Understanding Imagined Speech using EEG

Barak Oshri (boshri), Manu Chopra (mchopra) and Nishith Khandwala (nishith)

1. DATA

In the preliminary experiment, the pair of syllables "ba" and "ku" were tested in alternate to the pair of syllables "im" and "si" twice. Each rotation of a pair involved testing each word 100 times with the two words randomly sorted within the 200 trials. Two differently pitched tones would signify the syllable that the subject would imagine, the lower tone corresponding to "ba" or "im" and the higher tone corresponding to "ku" or "si".

After processing out the artifacts, we had to trim out the part of the trial (total of 2.5 seconds each) that wasn't related to thinking about the syllable. The tone was ramped up at the 0th second of the trial, and would ramp down by 0.3 seconds. The subject supposedly begins to prepare to utter the correct word at 0.5 seconds, thinks of the imagined word for 1.5 seconds, and rests for the last 1. Given that there is no method to calculate the exact amount of the trial that is irrelevant to the thinking of the syllable, a tradeoff must be made about whether the data should be trimmed conservatively or non-conservatively. Since it is more ideal to discard some relevant data than to include irrelevant data points, we will make conservative estimates by cutting out the first 0.5 and last 1 second of each trial.

The EEG cap collected data at 64 channels covering the entirety of the subjects head. Three additional nodes tracked eye and upper-facial movements to assist removing blinking and face movement artefacts from the data. The EEG sampled at a rate of 500HZ. Data processing identified that 134 of the 800 trials recorded were too noisy.

2. APPROACH

Machine learning using EEG data is typically divided among two methods in literature: clustering and supervised classifiers.

2.1 Clustering

The chief benefit of using clustering methods is that they can identify localized function in the brain. Where Support Vector Machines and Naive Bayes, for example, lose idiosyncratic patterns to generality, clustering methods, given the right parameters, can identify hidden structure that is lost in the uniformity of a binary classifier.

Clustering methods are especially relevant to this project. Given how little is known about where language is thought, let alone how thoughts are formed or what their neural features are, little can be assumed about the patterns of the data's structure, and clustering algorithms offer the highest-level, least assumptive methods.

K-Means clustering has proven successful in a range of epilepsy classifier tasks. Wavelet coefficients are obtained by decomposing the EEG signal using a Discrete Wavelet Transform (DWT), which the K-Means algorithm clusters using the Euclidean distance. DWT uses non-stationary time series analysis and leads to good time-frequency localization by using longer time windows at low frequencies and vice versa. The K-Means is used to compute probability distributions for each subband, and feed it into a Multilayer Perceptron Neural Network (MLPNN). The MLPNN aptly captures the structure of the problem by identifying nonlinear patterns that correspond to representation of the word in the mind. The MLPNN will be trained with a least squares backpropagation algorithm.

Comparing K-Means with K Nearest Neighbors (KNN) will provide important insight into the data. It has been reported that KNN performs better in classification of EEG signals [3], perhaps because function in the brain is less organized or uniform than we think. In such a case, it is expected that KNN will perform better because it doesn't make assumptions about any superstructure of the data, while K-Means assumes that there is a statistically significant level of difference between voicing different syllables in the mind. Comparing these two algorithms will thus inform us somewhat on the anatomization of words in imagined speech. Our results with K-Means have been noisy and we are investigating KNN and other models.

2.2 Classifiers

Classifiers, especially Support Vector Machines (SVMs), are considered the state-of-the-art approach in EEG classification. In this project, we will use and compare Gaussian Discriminate Analysis (GDA) and SVM.

A GDA is particularly useful because it involves probabilistic methods that are not as explicitly used in other algorithms such as SVM or K-Means. Probabilistic classification is especially important in Brain Computer Interface (BCI) where the interface should interact not only with what it predicts the subject is thinking, but to the degree with which it predicts it [2]

GDA in EEG classification tasks define the Gaussian Prior by feature extracting over a covariance matrix which is defined via a kernel function. That is,

$$f(X) = \mathcal{N}_f(0, C_\theta(X, X)) \tag{1}$$

where X is a $D \times N$ matrix corresponding to N data points of D dimensions. The initial kernel we will use to define the covariance matrix is the Gaussian kernel on our wavelet coefficients after extraction:

$$C_\theta(X, X)_{ij} = exp(-\phi \sum_{d=1}^{D} (x_{id} - x_{jd})^2 + \lambda) \tag{2}$$

paramaterized by $\theta = (\phi, \lambda)$. We proceed with ordinary GDA as discussed in lecture.

