Lab 3.1 - Predicting Blood-Oxygen Levels Across Brain Voxels During Podcast Listening, Stat 214, Spring 2025

April 14, 2025

1 Introduction

This report investigates the application of linear ridge regression models to predict blood-oxygen-level-dependent (BOLD) responses across brain voxels using functional Magnetic Resonance Imaging (fMRI) data collected from subjects while listening to podcasts. The central goal is to analyze how effectively different textual embedding methods—specifically Bag-of-Words (BoW), Word2Vec, and GloVe—capture linguistic information that correlates with voxel-level neural activity.

To evaluate these embeddings, ridge regression models are fitted to predict voxel responses from the embeddings derived from podcast transcripts. Model performance is assessed using correlation coefficients (CC) averaged across voxels. Cross-validation is employed to rigorously select the best embedding method based on multiple metrics (mean, median, top 1 percentile, and top 5 percentile CC). A detailed evaluation, including distribution analyses across voxels and a stability analysis across different subjects, further provides insights into the reliability and interpretability of these embeddings from the perspective of predictability and stability (PCS).

By systematically comparing these embedding methods, this study aims to illuminate not only the most predictive model but also contribute to the broader understanding of how linguistic information is represented and processed in the brain.

2 Generating Embeddings

In this section, we transform raw transcript data into numerical feature representations that can be used to predict fMRI brain responses. We implement and compare three types of embeddings: Bag-of-Words (BoW), Word2Vec, and GloVe.

2.1 Data Description

Each story is represented using a DataSequence object, which stores a sequence of tokenized words along with their corresponding word-level timestamps (data_times) and TR-level timestamps (tr_times). The data_times indicate the temporal midpoint of each spoken word, while tr_times correspond to the timepoints of fMRI acquisition (i.e., TRs). These timestamps enable precise temporal alignment between the story content and the neural responses captured during scanning, serving as the foundation for linking word embeddings to brain activity.

Story ID	# Words	# TRs	First 2 Words	First 2 data_times	First 2 tr_times
`sweetaspie`	697	172	[, i]	[0.006, 1.16]	[-9.0, -7.0]
`thatthingonmyarm`	2073	449	[, people]	[0.006, 0.307]	[-9.0, -7.0]
`tildeath`	2297	338	[um, it]	[3.794, 4.143]	[-9.0, -7.0]
`indianapolis`	1554	317	[, let's]	[0.006, 1.898]	[-9.0, -7.0]

Table 1: Summary of stories with word & TR counts and sample timing & word data

In table 1, the story "sweetaspie" consists of 697 words and 172 TR segments. Each TR segment aggregates

the activity of temporally adjacent words through interpolation methods, such as Lanczos resampling, to approximate the semantic content presented at each fMRI timepoint.

2.2 BoW Embedding Construction

2.2.1 Preprocessing

To convert raw story transcripts into a format suitable for machine learning models, we first transform the sequence of words into numerical vectors. One straightforward and interpretable approach is the **Bag-of-Words (BoW)** representation, which encodes each story as a vector of word frequency counts. This transformation serves as the basis for constructing the feature matrix X used in subsequent modeling.

We employ the `CountVectorizer` from `scikit-learn` to convert each story's transcript into a sparse matrix of word counts. The vocabulary is derived from all training set stories, with standard preprocessing steps applied to enhance consistency and reduce noise. Specifically:

- All words are lowercased prior to vectorization.
- Common English stopwords are removed via `stop_words="english"`.
- The vocabulary is fitted on the training data only, then applied to all stories to ensure consistency.

The resulting BoW matrix has one row per word (after TR-level downsampling) and one column per vocabulary word. Each entry in the matrix corresponds to the frequency of a particular vocabulary word within a given TR-aligned segment of the story.

