Temporally Dependent Mappings Between fMRI Responses and Natural Language Descriptions of Natural Stimuli

Kiran N. Vodrahalli

A Thesis
Presented to the Faculty
of Princeton University
in Candidacy for the Degree
of Master of Science in Engineering

Recommended for Acceptance
by the Department of
Computer Science
Advisers: Sanjeev Arora and Kenneth A. Norman

June 2017

Abstract

Several research groups have shown how to correlate fMRI responses to the meanings of presented stimuli. This paper presents new methods for doing so when only a natural language annotation is available as the description of the stimulus. We study fMRI data gathered from subjects watching an episode of BBCs Sherlock [4], and learn bidirectional mappings between fMRI responses and natural language representations. We show how to leverage data from multiple subjects watching the same movie to improve the accuracy of the mappings, allowing us to succeed at a scene classification task with 72% accuracy (random guessing would give 4%) and at a scene ranking task with average rank in the top 4% (random guessing would give 50%). The key ingredients are (a) the use of the Shared Response Model (SRM) and its variant SRM-ICA [5, 24] to aggregate fMRI data from multiple subjects, both of which are shown to be superior to standard PCA in producing low-dimensional representations for the tasks in this paper; (b) a sentence embedding technique adapted from the natural language processing (NLP) literature [3] that produces semantic vector representation of the annotations; (c) interpretably using previous timestep information in the featurization of the predictor data.

Acknowledgements

First and foremost, I would like to thank my advisors Sanjeev Arora and Ken Norman for being mentors to me over the past two years. I would also like to thank all my collaborators on the work documented in this thesis: Cameron Chen, Yingyu Liang, Cathy Chen, Viola Mocz, Chris Baldassano, Janice Chen, Christopher Honey, Uri Hasson and Peter Ramadge. I would also like to thank Esther Yong for helping create the dataset used throughout this work. The dataset is online [4] and the code used in this paper will be made available on GitHub. Additionally, we note that we used http://brainiak.org/ for some of the implementations of algorithms used in this paper. This work was funded by a grant from the Intel Corporation, NIMH R01MH112357 awarded to U. Hasson and K. Norman; NIH grants R01-MH094480 and 2T32MH065214-11; NSF grants CCF-1527371, DMS-1317308, Simons Investigator Award, Simons Collaboration Grant, and ONRN00014- 16-1-2329 awarded to S. Arora, and NSERC Discovery Grant RGPIN 2014-04465 awarded to C. Honey. P. H. Chen was supported by a Google Fellowship.

I am indebted to the teachers I have had at Princeton: I would like to thank Michael Damron, whose introductory analysis class convinced me to do math, the path which led me to theoretical machine learning. I would also like to thank Elad Hazan, Barbara Engelhardt, Samory Kpotufe, Assaf Naor, Emmanuel Abbé, Ramon van Handel, and Peter and Sanjeev again for teaching me a large portion of what I know about theory and machine learning.

I would also like to thank my friends: Thank you to Amy, for being my best friend and supporting me through all the ups and downs of research. Thank you to the ML Lounge crew (Misha, Nikunj, Ghassen, Xinyi) for providing interesting conversations about machine learning as well as hilarious stories, and for singing along with Alexa. Thanks to Alexa as well for helping out whenever I was feeling a little bit Genghis Khan, or whenever I was going slightly mad. Thanks to Misha,

Nikunj, and Ghassen again for all of the bromance (especially Ghassen). Thank you to Holden Lee, Karan Singh, Naman Agarwal, Brian Bullins, Cyril Zhang, Tengyu Ma, Yingyu Liang, Andrej Risteski, Zeyuan Allen-Zhu, and Yang Yuan for the theory-and-learning related conversations we've had.

Finally, I'd like to thank my family for supporting me through college and providing me opportunities to learn from the very beginning of my life.

Contents

List of Tables

List of Figures

Chapter 1

Introduction

Recent work has provided convincing evidence that fMRI readings from human subjects can be related to semantics of presented stimuli. Such experiments consist of finding (1) low-dimensional representations of the fMRI signals, and (2) low-dimensional semantic representations of the external stimulus. These tasks often build upon work in machine learning.

1.1 Background

The earliest work concerned simple settings with carefully controlled stimuli, such as subjects being presented (visually or auditorily) with one of a set of carefully selected words [17]. The semantic representation of a word was computed using word embeddings, a tool from natural language processing [7] that represents each word as a point in a k-dimensional meaning space. This work was extended [18, 19] to perform "brain reading", using fMRI readings and a popular text-analysis tool called topic modeling to reconstruct word clouds from brain activity evoked by a word/concept stimulus.

The next obvious step in this research program is to understand fMRI readings collected from subjects as they process more complex stimuli such as movies. In

such settings it is not clear how to represent the semantics of the stimulus, since a multitude of signals (auditory as well as visual) are presented within a short time interval. Ideally, this mapping between fMRI and stimuli should be meaningful across different human subjects, so that the accuracy of matching the two should improve by using data from multiple subjects. One approach to solving this task was presented in [13], which studied fMRI responses to a natural movie stimulus. In this case, the movie stimulus was represented with a feature space of 1705 distinct nouns and verbs. A subsequent study [12] examined fMRI responses to audio stories, and departed from the previous work by applying distributional embeddings to featurize the dialog and predict voxel activation. The goal in these papers was to derive a semantic word map for the voxels of the brain. Another paper [22] gathered fMRI data from subjects reading a story, and used unweighted averages of distributional embeddings to featurize sentences for predicting voxel activity.

1.2 Problem Setting and Goals

In this paper, we study the Sherlock fMRI dataset [4], which consists of fMRI recordings of 16 people watching the British television program "Sherlock" for 50 minutes broken into 1973 TRs, where each TR is 1.5 seconds of film. As a proxy for the semantics of the movie, we use externally annotated English text scene annotations of the program (average annotation length 15 words per TR). We examine brain data from predefined regions of interest (ROIs) in the brain, and separately analyze each one. In particular, we examine the default mode network (DMN), dorsal and ventral language areas, the occipital lobe, and a 26000-voxel mask containing voxels with high intersubject correlation across the whole brain. We seek to determine whether various modifications to fMRI and text featurization as well as the usage of previous timepoint information help to improve bidirectional mappings between fMRI data and

semantic meaning vectors. In particular, we examine the effects of three featurization methods for fMRI and text data: *Low-dimensional shared fMRI representation* across subjects, *weighted semantic embeddings* of text annotations, and using *previous timepoints* in the performance of linear maps between people.

Aggregating fMRI responses across subjects. In prior work, combining fMRI response data from multiple subjects is often solved by averaging, anatomical alignment and smoothing, or latent multivariate feature modeling [22, 6, 12]. Further work concludes that high-level representations of content from movies are shared across people and that there can be considerable de-noising benefits from averaging across people [4]. Another recent paper [5] introduced the Shared Response Model (SRM), an algorithm that stems from previous work on hyperalignment [10]. The SRM in [5] optimizes the objective $\sum_{i=1}^{n} \|X_i - W_i S\|_F$ for a low-dimensional shared space S and orthogonal-column subject specific maps W_i, and can be thought of as a multi-subject extension of PCA. Simultaneously reducing dimensionality across subjects outperforms other averaging approaches at matching up specific timepoints in a movie across subjects.

Semantic representation of stimulus. To find semantic representations of English annotations, it is natural to draw upon related work in natural language processing. One common approach involves word embeddings created by using co-occurrence information in a large corpus like Wikipedia. A simple technique for representing longer pieces of text is to average the vectors for the individual words [22]. Recently, this simplistic idea has been extended in natural language processing by using recurrent neural nets [15] or by modifying the original model for learning word vectors to learn word sequence chunks (for instance, paragraphs) directly from the text [16]. These more powerful methods have the drawback of requiring large corpora, making them unusable in our current setting where we only have 1973 brief text annotations. Very recently, [3] suggested a simpler method for this task that

requires no additional information beyond the existing word embeddings, yet beats these more complicated methods in standard natural language tasks. We adapt this method to construct *annotation embeddings* using weighted combinations of the vector representations for the words in each annotation. One of our key results is that this new embedding significantly outperforms unweighted averaging of word vectors.

Using previous timestep information. A movie stimulus naturally breaks up into multi-timestep scenes that occur at different timepoints. Thus, at any given timepoint, there may be a window of previous timesteps that are part of the current scene and thus are relevant to understanding the current time point in both fMRI and Text space. We would like to incorporate this past information shared within scenes in order to learn better maps between fMRI and Text. Other models [12, 22] incorporate past information by modeling the hemodynamic response function (HRF) that describes the fMRI BOLD response to a stimulus. However, this approach focuses on small timescales, and only accounts for the delayed and temporally-smeared BOLD response rather than attempting to aggregate scene information. Our approach is to first approximate the HRF delay with a simple one-time shift of 4.5 seconds, and to then incorporate longer time-scales into our model by including in the featurization a k-sized window of previous timesteps, where k is varied from 0 to 30 (these numbers correspond to $0 - 45$ seconds).

To evaluate the effect of each of these featurization methods, we use linear maps to relate the fMRI signal to the representation of the semantic content, using only the first half of the movie. These maps are validated with two experiments: scene classification and scene ranking. We divide up the second half of the movie into 25 uniformly-sized chunks. *Scene classification* is the task of using correlation to match predicted intervals of fMRI or semantic activity with the ground truth, and reporting the percentage of the time that the match is perfect. Since there are 25 intervals, random chance performance at this task is 4%. *Scene ranking* is the same task, except

we measure the average rank of the correct answer: Random chance performance here is 50%. For a visual summary of the setup, see Figure 1.1. These experiments are executed with the fMRI → Text maps (given fMRI data, predict text annotations) as well as the Text → fMRI maps (give text annotations, predict fMRI data).