A Support Vector Machine with a quadratic kernel function could be used to used as a final classifier. Previous works [1] have successfully classified imagined speech syllables using this SVM model 5 fold cross validation, where the training and testing set are kept separate.

3. CHANNEL SELECTION

3.1 Manual Channel Selection

Neuroscience research has proven that different brain regions control different human behaviours. Imagined speech is believed to activate the frontal cortex as well as Broca's and Wernicke's areas. The electrodes that are distant from the active region, such as those directly at the top or back of the head, may not provide any relevant information. Therefore, in addition to removing useless data, discarding these electrodes would furthermore considerably reduce the number of electrodes.

But, as we know, exact coordinates of the active regions cannot be provided. Inspired from a recent paper on EEG signal classification, we can select 10 electrodes roughly near the possible active region. Previous works similar to this have shown that even such an imprecise setup can still achieve reasonable results.

3.2 Automatic Channel Selection for Imagined Speech

Alternatively, the electrodes may be automatically selected based on the information in their signals. EEG is known to have poor spatial resolution, so adjacent electrodes tend to be highly correlated. The channels may therefore be clustered based on the correlation between electrodes, and this clustering may also reveal location information about where stronger signals may be found.

4. IMPLEMENTATION SPECIFICS

Although the area of BCI is particularly new and an evolving research field, we were able to find a couple of libraries that would be immensely helpful in interpreting and experimenting with EEG data. For the purposes of this project, we plan to use a combination of EEGLAB and BCILAB, which are MATLAB toolboxes developed by the University of California, San Diego. In the following sections, we explain the role of each toolbox in our implementation.

4.1 EEGLAB

EEGLAB is an open source environment for EEG signal processing and data processing. The advantages of using EEGLAB are multifold:

- It supports multiformat data importing - which is crucial to our team since our data is composed of raw 2D MATLAB arrays and Neuroscan's .CNT files. EEGLAB helps process and convert both these formats into one usable by both BCILAB and other EEG-analysis libraries (in Python or otherwise).
- It provides in-built tools to remove artifacts like blinking and muscle movement (like heart beats). Raw EEG data tends to be particularly very noisy and removing artifacts is usually the first step into processing EEG.
- It provides functionality to interactively plot and visualize the EEG across channels. Since the data collection process is standard and the EEG cap has marked electrodes, it is possible to see what part of the brain is active during experimentation.

4.2 BCILAB

BCILAB is a MATLAB toolbox and EEGLAB plugin for the design, prototyping, testing, experimentation with, and evaluation of Brain-Computer Interfaces (BCIs). Primarily, we will be using BCILAB for the variety of machine learning libraries it offers specifically for analyzing EEG data.

5. FURTHER WORK AND RESULTS

As of today, we have implemented the K-means clustering algorithm without the MLPNN classifer. However, we obtained chance results from this, namely clusters that are not significantly deviant from each other. As described above, this could be naturally because the data is not disposed to clustering, or because we have not defined the right features to perform analysis on. We are currently looking into more specific and better defined feature extractors for our assignment, which include processing and analyzing the α, β, γ waves, doing channel-wise comparisons between the data points to filter out the channels that are more expressive, and using parametric methods (such as an Autoregressive Model) to characterize the EEG signals. Given these extractions, we hope to run the algorithms discussed above, and other methods such as Independent Component Analysis.

6. REFERENCES

1. Advanced Signal Processing and Machine Learning Approaches for EEG Analysis
   `http://www.researchgate.net/publication/235054600_Advanced_Signal_Processing_and_Machine_Learning_Approaches_for_EEG_Analysis`
2. Classifying EEG for Brain Computer Interfaces using Gaussian Process
   `http://people.rennes.inria.fr/Anatole.Lecuyer/prl08_zhong.pdf`
3. Comparison Machine Learning Algorithms for Recognition of Epileptic Seizures in EEG
   `http://iwbbio.ugr.es/2014/papers/IWBBIO_2014_paper_1.pdf`
4. Toward EEG Sensing of Imagined Speech
   `http://link.springer.com/chapter/10.1007/978-3-642-02574-7_5#page-2`