BoW Matrix Shapes for Each Story

Story ID	Word-Level Shape	Vocabulary Size (V)
`sweetaspie`	(697, 12551)	12551
`thatthingonmyarm`	(2073, 12551)	12551
`tildeath`	(2297, 12551)	12551
`indianapolis`	(1554, 12551)	12551

Table 2: Shape of Bag-of-Words matrices for each story (rows = word tokens, columns = vocabulary terms)

Table 2 shows that each matrix has shape $[N, V]$, where N is the number of words, and $V = 12551$ is the size of the fixed vocabulary.

2.2.2 Downsampling and Trimming to TR Level

The raw Bag-of-Words (BoW) matrices are initially aligned at the word level, with each row corresponding to a specific word in the story transcript. However, since fMRI signals are acquired at fixed temporal intervals (i.e., TRs), a temporal mismatch arises between the word-level features and the brain response data. To enable regression modeling, it is essential to project the input features onto the same temporal axis as the fMRI recordings. This process yields TR-aligned feature matrices X that are temporally synchronized with the corresponding fMRI response matrices Y.

We perform temporal downsampling using the **Lanczos interpolation method**, a signal processing technique known for its ability to minimize aliasing and preserve high-frequency information. For each story, we extract `data_times`, which mark the temporal midpoints of each spoken word, and `tr_times`, which correspond to the fMRI scan timepoints. The function `lanczosinterp2D(data, data_times, tr_times)`, provided in the preprocessing module, is then applied to interpolate the BoW matrix from word-level to TR-level resolution.

This interpolation produces a matrix of shape $[N_{\text{TR}}, V]$, where each row represents a temporally weighted combination of vocabulary terms relevant to a specific TR. The method assigns higher weights to words near the TR and lower weights to more distant words, mimicking the brain's time-sensitive integration of linguistic information.

To further improve alignment between the stimulus features and the neural data, we trim the first 5 seconds

and the last 10 seconds of each story. This step removes the initial hemodynamic lag and any trailing TRs that may not correspond cleanly to stimulus content, thereby enhancing the quality of the temporal correspondence between X and Y.

Story	Word-Level Shape	TR-Level Shape	TR-Level Shape (Trimmed)
sweetaspie	(697, 12,551)	(172, 12,551)	(157, 12,551)
thatthingonmyarm	(2,073, 12,551)	(449, 12,551)	(434, 12,551)
tildeath	(2,297, 12,551)	(338, 12,551)	(323, 12,551)
indianapolis	(1,554, 12,551)	(317, 12,551)	(302, 12,551)

Table 3: Transformation of BoW matrix shapes from word-level to TR-level resolution via Lanczos interpolation and temporal trimming.

2.2.3 Delay Embedding

The brain does not respond instantaneously to external stimuli. Both the intrinsic delays in neural processing and the sluggish nature of the fMRI BOLD response introduce a temporal lag between stimulus presentation and measured brain activity. As a result, the fMRI signal observed at any given TR is likely influenced not only by the stimulus at that moment, but also by stimuli from several seconds earlier.

To account for this temporal dependency, we incorporate **delay embedding** into our modeling pipeline. This technique augments the input features with information from preceding timepoints, allowing the model to capture temporal context and better approximate the brain's integration of linguistic information over time.

After aligning the BoW features to the TR level using Lanczos interpolation and trimming the initial and final segments, we applied delay embedding via the `make_delayed()` function from the preprocessing module. Specifically, we selected a delay range of $[1, 2, 3, 4]$, meaning that for each TR t, the model receives the BoW vectors from the previous four TRs as input. The final feature vector at time t is thus constructed by concatenating these delayed vectors:

$$X_t = [x_{t-1},\ x_{t-2},\ x_{t-3},\ x_{t-4}]$$

This transformation preserves the number of TRs (rows) but expands the number of features (columns) by a factor equal to the number of delays. Given a vocabulary size of 12,551 and four delay steps, the final feature matrix for each story has dimensionality $[n,\ 50{,}204]$, where n is the number of TRs remaining after trimming.

