does produce better predictions, with $p < 10^{-60}$. One might argue that the difference between the distances need not be normally distributed. To address this we can perform a Wilcoxon rank sum test to test wether the medians of the distances are equal. The Wilcoxon test rejects the null hypothesis that the medians are equal with $p = 7 * 10^{-7}$.

Table 4: Median rank accuracy (over 941 words, as described in Section 5.1) for all 9 subjects and 7 feature types. Higher scores are better. A paired t-test shows that the Haar wavelet transform produces predicted vectors with smaller distance to the true semantic feature vector than predictions based on the raw signal ($p < 10^{-60}$).

| MEG Feature | S1 | S2 | S3 | S4 | S5 | S6 | S7 | S8 | S9 | All |
|---|---|---|---|---|---|---|---|---|---|---|
| Raw | 90.3 | 93.0 | 88.2 | 95.1 | 90.8 | 86.9 | 86.8 | 87.2 | 89.9 | 90.1 |
| Grad Norm | 86.2 | 90.1 | 82.5 | 94.1 | 92.2 | 79.9 | 88.3 | 85.9 | 87.8 | 88.2 |
| W Mean | 89.2 | 91.9 | 88.7 | 95.5 | 90.8 | 87.5 | 89.3 | 86.2 | 91.9 | 90.1 |
| W Slope | 81.5 | 89.6 | 70.4 | 91.2 | 79.0 | 76.3 | 78.5 | 84.6 | 85.2 | 82.9 |
| Power | 81.2 | 81.0 | 54.4 | 84.2 | 79.1 | 63.4 | 73.2 | 65.7 | 65.9 | 74.1 |
| Phase | 63.3 | 82.6 | 54.5 | 87.5 | 84.5 | 60.6 | 82.3 | 65.1 | 71.4 | 72.8 |
| Haar | 95.1 | 96.0 | 90.1 | 97.3 | 92.2 | 90.7 | 89.0 | 88.9 | 96.0 | 93.3 |

## 5.2 Analysis of the Haar Wavelet Features

The Continuous Haar Wavelet MEG feature is the top performing amongst the features we have evaluated here. But there are several dimensions to this feature: frequency, time and sensor space. Which among these dimensions carry the most decoding power? Figure 7 shows the rank accuracy as a function of time and frequency for the Continuous Haar Wavelet MEG feature. From this plot we can see that the majority of the power is focused between 200 and 300 ms after stimulus onset, and in the delta, theta and alpha frequency bands.

What areas of the brain contribute the most meaningful features? For this we separate out the 306 sensors into those approximately covering each of the four lobes of the brain: Temporal, Parietal, Frontal and Occipital. The best performing lobe is the occipital lobe, followed closely by the temporal lobes. Parietal gives the third best rank accuracy, and the frontal gives the

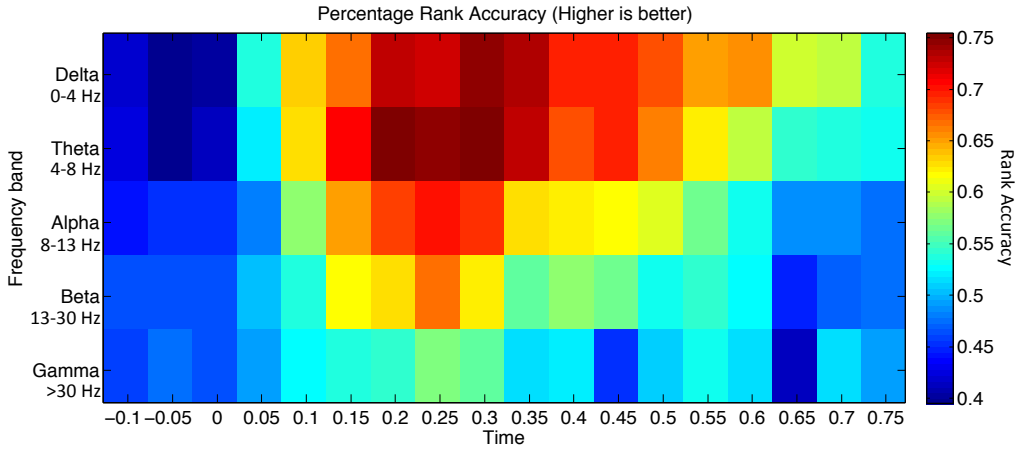


Figure 7: The rank accuracy for different scales and time segments of the coefficients produced by the Continuous Haar Wavelet Transform. Delta corresponds to frequencies 0-4Hz, theta 4-8Hz, alpha 8-13Hz, beta 13-30Hz, gamma >30 Hz. Our MEG data was lowpass filtered to exclude frequencies above 60Hz, so the gamma frequency band was truncated at 60 Hz. The plot shows that the most useful coefficients are focused between 200 and 300 ms after stimulus onset, and between the delta to alpha frequency bands.

