Lab 3.3 - FMRI
Stat 214, Spring 2025

These instructions are specific to **Lab 3.3** and cover the background and deliverables needed to complete the assignment. Note that Lab 3 consists of three parts. Please also see the general lab instructions in **discussion/week1/lab-instructions.pdf** for information on how to set up your environment, write the lab report, and submit the final product.

Contents

1 Submission

Push to the`lab3` folder created in lab 3.1 to your `stat-214 GitHub` repository by **23:59 on Wednesday, May 14th**. Only submit **one** report per group. We will run a script that will pull from each of your GitHub repositories promptly at midnight so take care not to be late as late labs will not be accepted.

The 12-page limit is not in place for this lab. The page limit for this lab is 20 pages. The bibliography and academic honesty sections don't count towards this limit. This is an upper bound, your reports are not necessarily expected to be 20 pages.

Follow the general lab instructions in stat-214-gsi/discussion/week1/lab-instructions.pdf for more details. Please do not try to make your lab fit the requirements at the last minute! In your `lab3` folder, please provide everything (except the data) that is needed for someone else to be able to compile your report and receive the exact same pdf. Please follow the provided template for your report.

2 Academic honesty and teamwork

Academic honesty statement

We ask you to draft a personal academic integrity pledge, addressed to Bin, that you will include with all of your assignments throughout the semester. This should be a short statement, in your own words,

that the work in this report is your own and that all sources you used are properly cited, including your classmates. Please answer the following question: Why is academic research honesty necessary? If you feel it is not, make a clear argument why not.

Collaboration policy

Within-group: In a file named `collaboration.txt` in your `report` directory, you must include a statement detailing the contributions of each student, which indicates who worked on what. After the labs are submitted, we will also ask you to privately review your group members' contributions.

With other people: You are welcome to discuss **ideas** with the course staff or other students, but your report must be written up and completed by your group alone. Do not share code or copy/paste any part of the writeup. If you discuss with other students, you must acknowledge these students in your lab report.

LLM usage policy

You are allowed to use LLMs (ChatGPT, GitHub Copilot, etc.) to *assist* in this lab, but are not allowed to use it for more than creating visualizations or helping correct grammar in the report. If we have reason to believe an LLM wrote a significant portion of your code (more than 5%) without your editing or iteration, or any section of your report word-for-word, this will constitute an honor code violation.

3 Lab 3.3 Overview

In Lab 3.1 and 3.2, we used pre-trained embeddings and trained our transformer respectively to predict BOLD scores for voxels. In this lab, you will do first take a pre-trained BERT model and fine-tune it on this dataset. Then, you will interpret the pre-trained model by understanding what words correspond to the prediction using SHAP and LIME scores.

Lab 3.3 Instructions

Data

Refer to instructions from lab 3.1 to get the data. The following are the instructions for the two main parts of the lab. Please carefully document your analysis pipeline, justify the choices you make, and place your work within the domain context. You should have an introduction and conclusion section in your report.

Part 1: Fine-tuning (50% of grade)

1. Using the following pre-trained BERT model `https://huggingface.co/google-bert/bert-base-uncased`, fit a ridge model on the embeddings as you did in 3.1 to predict across voxels.
2. Next, fine-tune this model for this task.
3. Investigate a parameter-efficient approach to fine-tuning this model such as Low-rank adaptation (LORA). Please refer to the following link as to how to to use LORA with Huggingface `https://huggingface.co/docs/diffusers/en/training/lora`
4. Compare the performance of the fine-tuned model to the pre-trained model and the ones you derived in 3.1 and 3.2. Specifically, perform a detailed analysis as in earlier labs to compare the performance of the embeddings across voxels.

Part 2: Interpretation (50% of grade)

Given this (fine-tuned) BERT model, we will now interpret the model by investigating the word importances for various voxels. Specifically, do the following.

- For a given test story, identify the voxels where the model performs well.
- For these voxels, run SHAP (`https://shap.readthedocs.io/en/latest/`) and LIME to (`https://github.com/marcotcr/lime`) to identify influential words that strong affect the response. Why do we only do this for the voxels that we perform well for? Think about the "P" in PCS.
- Compare the words discovered by SHAP and LIME, and visualize them. Do these words make intuitive sense to you? How are they different? How are the words discovered different across voxels?
- Repeat this analysis for another test-story.