Story	TR-Level Shape (Trimmed)	TR-Level Shape (After Delay)
sweetaspie	(157, 12,551)	(157, 50,204)
thatthingonmyarm	(434, 12,551)	(434, 50,204)
tildeath	(323, 12,551)	(323, 50,204)
indianapolis	(302, 12,551)	(302, 50,204)

Table 4: Shape of feature matrices before and after delay embedding.

This delay embedding step enables the regression model to access a temporally extended view of the stimulus, reflecting how linguistic and semantic context builds up over time in natural language comprehension.

2.2.4 Word2Vec & GloVe Embedding Construction

While the Bag-of-Words (BoW) representation provides a simple and interpretable encoding of text, it does not capture the semantic relationships between words. In contrast, distributed word embeddings such as Word2Vec and GloVe represent words as dense vectors in a continuous space, where semantic similarity

is reflected in vector proximity. These richer representations are more aligned with how humans process meaning and may therefore improve the alignment between language and brain activity.

To evaluate the benefit of semantically informed features, we incorporated pre-trained Word2Vec and GloVe embeddings into our pipeline. Both embedding types were processed using the same steps as BoW, differing only in the source of the word vectors, thereby enabling a controlled comparison between vector spaces.

Word2Vec. We utilized the pre-trained Word2Vec model trained on Google News (300 dimensions), obtained via `gensim.downloader`. For each word in the transcript, we retrieved its corresponding vector as follows:

- If the word appeared in the embedding vocabulary, we used its 300-dimensional vector.
- If the word was out-of-vocabulary (OOV), we assigned a zero vector of the same dimension.

GloVe. We used the GloVe Wikipedia + Gigaword 300D embedding set (also 300 dimensions), downloaded using the same interface. The same processing pipeline was applied, with the only difference being the source of the pre-trained embeddings.

Once word-level vectors were obtained for either embedding model, we applied the following steps: Downsample features, Trimming, Delay embedding. This embedding pipeline mirrors the BoW-based workflow while providing semantically richer representations that may better align with neural responses to natural language.

2.3 Benefits of Pre-trained Embeddings

Pre-trained word embeddings such as Word2Vec and GloVe offer several key advantages over traditional Bag-of-Words (BoW) representations when used as input features for modeling brain responses to language stimuli. These advantages contribute to improved model performance, robustness, and interpretability.

1. Semantic Richness. Unlike BoW, which treats each word as an independent token with no relation to others, pre-trained embeddings encode semantic similarity within a continuous vector space. Words with similar meanings are located near each other in this space, allowing the model to capture graded semantic relationships. For example, words like "*king*" and "*queen*", or "*dog*" and "*puppy*", will have similar representations in embedding space—facilitating better generalization across related concepts.

2. Lower Dimensionality. BoW vectors are typically high-dimensional and extremely sparse (e.g., over 12,000 dimensions), which increases the risk of overfitting and computational burden. In contrast, Word2Vec and GloVe embeddings are dense and low-dimensional (typically 300 dimensions), providing a more compact and efficient representation of lexical information.

3. Transfer Learning. Pre-trained embeddings are trained on massive external corpora such as Google News or Wikipedia, capturing a wide range of syntactic and semantic patterns beyond those present in the experimental stimulus set. This external knowledge can be leveraged to improve prediction performance, especially when working with relatively small datasets, as is common in neuroimaging studies.

4. Robustness to Rare Words. In BoW models, rare or unseen words are either ignored or mapped to zero vectors, leading to information loss. Pre-trained embeddings offer more robust handling of infrequent terms, and may still provide meaningful representations through subword information or similarity to known words. This improves generalization and helps the model better deal with vocabulary gaps.

These advantages make pre-trained embeddings a strong candidate for use in modeling tasks that aim to map natural language to brain activity, particularly when the goal is to move beyond surface-level word frequencies and toward semantically meaningful representations.