1.3 Main results

Our main results are (i) showing that fMRI responses from multiple individuals can be effectively combined using SRM to improve the matching accuracy (1.3× average improvement over our baseline, the average PCA representation) between the fMRI and the text annotation (Table 4.1, Figures 4.2, 4.3), (ii) demonstrating that a method for combining word vectors into annotation vectors via a suitable weighting [3] for averaging word vectors on average improves 1.2× over unweighted averaging (Table 4.1, Figures 4.2, 4.3), and (iii) finding that appropriate inclusion of information from previous time steps yields as much as a 5.3× improvement (on average, 1.8×) in tasks measuring the performance of mapping from fMRI to Text (see Figure 4.2, Dorsal Language ROI). There are diminishing returns after a certain point to including more time steps: The optimal number seems to be around $5 - 8$ previous time steps. For the Text → fMRI task, using previous time steps decreases performance.

We also report the top performances for each task. For the fMRI → Text task, our top scene classification performance is 72% accuracy, meaning that for 72% of the time intervals we examine, our predicted annotation representation correlates the most with the true annotation representation for that time interval (see Figure 4.1, Whole Brain ROI). Notably, this result improves considerably over the random guessing rate of 4%. The corresponding scene ranking performance is 96%, meaning that on average, the rank of the true annotation representation is within the top 4% when sorted by correlation with the predicted annotation representation.

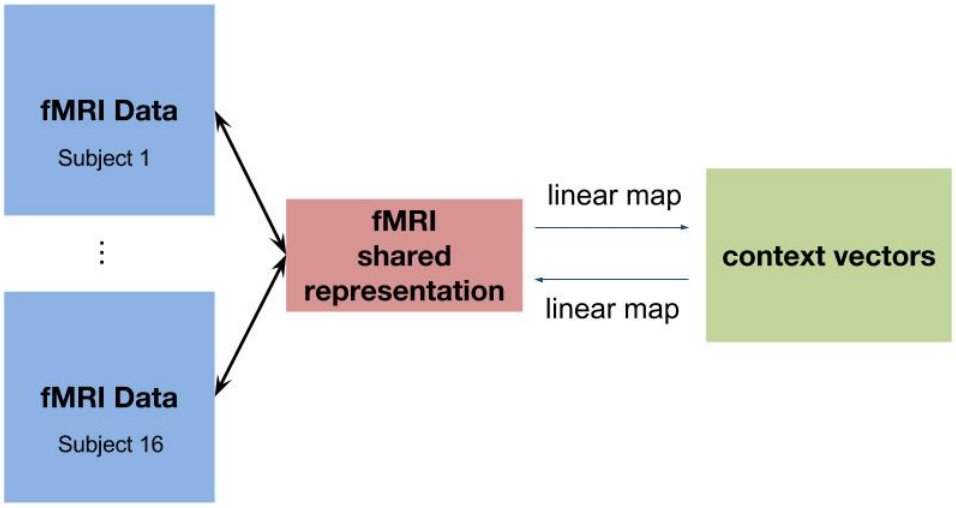

Figure 1.1: Summary of Experimental Setup: We learn a shared response for the brain activity of 16 different subjects watching BBC's Sherlock, construct semantic featurizations for associated semantic annotations, and learn bidirectional linear maps between the two data modes.

The Text $\rightarrow$ fMRI task had worse results. The top scene classification performance for Text $\rightarrow$ fMRI is 56% accuracy, and the corresponding scene ranking accuracy is 91% (see Figure 4.1, DMN-A ROI).

Chapter 2

Methods

2.1 Preprocessing the Dataset

Before performing any analysis, the fMRI data are preprocessed and standardized using the techniques described in [4]. Then, we identify six distinct brain regions of interest (ROIs) that we treat completely separately. That is, we first apply ROI masks to the whole-brain data and then learn SRM-representations for each of these ROIs separately. We use the ROIs for the default mode network (DMN-A, DMN-B) and the ROIs for the ventral and dorsal language areas identified in [21]. Methodology for finding the default mode network relies on intersubject functional correlation (ISFC), a technique first introduced by [9]. The central idea is that natural stimuli (like movies) evoke reliable, time-dependent activity across a variety of brain networks. For more details, see Figure 2.1. We are interested in the DMN ROIs in particular since prior work has demonstrated that these regions play a crucial role in tracking the narrative in settings such as watching movies or reading stories [9, 8, 11, 20, 1, 21, 23]. The "Whole Brain" ROI is a 26000-voxel mask of the brain that highlights voxels that have intersubject correlation > 0.2 on the data, and the Occipital Lobe ROI is defined

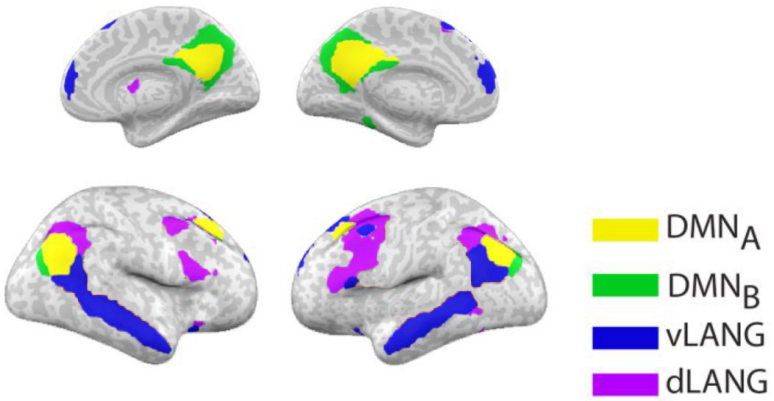

Figure 2.1: Visualization of the DMN and Ventral/Dorsal Language Area ROIs [21]: Here, we display four of the regions of interest on a brain map. These masks were collected on the Pie Man dataset [21], then fit to a standard anatomical brain (MNI152), and interpolated to 3-mm isotropic voxels [21]. In order to define the DMN-A and DMN-B regions, as well as the Ventral and Dorsal language area regions, the intersubject functional correlation matrix [9] was calculated from the fMRI data of 36 subjects collected while they were listening to stories [21]. Then, k-means clustering was applied to find the networks. The DMN-A and DMN-B networks were identified by comparing the resultant clusters to the DMN ROIs derived via thresholding the functional correlation between the posterior cingulate (identified by literature) and the rest of the brain for the fMRI data of 36 subjects during resting state [21]. The Ventral and Dorsal language areas were identified by comparing the clusters to previous results in the literature [21].

from the MNI Structural Atlas in FSL (https://fsl.fmrib.ox.ac.uk/fsl/fslwiki/Atlases). We include these ROIs for holistic comparison across the whole brain.

We also truncate the first three TRs of fMRI data and the last three TRs of semantic annotation data. This operation effectively aligns the fMRI and semantic data under the assumption that there is a 4.5 second delay between the onset of the stimulus and the BOLD response signal.

2.2 Constructing Semantic Embeddings

In order to represent words, we take advantage of the distributional properties of words in a large corpus - namely, English Wikipedia. We train word embeddings as described in [2], which perform on par with other standard word embedding techniques

like GloVe and Word2Vec [2]. Now, we diverge from the prior work by calculating and applying a domain specific re-centering of the embeddings. After creating an embedding for each word in the vocabulary of the Sherlock annotations, we calculate the top principal component of all word embeddings in the vocabulary. We then scale the normalized top principal component by the average Euclidean norm of a word embedding in the Sherlock vocabulary. This vector represents a kind of average topic for the Sherlock vocabulary. Since we would like our word embeddings to be discriminative within this average topic, we algebraically subtract out this component. We can view this step as finding a translation operation that moves the word embeddings away from the region of semantic space that is close to generic words in the Sherlock annotation corpus. Finally, in order to experiment with different dimensional word embeddings, we employ random dimension reduction, which is justified by the well-known Johnson-Lindenstrauss lemma. We check dimensions $20, 50, 100$, and 300. We achieve the best results using the 100-dimensional embeddings.

The central assumption in [2] is the probability model for a word w in a vocabulary V given a context c, where the context represents a small window of words in the corpus. This model is given by $\mathcal{P}[w|c] = \frac{1}{Z_c}\exp(v_w^T c)$ where v_w represents the vector for a given word and Z_c is a term that normalizes the distribution. The idea is that the context vector c represents the subject matter of the text at a given point in time.

Using this assumption and a few others, the word vector learning problem is phrased in [2] as the *squared-norm objective*:

$$\min_{\{v_w\}_{w\in V}, C} \sum_{w_1,w_2} X_{w_1,w_2} \left(\log(X_{w_1,w_2}) - \|v_{w_1} + v_{w_2}\|_2^2 - C\right)^2$$

where C is a bias term, X is the co-occurrence count matrix between single words in a small window of text (fixed at ≈ 5 words) and v_w are the word vectors we are trying to learn. This objective can be optimized with gradient descent. For a full

treatment of the theoretical properties of the word vectors and the derivation of the squared-norm objective, see [2].

For every 1.5-second time-point in our Sherlock movie, annotators were asked to provide a natural description of what is happening in the movie: actions, dialog, and so on. This annotation is typically a few sentences long, and contains around 15 words on average. We can think of each annotation as the current context of the movie narrative. The log-linear probability model of [2] for words given context c implies that the maximum likelihood estimator of the context is simply the average of all words in the annotation. (This formulation is a theoretical justification for a standard rule of thumb in natural language processing for representing the sense of a small piece of text by the average of the embeddings for the words in the text). We will call these representations the **unweighted** annotation vectors.

However, one imagines that not all words in the annotation are equally important, and that a better representation might be possible by taking this idea into account. This approach has been studied in various neural network frameworks [15]; however, applying these kinds of models requires a large annotation corpus, while we only have 1973 15-word annotations. A recent paper [3] suggests a principled approach for computing a representation of a small piece of text. The intuition from [3] is that words that occur with much greater frequency in the original corpus may inherently contain less information, since these words are in some sense uniform with respect to the whole word distribution. Therefore, more frequent words should be weighted less. The paper [3] modifies the above language generation model as follows: For a word w given context c, the probability of a word w given context c is

$$\mathcal{P}\left[w|c\right] = \alpha \mathcal{P}\left[w\right] + (1-\alpha)\frac{\exp(v_w^T c)}{Z_c} \tag{2.1}$$

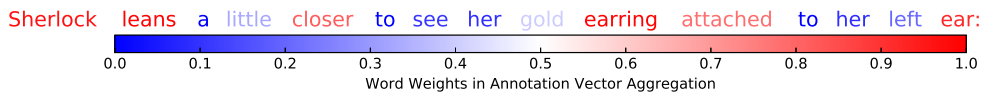

Figure 2.2: Visualization of Semantic Annotation Vector Weightings: We display an example sentence from the Sherlock annotations, where we have colored important words red, and unimportant words blue. Brighter red means more important, and darker blue means less important.

where Z_c normalizes the distribution and $\alpha \in [0, 1]$. We can think of this model as a weighted sum of the probability of a word w appearing not conditioned on the context c and the probability of a word w appearing conditioned on the context c.

