Supplement for: Classifying spoken syllables from human sensorimotor cortex with deep networks

Anonymous Author(s)
Affiliation
Address
email

1 Phonetic Organization from HMM-GMM

Figure 1 shows the results of hierarchical clustering on the confusion matrix from the HMM-GMM model. The errors the model makes capture the coarse relationships between syllables but the fine relationships are not apparent. Compared to the results from clustering on the DNN outputs in the main text, we find fewer of the phonetic relationships between syllables represented in the confusions of the HMM-GMM.

2 Cross-Validation of HMM-GMM

The HMM training starts with a uniform segmentation of observations followed by a Baum-Welch estimation of model parameters. The segmentation and model parameters are iteratively updated to converge untill segmentation remains unchanged across iterations. Since this method is prone to local minima, we report a 10-fold cross validated performance across different train/test splits to generalize the performance of the HMM-GMM classifier. Though the mean trends are indicative, the high variance in the mean reciprocal rank across folds (also seen in the cross-fold variance in DNN performance, Main Text: Figure 2(b)) suggests a train-test mismatch. One avenue for future investigation is appropriate pre-processing and normalization to improve consistency across trials, thereby addressing the issue of train/test mismatch. A 5-state HMM (2 non-emitting begin and end states, as shown in figure 2) was chosen, since higher numbers of states failed to segment the data. Other candidates evaluated with the HMM-GMM model include

- models for full syllable, without the phoneme split (e.g., /ba/ as the token as opposed to the sequence /b/ /a/),
- initial and final silence on either side of the syllable (e.g., *sil* /b/ /a/ *sil*).

These candidates, however, do not outperform the 5-state phoneme model.

A range of values (2,4,8,16 and 32) were evaluated for the number of Gaussian mixtures per state. The mean reciprocal rank of the true class from among 57 classes is computed for 10% unseen trials across 10 folds. This measure lies between [0,1] (greater value indicates better performance) is chosen, since it is more sensitive than classification accuracy for this model. As reported in Fig 3, the value of 8 mixtures is chosen for comparison with other models.

3 Additional Experimental Methods

The analysis window started 500ms before the acoustic onset of the consonant-to-vowel transition, and extended 800ms after. The analysis window of 1.3 seconds was chosen through extensive previous analysis of this data set. Specifically, the previous analysis of this data has demonstrated that the class information in the neural signals drops to near chance levels at the beginning and end of this time window.

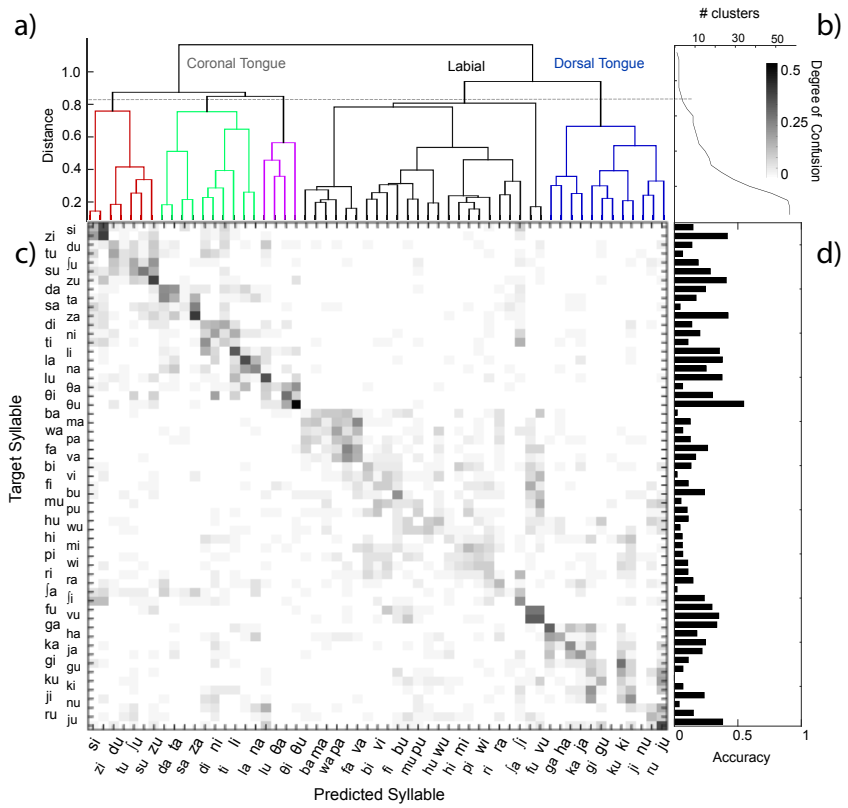

Figure 1: Analysis of the HMM-GMM outputs. **a)** Dendrogram showing hierarchical clustering based on the softmax outputs. **b)** The trend of the number of discovered clusters as a function of minimum allowed distance between clusters. The dashed line represents the distance .82 used for the clusters identified in panel **a**, which is in a relatively flat region of the number of clusters vs. threshold relationship. **c)** The confusion matrix shows the structure of the errors that the network makes. **d)** Mean accuracy per syllable across folds.