3 Ridge Regression and Evaluation

3.1 Introduction of ridge regression

Ridge regression is a linear regression method that adds an L2 penalty to the loss function to prevent overfitting. It is especially useful when there are many features or when the features are highly correlated. Moreover, bootstrap ridge extends ridge regression by using bootstrap resampling to improve generalization and estimate the stability of the learned weights. It is important to note, however, that in this context we are merely attempting to fit a linear model. This does not imply that ridge regression is the optimal solution for the problem we are studying.

3.2 Fit the Ridge Regression Model

To model the relationship between word embeddings and brain activity, we applied ridge regression with cross-validation using the provided bootstrap_ridge function. We firstly loaded three types of embedding representations—Word2Vec, GloVe, and BoW—along with corresponding fMRI response data from two subjects. We then identified the common set of stories shared across all datasets and split them into training (80%) and test (20%) sets at the story level to ensure disjoint temporal structure between training and evaluation.

Each pair of embedding and subject was treated as a separate regression task. For each task, the embeddings and brain responses were z-scored (standardized) to ensure numerical stability. We then fit ridge regression models with a range of regularization strengths (α) and selected the best α using cross-validation within the bootstrap_ridge framework, which repeatedly samples chunks of the training data to evaluate performance robustness. When running the model on X_bow, due to its large data size, we handled the computation in practice by processing it in multiple batches.

During training, we monitored the average and maximum test correlation coefficients (CC) for each model. Furthermore, all CC results were stored in a DataFrame for subsequent analysis.

3.3 Evaluation

Across all embeddings, the mean correlation coefficients are between 0.05 and 0.06. For instance, the mean CC for Word2Vec embeddings is about 0.586 and mean CC for Glove embeddings is about 0.521. The overall mean CC was approximately 0.054, with a median CC of 0.051. The top 5% and top 1% of voxels achieved CCs of 0.296 and 0.411, respectively. These values suggest that the ridge regression models were able to capture a modest relationship between linguistic embeddings and voxel-level fMRI responses.

Furthermore, we identified Word2Vec as the best-performing embedding and performed a more detailed evaluation by examining the distribution of the CC across voxels. The shape of the distribution in Figure 1 is approximately Gaussian and centered slightly above zero. Most voxels have CCs close to 0, indicating that

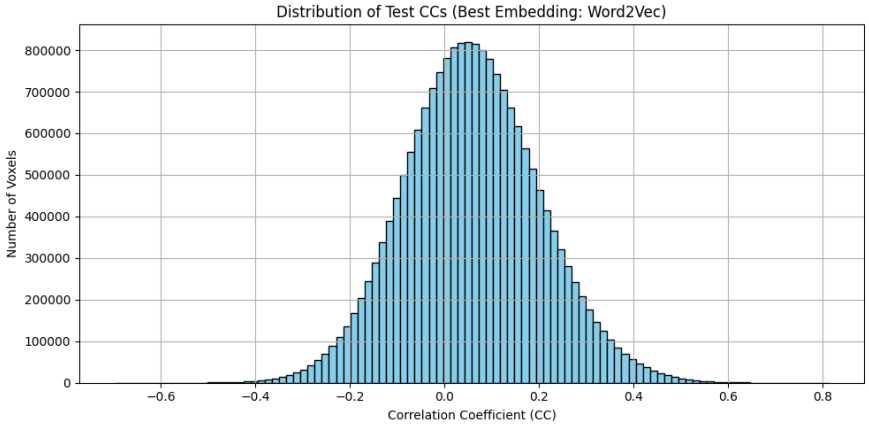

Figure 1: Distribution of W2V CCs across voxels

for a large portion of the brain, the linear model is only weakly predictive of voxel activity. A notable subset of voxels achieves moderate to high CCs, which are likely to correspond to brain regions where language information is more robustly encoded.

In conclusion, The broad range of correlation values reinforces the idea that neural encoding of linguistic information is spatially selective—only certain voxels exhibit strong model alignment. A reasonable interpretation threshold might be set at top 1–5% of CCs, based on the summary statistics we have computed.