lowest. All pairwise differences between the lobes are significant with $p = 0.05/(4*3/2)$ where $(4*3/2)$ is the number of pairwise tests performed. It is surprising that the occipital lobe gives such good word prediction, especially since we are decomposing the words into semantic feature vectors before training classifiers. High performance using features derived from activity in the occipital lobe may have something to do with the visualization of words or be a side-effect of working in sensor-space (which may attribute things to the visual lobe though the sensors are actually covering a nearby region).

# 6 Conclusion

We have explored the utility of several transformation of the MEG signal with respect to decoding the semantic properties of a word a subject is reading. We have shown that the Continuous Haar Wavelet Transform is by far the best MEG feature of those considered here. We propose that this is because the Wavelet transform contains most of the information available in other

Table 5: The average rank accuracy over 941 words of predictors trained with the Haar wavelet feature, but only a subset of the MEG sensors covering the four lobes of the brain. A paired t-test with Bonferroni correction shows that the difference in rank accuracy between all pairs of lobes is significant.

| Brain Region | Average Rank Accuracy |
|---|---|
| Temporal | 79.74 |
| Parietal | 72.76 |
| Frontal | 65.95 |
| Occipital | 80.80 |

feature types and thus represents the best of all worlds. We have shown that the wavelet transform performs well for both Percent of Variance Explained and for rank accuracy on a large list of words. Though the wavelet transformation results in many more MEG features (64x more features in this case), we think the improvement in decoding accuracy is worth the extra computational expense.

In the future we would like extend this work to incorporate multivariate feature types (those that combine sensors together) and to train larger classifiers that incorporate more than one feature type. In addition, we would like to use the wavelet transform to perform single trial analysis, where only one trial per word is available. We hope that the robustness of the wavelet transform to noise will prove advantageous in this challenging low SNR task. We also plan to extend this work to combinations of words, such as adjective noun pairs. We expect that the increases in accuracy found here will extent to multi-word paradigms as well.

# References

[1] Danielle S Bassett, Andreas Meyer-Lindenberg, Sophie Achard, Thomas Duke, and Edward Bullmore. Adaptive reconfiguration of fractal small-world human brain functional networks. *Proceedings of the National Academy of Sciences of the United States of America*, 103(51):19518–23, December 2006.

[2] Yoav Benjamini and Yosef Hochberg. Controlling the false discovery rate: a practical and powerful approach to multiple testing. *Journal of the Royal Statistical Society. Series B*, 57(1):289–300, 1995.

[3] Yoav Benjamini and Daniel Yekutieli. The control of the false discovery rate in multiple testing under dependency. *Annals of statistics*, 29(4):1165–1188, 2001.

[4] Alexander M Chan, Eric Halgren, Ksenija Marinkovic, and Sydney S Cash. Decoding word and category-specific spatiotemporal representations from MEG and EEG. *NeuroImage*, 54(4):3028–39, February 2011.

[5] Avniel Singh Ghuman, Jonathan R McDaniel, and Alex Martin. A wavelet-based method for measuring the oscillatory dynamics of resting-state functional connectivity in MEG. *NeuroImage*, 56(1):69–77, January 2011.

[6] Gene H. Golub, Michael Heath, and Grace Wahba. Generalized cross-validation as a method for choosing a good ridge parameter. *Technometrics*, 21(2):215–223, 1979.

[7] Peter Hansen, Morten Kringelbach, and Riitta Salmelin. *MEG: An Introduction to Methods*. Oxford University Press, USA, 2010.

[8] Trevor Hastie, Robert Tibshirani, and Jerome Friedman. *The Elements of Statistical Learning*. Springer New York, 2001.

[9] Ole Jensen, Jochen Kaiser, and Jean-Philippe Lachaux. Human gamma-frequency oscillations associated with attention and memory. *Trends in neurosciences*, 30(7):317–24, July 2007.

[10] Marcel Adam Just, Vladimir L Cherkassky, Sandesh Aryal, and Tom M Mitchell. A neurosemantic theory of concrete noun representation based on the underlying brain codes. *PloS one*, 5(1):e8622, January 2010.