The revised estimate of the context vector c in this modified objective is

$$v_{\text{annotation}} = \sum_{\text{word} \in \text{annotation}} \frac{\beta}{\beta + p_{\text{word}}} \cdot v_{\text{word}} \tag{2.2}$$

where $\beta := \frac{1-\alpha}{\alpha Z}$. Typically, we choose α such that $\beta \approx 10^{-4}$. These representations are called the **smooth inverse frequency (SIF)** annotation vectors, or **weighted** annotation vectors. Figure 2.2 depicts a example sentence with the respective word weights colored according to importance in the sentence embedding.

Using either the unweighted or weighted approach will produce one annotation vector for each of our T time steps. On the training portion of the data (the first half of the movie), we calculate an average annotation vector and subtract it from all data. Here, we assume that the average annotation vector is invariant, which turns out to be a good assumption.

2.3 Shared Response Models for Multi-Subject fMRI

The Shared Response Model (SRM) [5] is an unsupervised probabilistic latent variable model for multi-subject fMRI data under a time-synchronized stimulus. From each

subject's fMRI view of the movie, SRM learns projections to a shared space that captures semantic aspects of the fMRI response.

Specifically, SRM learns N maps W_i with orthogonal columns such that $\|X_i - W_i S\|_F$ is minimized over $\{W_i\}_{i=1}^N, S$, where $X_i \in \mathbb{R}^{v \times T}$ is the i^{th} subject's fMRI response (v voxels by T repetition times) and $S \in \mathbb{R}^{k \times T}$ is a feature time-series in a k-dimensional shared space. In this paper, $k = 20$ since low-rank SVD with 20 dimensions captures 90% of the variance of the original fMRI matrices [4]. We also experimented with using $k = 50, 80, 100, 1000$, but the results barely varied from using $k = 20$ dimensions. Note that, for testing, the learned W_i allow us to project unseen fMRI data into the shared space via $W_i^T X_i^{\text{test}}$ since W_i has orthogonal columns.

We also examine a variant of SRM called SRM-ICA [24] that modifies the SRM algorithm with an independent components analysis (ICA) objective. ICA is an unsupervised learning technique that identifies independent signals from a mixture by looking for rotations of the data that produce non-Gaussian signals. SRM-ICA brings this approach to learning a shared space: While in SRM we alternated by solving for W_i by minimizing $\|X_i - W_i S\|_F$ and updating S with the average of $W_i^T X_i$, we change the objective we use to update each W_i to an ICA objective: Maximizing the non-Gaussianity of the shared response $S = \frac{1}{n} \sum_{i=1}^n W_i^+ X_i$, individually with respect to each (X_i, W_i) pair.

In our experiments, we compare average SRM and SRM-ICA projections ($\frac{1}{N} \sum_{i=1}^N W_i^T X_i^{\text{test}}$) against the baseline average principal components analysis (PCA) projections. PCA is a standard linear dimensionality reduction technique that finds an optimal (in Frobenius norm) orthogonal projection of the data onto a low-dimensional subspace.

2.4 Learning Linear Maps

Our approach to predicting semantic annotation vectors from fMRI vectors and vice versa is simply linear regression with two kinds of regularization. Letting $X \in \mathbb{R}^{v \times T}$ represent the fMRI data matrix (either SRM, SRM-ICA, or PCA) for a specific ROI and $Y \in \mathbb{R}^{100 \times T}$ represent the annotation vectors, our main approach is given by solving the Procrustes problem $\min_\Omega \|Y - \Omega X\|_2^2$ with orthogonal columns constraint $\Omega^T \Omega = I_{v \times v}$. Thus, we learn a matrix $\Omega \in \mathbb{R}^{100 \times v}$ as a map from $X \to Y$, decoding fMRI vectors into semantic space. Our other approach is given by the ridge regression problem $\min_{\omega_j} \|y_j - \omega_j^T X\|_2^2 + \|\omega_j\|_2^2$ where $j \in [1, 100]$ for each word vector dimension. Putting the ω_j together forms $\Omega \in \mathbb{R}^{100 \times v}$ as before, with the orthogonality constraint replaced by a row-wise ℓ_2-norm regularization.

2.5 Adding Previous Timesteps

One could augment the fMRI and annotation vectors using past time steps by finding a complicated combination of the features at each time step, resulting in a representation with the same number of dimensions. For now, we sidestep the complexity of this task by simply concatenating k previous vectors to the predictor vector at each time step (TR) before learning mappings as before. A potential downside to this approach is that we linearly increase the dimensionality with k, which can be intractable for large k. However, this approach allows every predictor feature at every timepoint to have its own weight in the linear map, creating a powerful model. Thus, in the fMRI $\to$ Text case, we stacked the k previous fMRI vectors onto each fMRI vector, and did not modify the textual annotation vectors. In the Text $\to$ fMRI case, we stacked k previous text annotation vectors and left the fMRI vectors unmodified. When previous time steps do not exist, we append an all-zeros vector instead. We can think of the modified representations as capturing a notion of the dynamics occurring

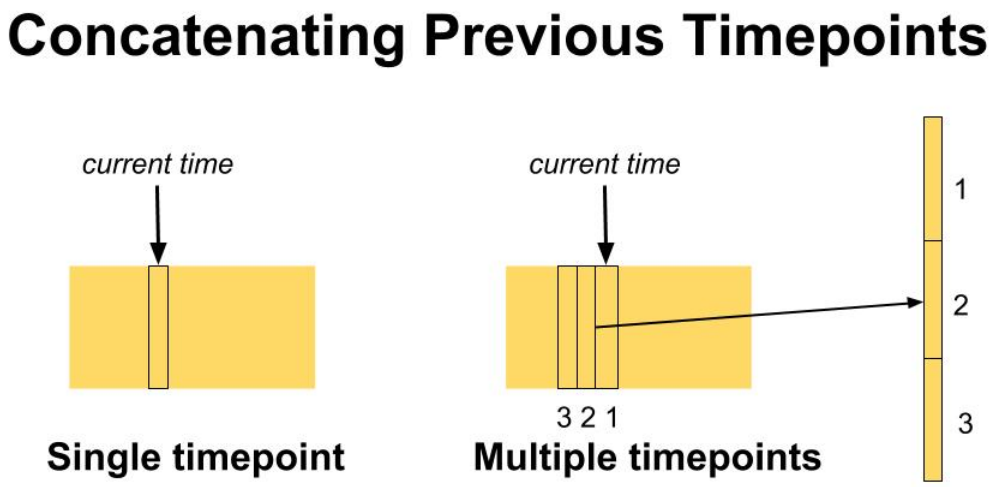

Figure 2.3: Visualizing Concatenation: We visualize what the single timestep case looks like compared to a case where we use the previous two timesteps in our featurization as well. The latter case results in a more complicated model, since one of the dimensions of our linear map triples in size.

over an interval of $1.5(k+1)$ (TR length $\times$ total number time points) seconds. In this paper, we tried $k = 1$ to 9 in steps of 1, and then $k = 10$ to 30 in steps of 5. See Figure 2.3 for a visualization.

2.6 Experiment Descriptions

First, we divide our 1973 TRs into 50 uniformly-sized chunks of time, the first 25 of which are our training data and the latter 25 of which are our testing data. We learn maps both from fMRI to text annotations and from text annotations to fMRI on the training data. From now on, we refer to **fMRI** $\rightarrow$ **Text** experiments as those which take an fMRI representation as input and attempt to predict a semantic annotation vector representation. Likewise, **Text** $\rightarrow$ **fMRI** experiments are those which take in a semantic annotation vector input and predict an fMRI representation. Also note that we train the linear maps on the individual TRs as opposed to the 25 chunks.

We perform two primary experiments in this paper, **scene classification** and **scene ranking**. These experiments are applied to both the fMRI $\rightarrow$ Text and Text $\rightarrow$ fMRI settings. In the following description, we denote the predictor space by X and the target space by Y.

Suppose we are in the $X \rightarrow Y$ setting. For each time chunk $i \in [1, 25]$ in X-space, we predict chunk i in Y-space using the learned map, by applying the map individually to each TR within the time chunk. Then, we calculate the Pearson correlation of the predicted chunk i (represented by concatenating the representations for each TR in the chunk into one long vector) with each of the actual time chunks $j \in [1, 25]$, and we rank the chunk indexes by correlation.

Scene classification. Given the ranking of actual time chunks by correlation with the predicted chunk, we report the proportion of the time that the correct chunk index is ranked the highest. This measure has a 4% chance rate, meaning that if we randomly ranked the actual chunks, any particular chunk would be the top chunk 4% of the time.

Scene ranking. Given the ranking of actual time chunks by correlation with the predicted chunk, we calculate $1 - \frac{\text{average rank of the correct index}}{25}$. This measure has 50% chance rate, meaning that if we randomly ranked the actual time chunks, the average rank of any particular chunk would be in the middle.

We report both of these metrics because the 4% chance rate task gives a better idea of the distribution of the ranking, while other authors have used the 50% chance rate, obtaining ranking scores between $70\% - 80\%$ [18, 22, 19].

We also give some additional analysis of the properties of stacking previous time points, and discuss how they affect prediction capabilities. In particular, we observe the dependence of classification accuracy on the number of previous time steps.