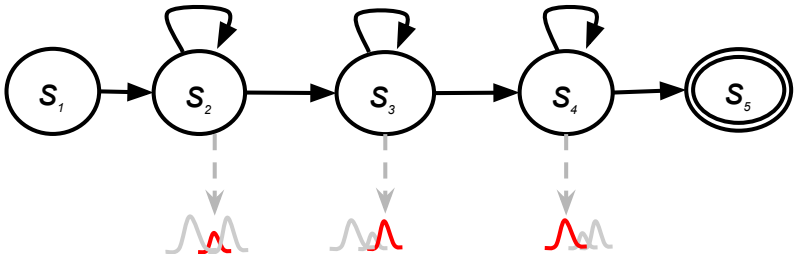

Figure 2: Illustration of a 5-state HMM with 3 emitting states

4 Classification on Acoustic Features

The ECoG data on which the classifiers were trained on, as described in the main text, encodes commands for motor control of the speech articulators. Articulator movement does not have a simple mapping to sounds produced or syllable class. The phonetic classes are most closely identifiable with the audio data collected in tandem to the ECoG data. Therefore, the classification performance on the acoustic features is a rough upper limit in performance achievable on ECoG data. 25 dimensional Mel frequency cepstral coefficients (24 dimensions+ 1 dimension of energy per frame) were extracted from 16000 Hz audio at a frame spacing of 25 millisecond and frame shift of 5 millisec-

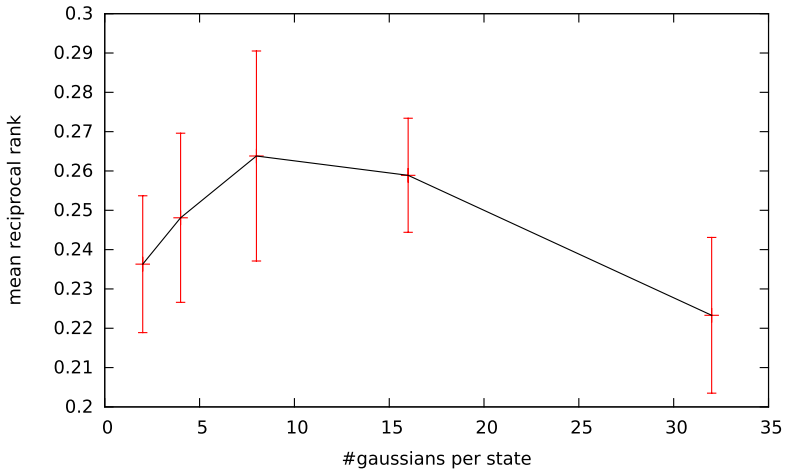

Figure 3: Mean reciprocal rank across 10-fold cross-validation for HMM-GMMs, also shown is the standard deviation across folds. A maximum is achieved at 8 mixtures per state.

ond. Additionally velocity features are included to account for spectral dynamics, thus giving 50 dimensional feature vectors per every 5 milliseconds. Table 1 shows the performance of fully connected networks on the audio features. As expected, they perform much better than the classifiers trained on ECoG data.

Table 1: Classification and ITR Results on Audio Features

Model	**Classification Accuracy**	**Folds over Chance**	**Approximate ITR (bits per second) Exact in parenthesis**
Fully Connected, 57 CV	$90.6 \pm 2.2\%$	53x	4.8 (5.3 exact)
Fully Connect Cons., 19 cons.	$88.2 \pm 0.8\%$	18x	3.3 (3.6 exact)
Fully Connected Vowel, 3 vowels	$97.3 \pm 1.0\%$	3x	1.4 (1.4 exact)

5 Performance of Convolutional Networks

The poor performance of convolutional networks could be due to this dataset not following the usual assumptions that explain the performance of CNNs. The utterance class depends on the precise timing of articulatory commands in some voiced versus unvoiced syllables, e.g. /ba/ versus /pa/ or /gi/ versus /ki/. Convolution and pooling may throw away this information. To try and test this hypothesis, we compared the percent of phonetically similar errors (5 consonant pairs comparisons with 3 different vowels) the best fully connected network makes compared to the best performing convolutional network, both trained without data augmentation. We find that $16.2 \pm 2.0\%$ of the errors the fully connected makes are between similar phonemes. Only $14.0 \pm 2.2\%$ of the errors convolutional networks make are between phonetically similar phonemes, which does not support this hypothesis. It might also be the case that the temporal alignment removed all time translation symmetry and the temporal statistics might not be stationary making shared local filters ineffective. Finally, it is also possible that convolutional networks will require more data to surpass fully connected networks.