[11] Huan Luo and David Poeppel. Phase patterns of neuronal responses reliably discriminate speech in human auditory cortex. *Neuron*, 54(6):1001–10, June 2007.

[12] Mark Palatucci, Geoffrey Hinton, Dean Pomerleau, and Tom M Mitchell. Zero-Shot Learning with Semantic Output Codes. *Advances in Neural Information Processing Systems*, 22:1410–1418, 2009.

[13] G Pfurtscheller and F H Lopes da Silva. Event-related EEG/MEG synchronization and desynchronization: basic principles. *Clinical neurophysiology : official journal of the International Federation of Clinical Neurophysiology*, 110(11):1842–57, November 1999.

[14] David C Plaut and James L McClelland. Locating object knowledge in the brain: comment on Bowers's (2009) attempt to revive the grandmother cell hypothesis. *Psychological review*, 117(1):284–8, January 2010.

[15] F Pulvermüller, N Birbaumer, W Lutzenberger, and B Mohr. High-frequency brain activity: its possible role in attention, perception and language processing. *Progress in neurobiology*, 52(5):427–45, August 1997.

[16] Riitta Salmelin and Jan Kujala. Neural representation of language: activation versus long-range connectivity. *Trends in cognitive sciences*, 10(11):519–25, November 2006.

[17] Jesse Sherwood and Reza Derakhshani. On Classifiability of Wavelet Features for EEG-Based Brain-computer Interfaces. In *International Joint conference on Neural Networks*, pages 1–8, 2009.

[18] S Taulu and J Simola. Spatiotemporal signal space separation method for rejecting nearby interference in MEG measurements. *Physics in Medicine and Biology*, 51:1–10, 2006.

[19] Samu Taulu and Riitta Hari. Removal of magnetoencephalographic artifacts with temporal signal-space separation: demonstration with single-trial auditory-evoked responses. *Human brain mapping*, 30(5):1524–34, May 2009.

[20] Samu Taulu, Matti Kajola, and Juha Simola. The Signal Space Separation method. *ArXiv Physics*, 2004.

[21] M A Uusitalo and R J Ilmoniemi. Signal-space projection method for separating MEG or EEG into components. *Medical & biological engineering & computing*, 35(2):135–40, March 1997.

[22] Larry Wasserman. *All of Statistics*. Springer New York, 2004.

Table 6: Variable names and definitions as used in this report.

| Variable | Definition |
|---|---|
| $m$ | number of semantic features (218) |
| $s$ | number of subjects |
| $T$ | total number of time points |
| $X$ | MEG time series ($306 \times T$) |
| $X_{i,j}$ | value for sensor i and time point t in the MEG time series |
| $R$ | number of single trials collected per word (20) |

# A MEG Data Acquisition

All subjects gave their written informed consent approved by the University of Pittsburgh (protocol PRO09030355) and Carnegie Mellon (protocol HS09-343) Institutional Review Boards. MEG data were recorded using an Elekta Neuromag device (Elekta Oy). The data was acquired at 1 kHz, high-pass filtered at 0.1 Hz and low-pass filtered at 330 Hz. Eye movements (horizontal and vertical eye movements as well as blinks) were monitored by recording the differential activity of muscles above, below, and beside the eyes. At the beginning of each session we recorded the position of the subject's head with four head position indicator (HPI) coils placed on the subject$\tilde{\text{O}}$s scalp. The HPI coils, along with three cardinal points (nasion, left and right pre-auricular), were digitized into the system.

The data were preprocessed using the Signal Space Separation method (SSS) [20, 18] and temporal extension of SSS (tSSS) [19] to remove artifacts and noise unrelated to brain activity. In addition, we used tSSS to realign the head position measured at the beginning of each block to a common location. The MEG signal was then low-pass filtered to 50 Hz to remove the contributions of line noise and down-sampled to 200 Hz. The Signal Space Projection method (SSP) [21] was then used to remove signal contamination by eye blinks or movements, as well as MEG sensor malfunctions or other artifacts. Each MEG repetition starts 260 ms before stimulus onset, and ends 1440 ms after stimulus onset, for a total of 1.7 seconds and 340 time points of data per sample. MEG recordings are known to drift with time, so we corrected our data by subtracting the mean signal amplitude during the 200ms before stimulus onset, for each sensor/repetition pair. Because the

magnitude of the MEG signal is very small, we multiplied the signal by $10^{12}$ to avoid numerical precision problems.