Chapter 3

An Interpretable Temporal Dynamics Model

When predicting fMRI $\rightarrow$ Text and Text $\rightarrow$ fMRI, there are many possible approaches to using previous time steps in our linear maps. Throughout this chapter, we will suppose that X represents fMRI data ($n \times T$ data matrix, where n is the dimension of the fMRI data) and Y represents the semantic annotation data ($m \times T$ data matrix, where m is the dimension of the text representations). For simplicity, we wil discuss the fMRI $\rightarrow$ Text problem.

As discussed in the previous chapter, the most basic version of the mapping problem is to solve the equation $WX = Y$ where $W \in \mathbb{R}^{m \times n}$ may have constraints imposed on it (for instance, requiring its columns to be orthogonal turns this problem into the Procrustes problem). Here, we make no use of previous time points and note that the learning problem is invariant to column order. Now, we provide a discussion of various other methods of attempting to integrate timepoint information into a single representation.

3.1 Weighted Combination of Time Points

The simplest thing we can do to begin to integrate previous time points is to suppose that there are some weights $\phi = [1, \cdots, \phi_k]$ such that it is easier to predict column Y_t by learning a linear map between a weighted average of the columns $X_t, \cdots, X_{t-k}$ according to ϕ. For illustration purposes, we can write this out with matrices in the case where $k = 3$. Let us define a convolution matrix

$$\Phi_3 = \begin{bmatrix} 1 & \phi_1 & \phi_2 & \phi_3 & 0 & \cdots & 0 \\ 0 & 1 & \phi_1 & \phi_2 & \phi_3 & \ddots & 0 \\ 0 & 0 & 1 & \phi_1 & \phi_2 & \ddots & 0 \\ 0 & 0 & 0 & 1 & \phi_1 & \ddots & \phi_3 \\ 0 & 0 & 0 & 0 & 1 & \ddots & \phi_2 \\ 0 & 0 & 0 & 0 & 0 & \ddots & \phi_1 \\ 0 & 0 & 0 & 0 & 0 & \cdots & 1 \end{bmatrix}$$

In general, $\Phi \in \mathbb{R}^{T \times T}$, and column Φ_t is defined by

$$\Phi_t(i) = \begin{cases} \phi_j & \text{if } i = t - j \\ 1 & \text{if } i = t \\ 0 & \text{otherwise.} \end{cases} \tag{3.1}$$

Thus, we get the simple equation

$$WX\Phi = Y \tag{3.2}$$

We can also choose the normalize the columns of Φ, in which case we get a weighted average of the columns. If we set all $\phi_j = 1$, then we get the average.

In all cases, this model does not improve upon $WX = Y$, and in fact performs worse.

3.2 Temporal Dynamic Data Matrices

Since the weighted average idea above fails to work, we consider enlarging the hypothesis class of maps between X and Y. Consider stacking k previous time points below each time point to form $\hat{X} \in \mathbb{R}^{n*(k+1)\times T}$, as in Figure 2.3.

Notably, we can now consider each column of $\hat{X}$ to be a representation of the **dynamics** of the fMRI data over the past k timepoints plus the current timepoint, at the current timepoint. Thus, the space that the maps which take this data matrix as input is considerably larger than before (by a factor k). We therefore note that we are now dealing with models **linear in the temporal dynamics**, which means we are in fact modeling time series dependencies between two vector time series. See Figure 2.3.

3.2.1 Weighted Average Model

Then, we can rewrite the weighted average model from the previous section. Define the matrix $C_k \in \mathbb{R}^{n\times n*(k+1)}$ as

$$C_k = \begin{bmatrix} 1 & 0 & \cdots & 0 & \phi_1 & 0 & \cdots & 0 & \cdots\cdots & \phi_k & 0 & \cdots & 0 \\ 0 & 1 & & 0 & 0 & \phi_1 & & 0 & \cdots\cdots & 0 & \phi_k & & 0 \\ \vdots & & \ddots & & \vdots & & \ddots & & \cdots\cdots & \vdots & & \ddots & \\ 0 & & & 1 & 0 & & & \phi_1 & \cdots\cdots & 0 & & & \phi_k \end{bmatrix} \tag{3.3}$$

Then, the weighted average model becomes

$$WC_k\hat{X} = Y \tag{3.4}$$

where W's shape is unchanged as $W \in \mathbb{R}^{m \times n}$. C_k is completely described by the k parameters $\phi_1, \cdots, \phi_k$. Note that in either formulation of this model, the weighted average model only has k parameters in addition to $n * m$ parameters of the weight matrix W, for a total of $\Theta(k + n * m)$ parameters.

3.2.2 Full Temporal Model

The previous timepoint stacking approach discussed in the previous chapter learns a unique weight for every feature $1, \cdots, n$ of X for every single timepoint in the previous k time points can be described in this language as well. This approach corresponds to the other extreme, where we learn a unique weight for every feature $1, \cdots, n$ of X for every single timepoint in the previous k time points. This model is simply expressed as learning $\hat{W} \in \mathbb{R}^{m \times n*(k+1)}$ such that we have

$$\hat{W}\hat{X} = Y \tag{3.5}$$

We note that this model is much more expressive and is a much larger hypothesis class than the weighted average model: The full temporal model with $\hat{W}$ has $\Theta(n * m * k)$ parameters.

3.3 The Temporal Decay Model: Parametrizing Φ Over Time

In the previous section, we considered assigning a different weight to every fMRI feature at every time step. However, we might consider this setup to be an overly complex model. We may believe that for instance, each fMRI feature should have the same weight ϕ_j at timestep $t - j$ for current timestep t. However, this assumption leads to the old weighted average model, which experimentally does not work. We

thus relax the requirement that we have a single weight for each column, and allow different columns to have different weights as before. However, we want to enforce additional assumptions on the parameters so that the model is more interpretable. We can compromise by assuming that the weight parameters **decay** exponentially over the past k timesteps, at a different rate for each representation feature. Thus, we assign a single parameter to govern the weights over the **time course**, by assuming that the weights should decay with time and giving the weight decay a functional form.

3.3.1 Decay Weights

Since we believe that generally, the further in the past an event occurs, the less effect it should have on the present, we propose to weight previous timesteps with exponential decay. Given an fMRI feature dimension i from $1, \cdots, n$, assign a parameter λ_i such that the weight at time $t - j$ is

$$\phi_{i,j} = \frac{e^{j\lambda_i}}{\sum_{j^*=t}^{t-k} e^{(t-j^*)\lambda_i}} \tag{3.6}$$

and is 0 otherwise, where the weights are normalized over the previous timesteps. This equation defines our whole matrix $\Phi_{i,j} \in \mathbb{R}^{n\times(k+1)}$ with only n parameters. We call the vector of decay weights $\lambda = [\lambda_1, \cdots, \lambda_n]$ the **decay** vector.

3.3.2 Visualizing Decay Weights

In Figure 3.1, we briefly visualize what the decay weights look like in the case where we assume that each fMRI feature $i \in [n]$ has the same decay weight $\tilde{\lambda}$. Note that in this case, our convolution matrix/ decay matrix has only one additional parameter, for a total of $\Theta(n * m + 1)$ parameters, which is fewer parameters than even the weighted averaging from before. As is expected, this model does not perform well.

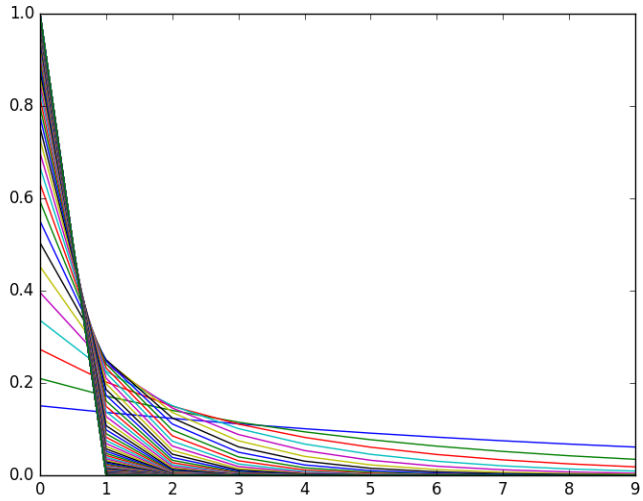

Figure 3.1: We demonstrate the weight values for 100 evenly spaced settings of $\tilde{\lambda}$ between 0.1 and 10 for $k = 10$. The y-axis is the weight value while the x-axis is the number of timesteps into the past. Larger values of $\tilde{\lambda}$ push more of the mass onto the current timestep, while small values of $\tilde{\lambda}$ make the distribution over the past k timesteps essentially uniform.

3.3.3 Unique Decay Weights for fMRI Features

We now specify n different decay weights $\lambda = [\lambda_1, \cdots, \lambda_n]$ for each of the fMRI features in the fMRI $\rightarrow$ Text setting. These decay weights completely define Φ. We define

$$C_k = [\Gamma_0, \Gamma_1, \cdots, \Gamma_k] \text{ where}$$
$$\Gamma_j = \begin{bmatrix} \frac{e^{j\lambda_1}}{Z_1} & 0 & \cdots & 0 \\ 0 & \frac{e^{j\lambda_2}}{Z_2} & & 0 \\ \vdots & & \ddots & \\ 0 & & & \frac{e^{j\lambda_n}}{Z_n} \end{bmatrix} \tag{3.7}$$

letting $Z_i = \sum_{j^*=t}^{t-k} e^{(t-j^*)\lambda_i}$ be the normalizing factor for each row. This version of the problem has $\Theta(n + m * n)$ parameters, since C_k is parametrized by n values.

