Cognitive Science and AI: Assignment 3

Textual Brain Encoder and Decoder

February 25, 2022

The brain encoding problem aims to automatically generate fMRI brain representations given a stimulus. The brain decoding problem is the inverse problem of reconstructing the stimuli given the fMRI brain representation. In this assignment, your task would be to construct an encoder as well as a decoder for textual stimuli. Details about the dataset, methodology and the tasks are provided in the section below.

1 Dataset

The dataset to be used for the assignment can be downloaded from here. Dataset consists of 627 sentences and the corresponding fMRI, recorded when the sentences were presented to a subject one by one. fMRI is provided for four different brain ROIS listed below:

- **Language:** Related to language processing, understanding, word meaning, and sentence comprehension
- **Vision:** Related to the processing of visual objects, object recognition
- **Task Positive:** Related to attention, salience information
- **Default Mode Network (DMN):** Linked to the functionality of semantic processing

Dataset contains three files:

- **stimuli.txt :** It contains 627 sentences, each in one line.
- **subj1.npy :** Contains fMRI data for Subject 1. Stored as a dictionary, with keys as the four brain regions and values as the corresponding fMRI for 627 sentences.
- **subj2.npy :** Contains fMRI data for Subject 2. Stored as a dictionary, with keys as the four brain regions and values as the corresponding fMRI for 627 sentences.

The data in the npy files can be accessed using the following python code:

```
"data = np.load("subj1.npy").item()"
```

2 Tasks

2.1 Brain Decoder

First task is to build 4 different decoders one for each brain ROI. This needs to be done for each subject. For example, Decoder for Vision area of subject 1 would predict the sentence representations given the vision area voxels of subject 1.

The decoder could be simply a regression based model like ridge/lasso regression or any of your choice. You should perform a k-fold cross validation, and report the average of the evaluation metrics across folds.

You need to evaluate your decoders using both the evaluation metric listed below.

2.1.1 Evaluation Metrics

Given a subject and a brain region, let N be the number of samples. Let $\{Y_i\}_{i=1}^N$ and $\{\hat{Y}_i\}_{i=1}^N$ denote the actual and predicted sentence vectors for the i^{th} sample. Thus, $Y \in R^{N \times D}$ and $\hat{Y} \in R^{N \times D}$ where D is the dimension of each sentence vector.

2V2 Accuracy is computed as follows.

$$\text{2V2Acc} = \frac{1}{N_{C_2}} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} I[\{cosD(Y_i, \hat{Y}_i) + cosD(Y_j, \hat{Y}_j)\} < \{cosD(Y_i, \hat{Y}_j) + cosD(Y_j, \hat{Y}_i)\}]$$

where $cosD$ is the cosine distance function. $I[c]$ is an indicator function such that $I[c] = 1$ if c is true, else it is 0. The higher the 2V2 accuracy, the better.

Pearson Correlation (PC) is computed as PC=$\frac{1}{N} \sum_{i=1}^{n} corr[Y_i, \hat{Y}_i]$ where corr is the Pearson correlation function.

2.2 Brain Encoder

Second task is to build 4 different encoders one for each brain area. This is to be done for each subject. For example, Encoder for Vision area of subject 1 would predict the vision area voxels of subject 1 given the sentence representations.

Similar to decoder, encoder could also be simply a regression based model like ridge/lasso regression.You should perform a k-fold cross validation, and report the average of the evaluation metric across folds.

You need to evaluate your encoders using both the evaluations metric listed below.

2.2.1 Evaluation Metric

Given a subject and a brain region, let N be the number of samples. Let $\{Y_i\}_{i=1}^N$ and $\{\hat{Y}_i\}_{i=1}^N$ denote the actual and predicted voxel value vectors for the i^{th} sample. Thus, $Y \in R^{N \times V}$ and $\hat{Y} \in R^{N \times V}$ where V is the number of voxels in that region.

2V2 Accuracy is computed as follows.

$$2\text{V2Acc} = \frac{1}{N_{C_2}} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} I[\{cosD(Y_i, \hat{Y}_i) + cosD(Y_j, \hat{Y}_j)\} < \{cosD(Y_i, \hat{Y}_j) + cosD(Y_j, \hat{Y}_i)\}]$$

where $cosD$ is the cosine distance function. $I[c]$ is an indicator function such that $I[c] = 1$ if c is true, else it is 0. The higher the 2V2 accuracy, the better.

Pearson Correlation (PC) is computed as PC=$\frac{1}{N} \sum_{i=1}^{n} corr[Y_i, \hat{Y}_i]$ where corr is the correlation function.

3 Sentence Representations

You have to extract the sentence embeddings from transformer based language models like BERT/RoBERTa. You refer to this link for how to extract sentence embeddings from BERT. You can work the CLS token or the pooled embeddings of all the words in the sentence.

For every sentence, you should finally have a D sized vector.

4 Deliverables

You should report the results on the bar plot, comparing the evaluation metrics for different brain ROIS. One bar plot each for 2v2 Accuracy and Pearson correlation per subject comparing the metrics for the brain areas. This needs to be done for both the tasks. You are expected to provide some insights about the results.

You need to submit a tar file named <roll_no.>.tar consisting of two files:

- **code.ipynb** - consisting of the code
- **report.pdf** - consisting of the bar plots and insights from the results.

IMPORTANT: Make sure that the assignment that you submit is your own work. Do not copy any part from any source including your friends, seniors or the internet. Any breach of this rule could result in serious actions including an F grade in the course.