6 Optimized Hyperparameters for DNNs

For the fully connected and convolutional networks, a number of architecture and training parameters were optimized through cross-validation. The full list of optimized hyperparameters is in Table

2. The first set were not optimized over, but important for specifying model architecture and training. The second set were shared between fully connected and convolutional networks and the third set are unique to convolutional networks. Hyperparameter types were either drawn from a range of integers (Int), a range of floating point values (Float), or from a list of options (Enum). Hyperparameters that might range over many orders of magnitude were optimized in $\log_{10}$ space. Additionally, some parameters were selected as their differences from 1 (one-minus) or the $\log_{10}$ of their distance from 1. All networks were trained using stochastic gradient descent with Nesterov momentum. The main results were found by running 200 trials with different sets of hyper-parameters, but optimal hyper-parameters were often found with fewer than 50 iterations.

Table 2: Hyperparameters for DNNs

Name	Type	Range/Options
Init. Momentum	Float	.5 (Fixed)
Conv. Layer Type	Enum	RectifiedLinear (Fixed)
Terminate After No Improvement Epochs	Int	10
Num FC Layers	Int	1 : 4
Up to 3× FC dim	Int	3 : 1000
FC Layer Type	Enum	RectifiedLinear, Tanh, Sigmoid
Cost Type	Enum	Cross-Entropy, Hinge L1, Hinge L2
$\log_{10}$ Weight Init Scale	Float	-5 : 0
$\log_{10}$ Learning Rate (LR)	Float	-3 : -1
$\log_{10}$ Min. LR	Float	-5:-1
$\log_{10}$ One-minus LR Decay	Float	-5 : -1
$\log_{10}$ One-minus Final Momentum	Float	-2 : -3.0102e-1
Momentum Saturation Epoch	Int	1 : 50
Batch Size	Int	15 : 256
Max Epochs	Int	10 : 100
One-minus Input Dropout Rate	Float	3.0e-1 : 1
Input Rescale	Float	1 : 3
One-minus Default Dropout Rate	Float	3.0e-1 : 1
Default Rescale	Float	1 : 3
$\log_{10}$ L2 Weight Decay	Float	-7 : 0
Max Filter Norm	Float	0 : 3
Num Conv. Layers	Int	1 : 3
Up to 3× Num. Filters	Int	8: 128
Up to 3× Filter Shape	Int	3: 50
Up to 3× Pool Shape	Int	3: 50
Up to 3× Pool Stride	Int	3: 50
Max Conv. Filter Norm	Float	0 : 3

7 Hyperparameters for Best Network

The hyperparameters for the fully connected and augmented network are shown in Table 3. Models with nearly equivalent performance could be found with 200 to 1000 hidden units and up to two hidden layers. Performance was sensitive to the number of layers, learning rate, weight initialization, regularization parameters, the nonlinearity of the final layer, and the associated cost function used to train the network.

Table 3: Hyperparameters for Best Network

Name	Type	Range/Options
Init. Momentum	Float	.5 (Fixed)
Terminate After No Improvement Epochs	Int	10
Num FC Layers	Int	1
FC dim	Int	798
FC Layer Type	Enum	Tanh
Cost Type	Enum	Cross-Entropy
$\log_{10}$ Weight Init	Float	-2.70
$\log_{10}$ Learning Rate (LR)	Float	-2.48
$\log_{10}$ Min. LR	Float	-3.92
$\log_{10}$ One-minus LR Decay	Float	-3.21
$\log_{10}$ One-minus Final Momentum	Float	-0.83
Momentum Saturation Epoch	Int	47
Batch Size	Int	163
Max Epochs	Int	58
One-minus Input Dropout Rate	Float	0.44
Input Rescale	Float	1.61
One-minus Default Dropout Rate	Float	0.70
Default Rescale	Float	2.2
$\log_{10}$ L2 Weight Decay	Float	-3.06
Max Filter Norm	Float	0.66