Let us now write down the model where we specify n different decay weights $\lambda = [\lambda_1, \cdots, \lambda_n]$ for each of the fMRI features. These decay weights completely define Φ. Writing out the model, and letting $Z_i = \sum_{j^*=t}^{t-k} e^{(t-j^*)\lambda_i}$ the normalizing factor, we have that our convolution matrix C_k becomes

We formulate the problem setting

$$WC_k\hat{X} = Y \tag{3.8}$$

again, where $W \in \mathbb{R}^{m\times n}$, $C_k \in \mathbb{R}^{n\times n*(k+1)}$, $\hat{X} \in \mathbb{R}^{n*(k+1)\times T}$, and $Y \in \mathbb{R}^{m\times T}$. This version of the problem has $\Theta\left(n + m * n\right)$ parameters, since C_k is parametrized by n values. Notably, the number of parameters in this model has no dependence on the number of previous time steps chosen. Typically, this model will have slightly more parameters than the weighted average model which has $\Theta\left(k + m * n\right)$ parameters, since k ranges from $0 \rightarrow 30$ while $n = 20$. So the number of parameters is of a similar order. However, we are also incorporating an additional assumption encoded in the exponential decay. If the assumption is correct, then this model may be more appropriate for the actual data.

The neuroscientific motivation behind the assumption that there may be different rates of decay for different fMRI features comes from the notion that different parts of the brain operate over different time scales: The neurons in some parts of the brain fire a lot more rapidly and react to quickly changing stimuli, while other parts of the brain fire much more occasionally and change according to real world stimuli which occur at longer time scales.

The weighted average model did not take this into account, and also imposed no restriction on the weights over time. Therefore, if it is true that the weights generally exponentially decay, the model described in this section (unique decay) is in fact more general than the weighted-average model from earlier, since it allows parameters to

vary over where they actually do vary. It is also interesting to note that we cannot express this unique decay model in terms of a $\Phi \in \mathbb{R}^{T \times T}$ matrix as we did with the weighted average model.

3.4 The Learning Problem

In order to learn any of the models which require us to learn parameters W in addition to either C_k or Φ, the learning problem is non-convex, and we resort to alternating optimization without guarantees. In this section, only the models which have W and C_k are of interest since the weighted average model does not work in practice.

In our previous models, $(WX = Y; \hat{W}\hat{X} = Y)$, the solutions were linear and had a simple closed form whether we used ridge-regression or the Procrustes problem to solve them. This nice property is no longer the case when we try to solve $WC_k\hat{X} = Y$, since we are trying to learn two different matrices W, C_k with different constraints. Our alternating optimization problem proceeds as follows:

1. Randomly initialize λ and construct C_k.
2. Calculate $A = C_k\hat{X}$ and solve for W in $WA = Y$. This solution has a closed form if we use ridge-regression or the Procrustes problem.
3. Fix W. Use gradient descent on λ with loss function

$$f(\lambda) = \left\| Y - WC_k(\lambda)\hat{X} \right\|_F^2 \tag{3.9}$$

4. Return to step 2 and alternate until the solution converges.

We note that this procedure does not have guarantees. Empirically our procedure converges to reasonable optima when λ is initialized to be small.

Chapter 4

Results

4.1 Algorithm Improvement Ratios

In Table 4.1, we give both the maximum and average performance increase due to each of the individual methods. Here, we report the algorithmic improvement for the scene classification task, a task with a base 4% chance rate.

fMRI → Text	Maximum	Average
Previous Timesteps vs. None	5.3×	1.8×
Procrustes vs. Ridge	2.8×	1.3×
SRM/SRM-ICA vs. PCA	1.8×	1.3×
Weighted-SIF vs. Unweighted	1.6×	1.2×
Text → fMRI	**Maximum**	**Average**
Previous Timesteps vs. None	2.5×	0.5×
Procrustes vs. Ridge	3.0×	0.8×
SRM/SRM-ICA vs. PCA	2.3×	1.2×
Weighted-SIF vs. Unweighted	1.8×	1.1×

Table 4.1: Table of Improvement Ratios for Various Algorithmic Parameters: In this table we give the maximum and average improvement ratios for a specific algorithmic technique over another, including usage of previous time steps, SRM/SRM-ICA versus PCA, SIF-weighted annotation embeddings versus unweighted annotation embeddings, and Procrustes versus ridge regression for both fMRI → Text and Text → fMRI. When we use previous timesteps, we consider the results for using $5 - 8$ previous time steps. These numbers are all for the scene classification task. Note that the values from the maximum columns can be seen visually in Figures 4.2 and 4.3 respectively.

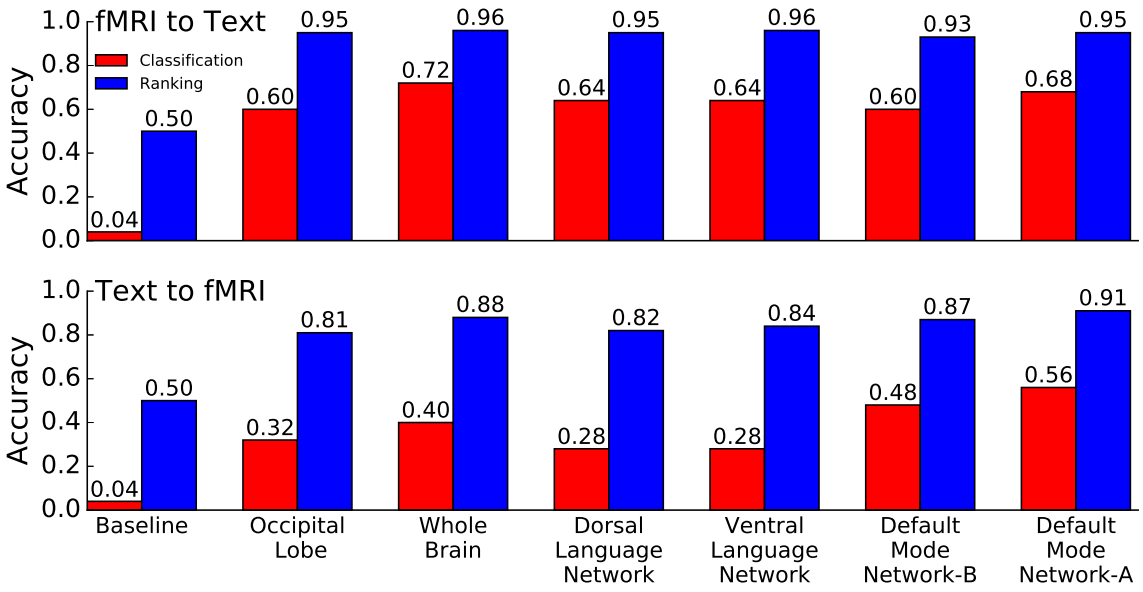

Figure 4.1: Best Bidirectional Accuracy Scores for Each Brain Region of Interest for both Scene Classification and Ranking: In this figure, for each ROI and for each experiment (Text $\rightarrow$ fMRI 4% (red), 50% (blue) chance rates; fMRI $\rightarrow$ Text 4% (red), 50% (blue) chance rates), we give the best performance as a percentage. For all measures, closer to 100% is better. We can see that Whole Brain, DMN-A, and DMN-B tend to perform the best, and that fMRI $\rightarrow$ Text performs better than Text $\rightarrow$ fMRI.

4.2 Top Performances over Algorithmic Choices

Figure 4.1 demonstrates that the DMN regions have nearly the best performance over the other ROIs studied, which fits with prior research in this area ([20], [21]). We achieve 72% accuracy over 4% chance with the Whole Brain region in the scene classification task. Since the scene ranking measure is always $\geq 80\%$, the average rank of the correct answer is in the top 20% of the scenes, which translates to top 5 scenes out of 25. For fMRI $\rightarrow$ Text we perform even better, where the average rank of the correct answer is in the top 10% of the scenes (top 3 scenes out of 25). Notably, we get excellent performance out of the Whole Brain region, which has 26000 voxels selected by merely choosing voxels whose intersubject correlation is above a certain threshold. This result demonstrates that our methods are not overly dependent on applying domain-specific knowledge (we do not necessarily have to preselect an ROI to get good results).

fMRI $\rightarrow$ Text. Here we discuss the performance of the fMRI $\rightarrow$ Text experiments. In Figure 4.1, we display the top accuracy over all algorithmic choices for each experiment. We achieve high accuracy performance, reaching 72% for the scene classification task for fMRI $\rightarrow$ Text and in the mid-90%s for the scene ranking tasks. In particular, the Whole Brain and the DMN regions perform best, supporting previous work by [20] and others demonstrating that the DMN plays an important role in narrative processing.

Text $\rightarrow$ fMRI. On the other hand, we see that the Text $\rightarrow$ fMRI experiments perform worse than the fMRI $\rightarrow$ Text experiments. The best top-1 scene classification accuracy performance is 56% for the DMN-A region, and the other top performing regions get accuracy in the mid-to-high 40% accuracy. For the ranking task, performance ranges from $80\% - 90\%$, which is again slightly worse than the fMRI $\rightarrow$ Text ranking experiment.

4.3 Comparing Algorithmic Choices

In order to simplify presentation for Figures 4.2 and 4.3, we chose to fix the algorithmic parameters that uniformly outperformed other options. All linear maps for fMRI $\rightarrow$ Text were learned using the Procrustes method and all linear maps for Text $\rightarrow$ fMRI were learned using the ridge regression approach. We fixed these for comparison purposes since, for fMRI $\rightarrow$ Text scene classification, Procrustes performed 1.25$\times$ better than ridge on average (Table 4.1). On the other hand, ridge performed 1.2$\times$ better than Procrustes on average over Text $\rightarrow$ fMRI scene classification (Table 4.1). As a caveat, there were exceptions to the rule, as the max ratios in Table 4.1 indicate. In Figures 4.2 and 4.3, for the data points that are labeled as using previous time steps, we reported the result for 8 previous time steps. The optimal number of previous time steps for fMRI $\rightarrow$ Text was typically between $5-8$, and so we fixed that choice of parameter across all of the graphs in these figures.