3.4 Stability Analysis

To evaluate the stability of the regression model, we compared performance across both subjects and across different test stories. Specifically, for each embedding type, we computed the mean CC separately for subject2 and subject3. The results showed only small differences in mean CC between the two subjects—typically within 0.003—which suggests that the model generalizes well across individuals.

Furthermore, we examined the variation in performance across stories by tracking the mean CC per story within each subject. We found that while some stories yielded slightly higher or lower performance, the distribution of CCs was generally consistent, with no single story causing a dramatic change in model accuracy.

Taken together, these observations indicate that the model's performance is relatively stable across both subjects and stories.

4 Conclusion

In this study, we evaluated the effectiveness of three different word embedding methods—Bag-of-Words, Word2Vec, and GloVe—in predicting brain voxel-level fMRI responses to natural language stimuli. By applying ridge regression with bootstrap-based cross-validation, we estimated model performance across two subjects using multiple correlation-based metrics. Among the three methods, Word2Vec consistently achieved the highest mean and median correlation coefficients, as well as the highest top 1% and top 5% CCs, suggesting its superior ability to align linguistic features with neural representations.

A detailed analysis of voxel-wise correlation coefficients revealed a roughly Gaussian distribution centered near zero, indicating that while most voxels had weak predictability, a distinct subset showed strong alignment with the language model. This highlights the spatial selectivity of linguistic encoding in the brain, consistent with previous findings. Setting an interpretability threshold at the top 1–5% of voxel CCs provides a reasonable PCS-based approach to identifying voxels that meaningfully reflect linguistic structure.

Our stability analysis showed robust model performance across both subjects and across different stories, supporting the generalizability of our findings. Overall, this work contributes to understanding the relationship between semantic representations and brain activity, and reinforces the utility of distributional embeddings like Word2Vec in cognitive neuroscience applications.

References

[1] Shailee Jain and Alexander G Huth. "Incorporating context into language encoding models for fMRI". In: *Advances in Neural Information Processing Systems*. Vol. 31. Curran Associates, Inc., 2018, pp. 6628–6637.

A Academic honesty

A.1 Statement

We, the members of this group, make the following truthful statements:

The data analysis process in this report was designed and executed by our group collectively, with each member contributing their expertise to different aspects of the work. All texts and figures were produced by our group members, and the analysis steps were fully documented to ensure the reproducibility of the results. Contributions from each group member have been clearly indicated throughout the report.

All citations of other people's research have been clearly marked with the source. We understand that academic integrity is the cornerstone of research, and ensuring the authenticity and reproducibility of research is a responsibility not only to ourselves individually, but also to our colleagues and the broader scientific community.

We recognize that plagiarism not only fails to contribute to knowledge innovation but also infringes upon the rights and interests of original authors. Therefore, we are collectively committed to upholding the principles of honest, transparent, and reproducible research at all times.

As a group, we have reviewed each other's work to maintain consistency with these principles, and we stand by the integrity of our collaborative efforts presented in this report.

A.2 LLM Usage

This project was completed collaboratively by our group, with all data analysis, writing, and visualization being performed by group members. Large Language Models (LLMs) including ChatGPT and Claude were used as complementary tools to:

- Provide inspiration and generate ideas during brainstorming sessions
- Refine explanations of complex concepts
- Improve clarity and readability of our written content
- Assist with language polishing and grammar refinement

However, all content, analyses, and conclusions presented in this work were determined by our group members, guaranteeing originality and accuracy. LLMs were not used to generate original analysis or to draw conclusions from our data. Our group maintained critical oversight of all LLM-assisted content, ensuring that the final product accurately represents our own understanding and perspectives.

The use of LLMs was limited to supporting roles that enhanced our own work rather than replacing it. We acknowledge the use of these tools in the spirit of transparency while affirming that the intellectual contribution and academic responsibility remain entirely with our group members.