Comparing SRM and SRM-ICA to PCA. We see considerable improvement on best-case performance when using SRM or SRM-ICA over PCA, particularly on the fMRI $\rightarrow$ Text tasks, in some cases gaining as much as 1.8$\times$ the top-1 scene classification performance of PCA, as demonstrated in Figure 4.2. Typically, SRM-ICA tends to perform slightly better, especially on the Whole Brain ROI. The case is weaker for Text $\rightarrow$ fMRI, since though we can find that performance increases by as much as 2.3$\times$ the top-1 scene classification performance, the average benefit is smaller (Table 4.1, Figure 4.3). If we look at average case improvements, we see considerable gains in both directions: SRM/SRM-ICA improve on average by 1.3$\times$ over PCA for fMRI $\rightarrow$ Text scene classification, and on average by 1.2$\times$ over PCA on Text $\rightarrow$ fMRI scene classification. For the ranking tasks, we note that while performance improvement for the best selections of algorithm parameters is not as distinct, SRM and SRM-ICA can drastically improve upon PCA performance for

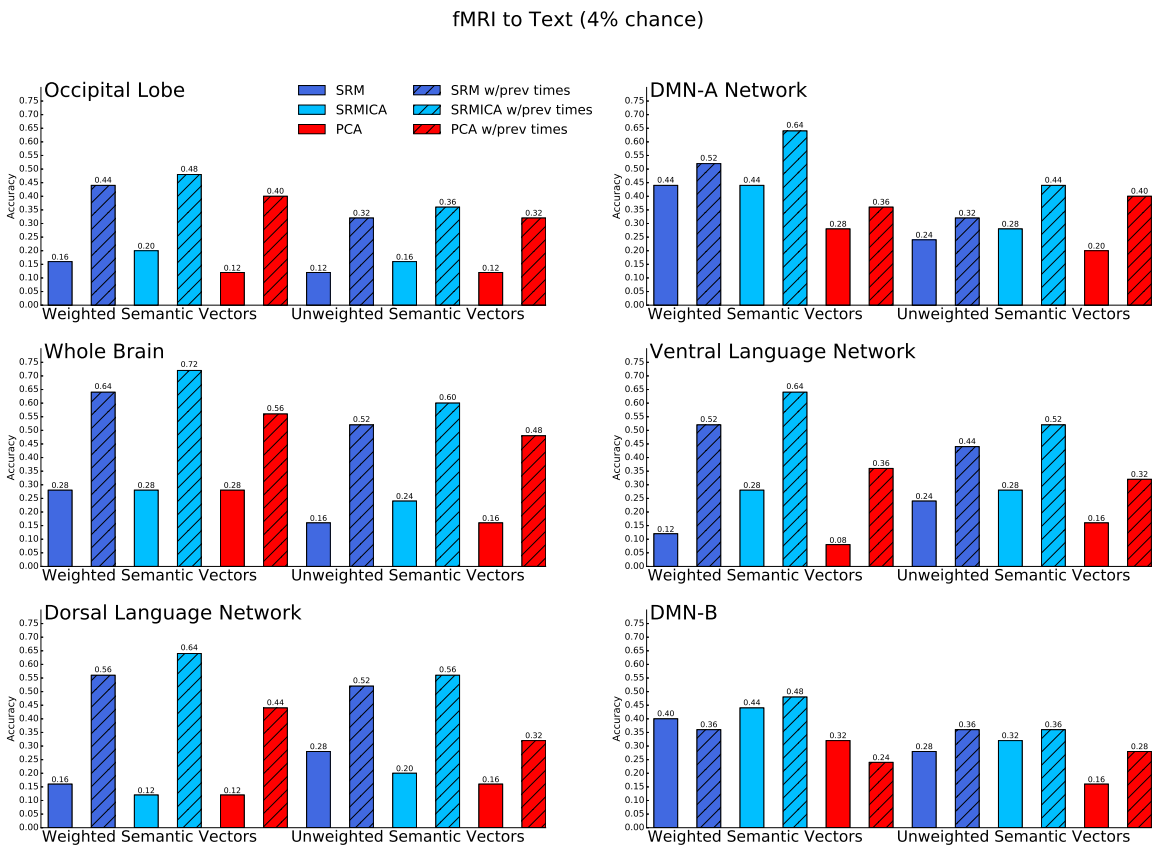

Figure 4.2: Comparisons for all ROIs for the fMRI $\rightarrow$ Text Top-1 Scene Classification Experiment: The chance rate for this task is 4%. Each plot is for a different ROI. Here, we only display results which use the Procrustes linear map since it on average performs better than ridge regression for fMRI $\rightarrow$ Text. We also fix the number of previous time points used for the shaded bars at 8 previous time steps, since that tends to be near optimal. We present comparisons between SRM/SRM-ICA and PCA using blue colors versus red colors, and compare weighted semantic aggregation (left) to unweighted semantic aggregation (right) by x-axis position.

poor selection of parameters. This fact suggests that one should always use SRM or SRM-ICA over PCA, since on new datasets where it is not known which linear map to use, or the number of previous time points to incorporate in the analysis and so on, our results here suggest that these SRM-variants will improve strongly upon PCA if the parameters are poorly chosen, and still improve decently upon PCA otherwise.

Weighted vs. Unweighted Aggregation of Word Embeddings. Using the SIF-weighted embeddings improves upon unweighted averaging when featurizing the

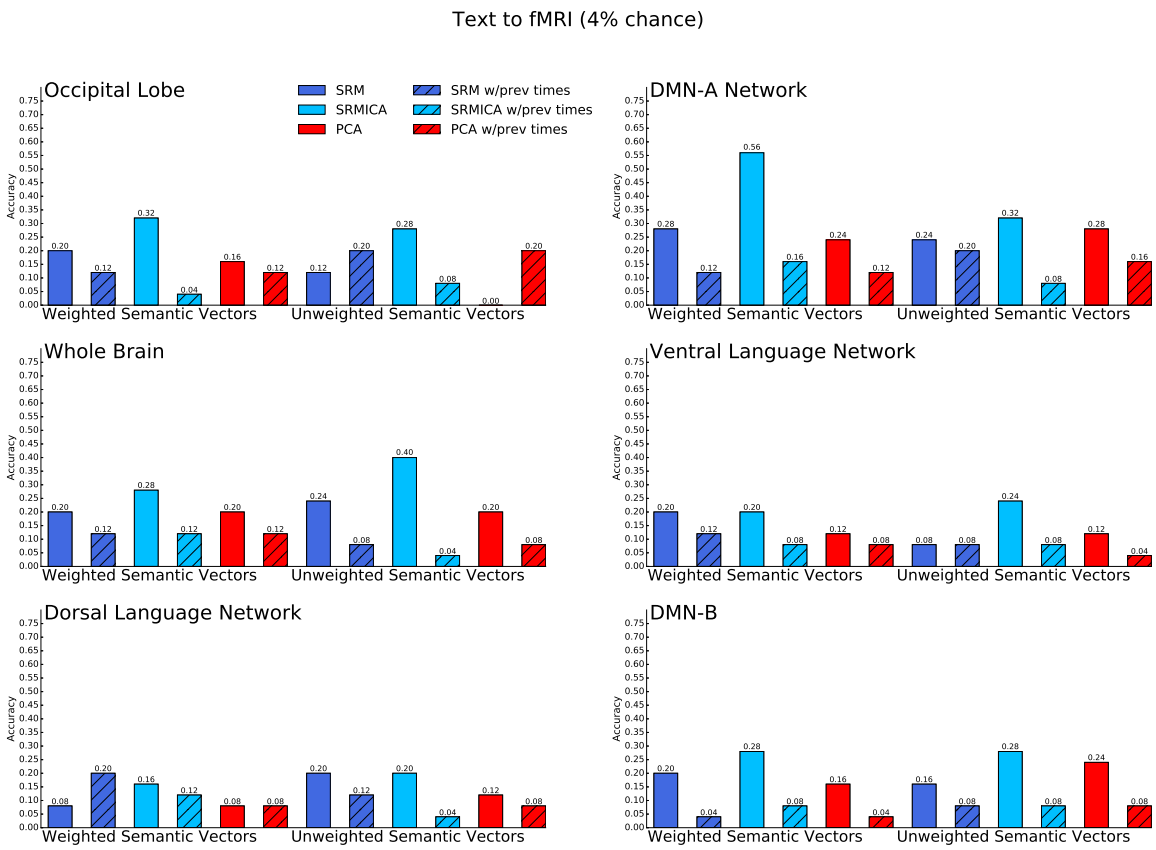

Figure 4.3: Comparisons for all ROIs for the Text → fMRI Top-1 Scene Classification Experiment: The chance rate for this task is 4%. Each plot is for a different ROI. Here, we only display results which use the ridge regression linear map since it on average performs better than Procrustes for Text → fMRI. We also fix the number of previous time points used for the shaded bars at 8 previous time steps, since that tends to be near optimal. We present comparisons between SRM/SRM-ICA and PCA using blue colors versus red colors, and compare weighted semantic aggregation (left) to unweighted semantic aggregation (right) by x-axis position.

annotation vectors as well. Examining Table 4.1 and Figure 4.2, we see that for fMRI → Text top−1, there is improvement on best-case performance by as much as 1.3× by using weighted embeddings. On average, we see that weighted embeddings improve by 1.2× over the unweighted embeddings. Looking at Figure 4.3, the case is weaker for Text → fMRI top−1; while for some algorithms and ROIs we see as much as 2.5× improvement on best-case performance by weighted aggregation embeddings, we also see that sometimes unweighted averaging can outperform weighted averaging.

However, on average, weighted embeddings improve by 1.1× over unweighted averaged embeddings.

The Effects of Previous Time Points. Figure 4.2 demonstrates the positive effect of adding previous time steps to the accuracy scores for the fMRI → Text case. Table 4.1 demonstrates that at best, using previous timepoints can improve performance by as much as 5.3×. On average, this improvement is 1.8×, nearly doubling performance. On the other hand, Figure 4.3 shows that for Text → fMRI, adding previous time steps almost universally hurts performance and on average halves performance (Table 4.1). This fact is also evident from Figure 4.4, which illustrates the situation for the DMN-A ROI.

Notably, the effect of using previous time steps is different from learning a hemodynamic response function, which other authors [22, 12] have done in the past. Instead, we are investigating whether information from longer time scales helps improve performance. In Figure 4.4, we see that there are some peaks in classification performance between 5 and 8 previous time steps ago (or $7.5 - 9.0$ seconds ago, after having taken into account the HRF). However, using any number of previous time steps (up to as long as 30 TRs ago, or 45 seconds) still improves over the baseline of using no previous time steps.

For Text → fMRI however, the story is different. We see no improvement in performance when using previous time points, and in fact performance decreases (Figure 4.4). We can first examine the situation from a generative model perspective. After aligning the fMRI and text data, our linear model either says the fMRI is explained by a linear transformation of the text data ($X = WY$) or the text data is explained by a linear transformation of the fMRI data ($Y = WX$). If the former case is true, then learning a linear map in the Text → fMRI problem should work well. If the latter is true, then the fMRI → Text task should succeed. Note that a good

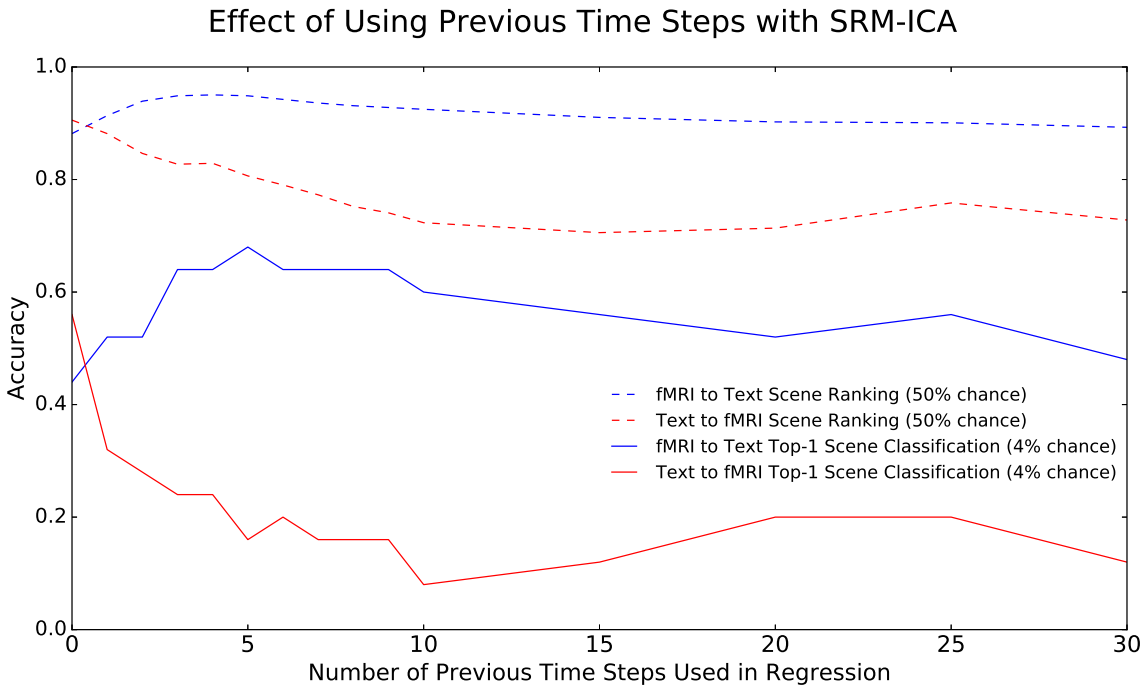

Figure 4.4: Varying Previous Timesteps: For the DMN-A region, choosing SRM-ICA, weighted average, Procrustes for the fMRI $\rightarrow$ Text linear map, and ridge for the Text $\rightarrow$ fMRI linear map, we plot the relationship between accuracy (y-axis) and number of previous time points used in the linear map fit (x-axis). We can see a peak at around using $5-8$ previous TRs as optimal for the fMRI $\rightarrow$ Text tasks, and a relatively monotone decay for using any previous TRs in the Text $\rightarrow$ fMRI tasks.

linear map existing for one direction does **not** imply that a good linear map exists for the other direction, since these linear maps are not invertible due to their low rank. We believe the central reason that a better linear map exists for the fMRI $\rightarrow$ Text direction is due to the relatively high correlation between the semantic representations compared to the correlation between fMRI states. As a result, the fMRI $\rightarrow$ Text task is a many-to-one problem, while we perform a more difficult one-to-many task when we attempt Text to fMRI.

4.4 Analyzing the Interpretable Temporal Model

In Figure 4.5, we see that performance increases when we use the temporal decay model for the fMRI $\rightarrow$ Text task, but decreases for the Text $\rightarrow$ fMRI task. Addi-

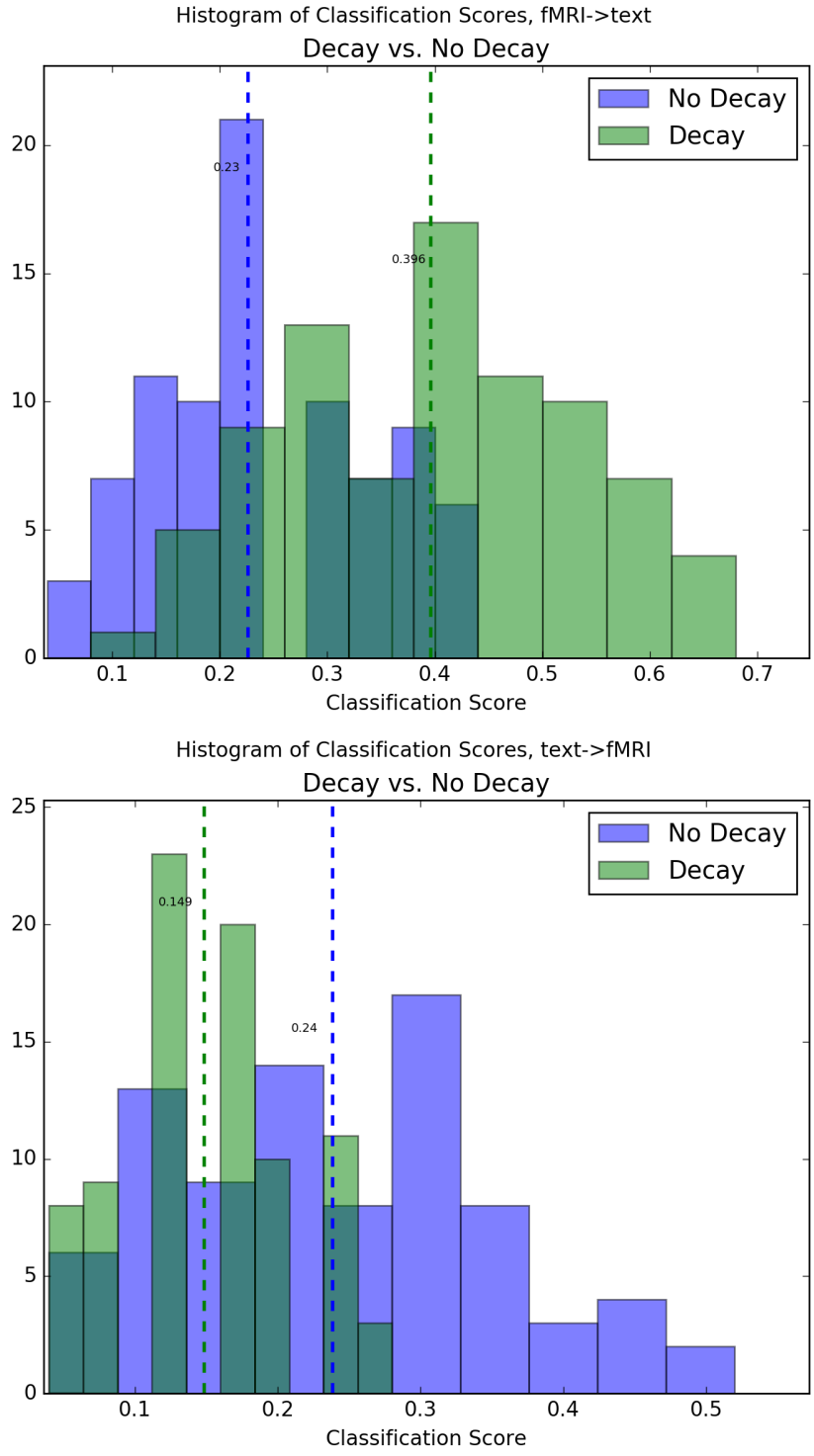

Figure 4.5: Decay vs. No Decay: We plot the histogram over all algorithmic choices (choice of dimension reduction, choice of linear map, choice of ROI, etc.) of the performance for scene classification. In this figure, "No Decay" refers to the setting where **no previous time points are used**. "Decay" refers to the performance of the interpretable temporal dynamics model. Note that peak performance accuracy is similar to the full temporal model.

tionally, performance for each brain area matches or is slightly less than performance in the temporal model used in this work (obtaining 64% accuracy over 4% chance rate for the scene classification task for the DMN-A region), thus demonstrating that we can replace the full temporal model with a more interpretable model and lose very little. The same conclusions with respect to shared space dimension reduction, word embeddings, and brain ROI performances hold when we replace the temporal dynamics model.

Chapter 5

Conclusions and Future Work

5.1 Conclusions

In this paper, we have explored several methods that improve our success at mapping between fMRI response to a natural stimulus and semantic text data describing this stimulus. We see that SRM and SRM-ICA perform considerably better than simple averaging or using PCA. Figure 4.2 demonstrates that weighted aggregation of the words in semantic space to form annotation vectors over simple averaging improves the baseline accuracy by a reasonable amount. We also show that adding previous time steps improves accuracy substantially.

Using SRM-ICA in fMRI space, weighted annotation vectors in semantic space and a Procrustes linear map learned between the concatenations of five previous time points in fMRI and semantic space, we are able to achieve 72% scene classification accuracy over 4% chance rate for the Whole Brain region on the fMRI $\rightarrow$ Text task.

Other ROIs are typically above 60% scene classification accuracy as well. Similarly, in the scene ranking task, we achieve $> 90\%$ average rank for the correct answer across ROIs. Text $\rightarrow$ fMRI does not perform as well but is still far above chance (56% with DMN-A ROI for 4% chance rate, and $> 80\%$ average rank across ROIs).

Another takeaway is that SRM and SRM-ICA improve upon PCA almost always, and provide particularly substantial improvement in cases where the other parameter settings (like the semantic featurization or selection of linear map and associated hyper-parameters) are not necessarily tuned. These results indicate that we are able to use multiple subjects to learn a 20-dimensional shared space for the fMRI data that increases performance on our experiments. Thus, we provide concrete evidence towards the hypothesis made in [12] regarding the existence of a **shared** fMRI representation across multiple subjects that correlates significantly with **fine-grained** semantic context vectors derived via statistical word co-occurrence properties.

The method of combining word vectors is another essential part of our results. We demonstrate that weighted-SIF averaging [3] for aggregating individual elements of a word sequence performs on average 1.2× better than unweighted averaging for fMRI $\rightarrow$ Text top-1 scene classification, and on average 1.1× better for Text $\rightarrow$ fMRI top-1 scene classification. Since we use only semantic vectors to featurize a movie stimulus dataset, our work provides additional support for the notion that the distributional hypothesis of word meaning may extend to real life multi-sensory stimuli.

Finally, we note that using multiple previous timepoints when mapping from fMRI $\rightarrow$ Text is very beneficial and significantly improves results by a factor of as much as 5.3×, and on average nearly doubles performance (Table 4.1).

We also presented an interpretable temporal dynamics model which improves maps from fMRI to Text. It remains to explain why Text to fMRI does not perform as well. We believe the central reason is due to the relatively high correlation between the semantic representations compared to the correlation between fMRI states. As a result, we perform a one-to-many task when we attempt Text to fMRI, which is more difficult. This idea suggests one avenue of future work; namely to decorrelate the annotation representations further.

The main thrust of the temporal dynamics model is that we are trying to integrate information from the past over different timescales: Our assumption is that all the information for the current scene is not located inside a single timepoint representation, but is rather spread out across past timepoints. One explanation of our decreased performance with the temporal dynamics model is that annotation representations already contain integrated information across time. When we try to integrate in semantic text space, we end up only adding noise to the annotation representation: A human has already processed the information in the past and integrated it into the present representation. Thus for annotation embeddings, it may not make sense to use previous timepoints in the representation, while it still may make sense to use previous timepoints for the fMRI in the representation.

5.2 Future Work

As for future work, the first question that arises is whether the results on the Sherlock dataset generalize to other datasets, and whether the techniques which were the best on the Sherlock dataset are the best on the other datasets. Preliminary results verify that our techniques in matching fMRI and Text generalize to more varied stories, in both audio and visual formats on other datasets.

We would also like to experiment further with de-correlation procedures for the text annotation embeddings. As noted above, it is conceivable that coming up with text embeddings which are less correlated with each other will aid the performance of the Text $\rightarrow$ fMRI task. Initial directions will focus on removing common linear components, as determined by spectral procedures like singular value decomposition (SVD).

The ultimate goal of the experiments discussed in this paper is to enable the true decoding of thoughts in the human brain: Namely, given some measurements of

brain activity, can we reconstruct in text the thoughts that occurred during the measurement? There are several obstacles to this dream: For one, despite our mapping accuracy in this work, fMRI data is still considerably noisy and blurry, especially in the temporal domain. Thus, it seems likely that we would only be able to reconstruct relatively coarse-grained descriptions of human thought with this technology. The second considerable obstacle is the amount of existing data of this format, a factor intrinsic to the level of difficulty of accurate fMRI data collection. Currently, we have a total of nearly 2.5 hours worth of fMRI data recorded of subjects being exposed to natural stimuli like movies and audio stories. Additionally, the subjects are not necessarily the same for each experiment, and we only have a small set of different human subjects (around 50). Though techniques like the shared response model suggest that generalization over human subjects is possible with a small number of data points, it is unclear the extent to which generalization will occur given larger sets of people. However, techniques to analyze multi-dataset data and efforts to gather more fMRI data from more people are ongoing. Dataset open-sourcing is becoming a continuing trend. It is conceivable that in several years, we will be able to obtain enough fMRI data associated with the human experience of stories and narratives to be able to entertain approaches similar to the image captioning literature, recently made possible and popular by advances using deep neural networks [14]. The futuristic notion of automatic thought decoding (as recently popularized by the companies Neuralink and Facebook) probably requires much more fine-grained methods of measuring the human brain.

Bibliography

[1] D.L. Ames, C. J. Honey, M.A. Chow, A. Todorov, and U. Hasson. Contextual Alignment of Cognitive and Neural Dynamics. *Journal of Cognitive Neuroscience*, 27:655–664, 2015.

[2] Sanjeev Arora, Yuanzhi Li, Yingyu Liang, Tengyu Ma, and Andrej Risteski. A Latent Variable Model Approach to PMI-based Word Embeddings. *Transactions of the Association for Computational Linguistics*, 4, 2016.

[3] Sanjeev Arora, Yingyu Liang, and Tengyu Ma. A Simple but Tough-to-Beat Baseline for Sentence Embeddings. *International Conference on Learning Representations (ICLR) 2017*, 2017.

[4] Janice Chen, Yuan Chang Leong, Christopher J. Honey, Chung H Yong, Kenneth A. Norman, and Uri Hasson. Shared memories reveal shared structure in neural activity across individuals. *Nature Neuroscience*, 20:115–125, 2017.

[5] Po-Hsuan Chen, Janice Chen, Yaara Yeshurun, Uri Hasson, James V. Haxby, and Peter J. Ramadge. A Reduced-Dimension fMRI Shared Response Model. *The 29th Annual Conference on Neural Information Processing Systems (NIPS)*, 2015.

[6] B.R. Conroy, B.D. Singer, J.S. Guntupalli, P.J. Ramadge, and J.V. Haxby. Inter-subject alignment of human cortical anatomy using functional connectivity. *NeuroImage*, 81:400–411, 2013.

[7] Scott Deerwester, Susan T. Dumais, George W. Furnas, Thomas K. Landauer, and Richard Harshman. Indexing by Latent Semantic Analysis. *Journal of the American Society for Information Science*, 41:391–407, 1990.

[8] U. Hasson, R. Malach, and D.J. Heeger. Reliability of cortical activity during natural stimulation. *Trends in Cognitive Science*, 14:40–48, 2010.

[9] Uri Hasson, Yuval Nir, Ifat Levy, Galit Fuhrmann, and Rafael Malach. Inter-subject Synchronization of Cortical Activity During Natural Vision. *Science*, 303:1634–1640, 2004.

[10] James V. Haxby, J. Swaroop Guntupalli, Andrew C. Connolly, Yaroslav O. Halchenko, Bryan R. Conroy, M. Ida Gobbini, Michael Hanke, and Peter J.

Ramadge. A Common, High-Dimensional Model of the Representation Space in Human Ventral Temporal Cortex. *Neuron*, 72:404–416, 2011.

[11] Christopher J. Honey, Christopher R. Thompson, Yulia Lerner, and Uri Hasson. Not Lost in Translation: Neural Responses Shared Across Languages. *Journal of Neuroscience*, 32:15277–15283, 2012.

[12] Alexander G. Huth, Wendy A. deHeer, Thomas L. Griffiths, Frédérick E. Theunissen, and Jack L. Gallant. Natural speech reveals the semantic maps that tile human cerebral cortex. *Nature*, 532:453–458, 2016.

[13] Alexander G. Huth, Shinji Nishimoto, An T. Vu, and Jack Gallant. A Continuous Semantic Space Describes the Representation of Thousands of Object and Action Categories across the Human Brain. *Neuron*, 76:1210–1224, 2012.

[14] Andrej Karpathy and Fei-Fei Li. Deep Visual-Semantic Alignment for Generating Image Descriptions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[15] Ryan Kiros, Yukun Zhu, Ruslan Salakhutdinov, Richard S. Zemel, Antonio Torralba, Raquel Urtasun, and Sanja Fidler. Skip-Thought Vectors. *Advances in Neural Information Processing Systems*, 2015.

[16] Quoc Le and Tomas Mikolov. Distributed Representations of Sentences and Documents. *Proceedings of the 31st International Conference on Machine Learning, JMLR*, 32, 2014.

[17] Tom M. Mitchell, Svetlana V. Shinkareva, Andrew Carlson, Kai-Min Chang, Vicente L. Malave, Robert A. Mason, and Marcel Adam Just. Predicting Human Brain Activity Associated with the Meanings of Nouns. *Science*, 320:1191–1194, 2008.

[18] Francisco Pereira, Greg Detre, and Matthew Botvinick. Generating Text from Functional Brain Images. *Frontiers in Human Neuroscience*, 5:72, 2011.

[19] Francisco Pereira, Bin Lou, Brianna Pritchett, Nancy Kanwisher, Matthew Botvinick, and Evelina Fedorenko. Decoding of generic mental representations from functional MRI data using word embeddings. *bioRxiv preprint*, 2016.

[20] M. Regev, C. J. Honey, E. Simony, and U. Hasson. Selective and Invariant Neural Responses to Spoken and Written Narratives. *J Neurosci*, 33:15978–15988, 2013.

[21] Erez Simony, Christopher J. Honey, Janice Chen, O. Lositsky, Y. Yeshurun, and Uri Hasson. History dependent dynamical reconfiguration of the default mode network during narrative comprehension. *Nature Communications*, 7, 2016.

[22] Leila Wehbe, Brian Murphy, Partha Talukdar, Alona Fyshe, Aaditya Ramdas, and Tom Mitchell. Simultaneously Uncovering the Patterns of Brain Regions Involved in Different Story Reading Subprocesses. *PLOS ONE*, 9, 2014.

[23] Y. Yeshurun, S. Swanson, E. Simony, J. Chen, C. Lazaridi, C. J. Honey, and U. Hasson. Same story, different story: the neural representation of interpretive frameworks. *Psychological Science*, January 2017.

[24] Hejia Zhang, Po-Hsuan Chen, Janice Chen, Xia Zhu, Javier S. Turek, Theodore L. Willke, Uri Hasson, and Peter J. Ramadge. A searchlight factor model approach for locating shared information in multi-subject fmri analysis. *arXiv preprint*, 2016.