Brain2Speech

Speech reconstruction from
direct brain signals
by using NeuralNetwork

Gergely Zoltan Gulyas – EH9JTC
Krisztian Imre – GSE1U4
Budapest University of Technology and Economics

Purpose

Speech is the primary and most essential means of human communication. However, many people have lost this ability through illness or ill health. The real-time synthesis of speech directly from measured neural activity (BRAIN2SPEECH) would enable natural speech and significantly improve quality of life, especially for severely communication-impaired individuals.

Within the project, students study the BRAIN2SPEECH domain and then develop and train new types of neural network architectures. The suggested data is the ‘Dataset of Speech Production in intracranial Electroencephalography’ (SingleWordProductionDutch), available at https://osf.io/nrgx6/.

The original github repository contains a linear regression model which should be replaced by deep neural networks. Basic task: train/valid/test set from single speaker. Advanced task: cross-speaker train and synthesis.

Original Work:

https://github.com/neuralinterfacinglab/SingleWordProductionDutch

https://www.nature.com/articles/s41597-022-01542-9#Tab1

M. Verwoert, M. C. Ottenhoff, S. Goulis, A. J. Colon, L. Wagner, S. Tousseyn, J. P. van Dijk, P. L. Kubben, and C. Herff, “Dataset of Speech Production in intracranial Electroencephalography,” Scientific Data 2022 9:1, vol. 9, no. 1, pp. 1–9, jul 2022. [Online]. Available: https://www.nature.com/articles/s41597-022-01542-9

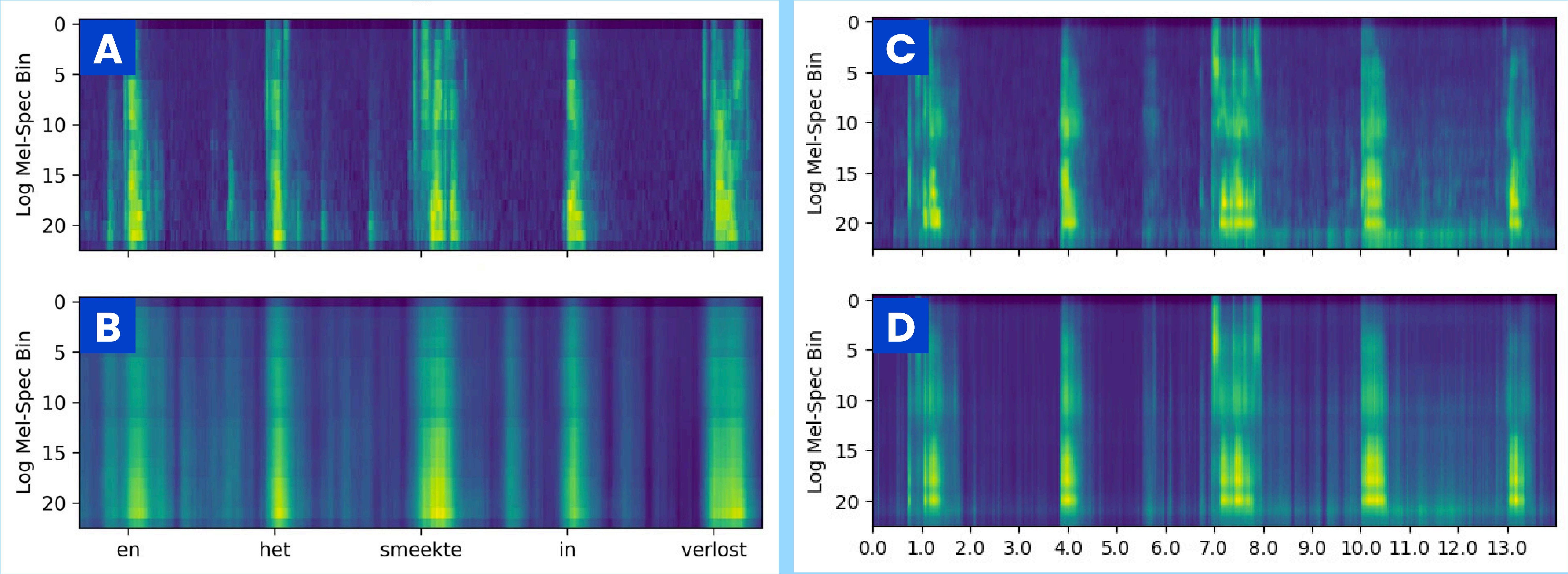
Bayram, I. An analytic wavelet transform with a flexible time-frequency covering. IEEE Transactions on Signal Processing 61, 1131–1142 (2012).

Our Solution

We used a FC-DNN with descending numbers of nodes in its four hidden layers, each followed by ReLU normalisation. We used five-fold crossvalidation during the training:

https://github.com/eva-vision/2BRAINS

Our solution created a more detailed LogMEL spectrogram, than the regression in the original work of the fellow Dutch researchers:



A-C: Original spectrogram created from recorded speech,
B: Solution using simple regression, D: Our soulution with FC-DNN regression

```
# Define the neural network architecture
class TimeSeriesNet(nn.Module):
    def __init__(self):
        super(TimeSeriesNet, self).__init__()

        # Input layer
        self.input = nn.Linear(1143, 512)

        # Hidden layers
        self.hidden1 = nn.Linear(512, 256)
        self.hidden2 = nn.Linear(256, 256)
        self.hidden3 = nn.Linear(256, 128)
        self.hidden4 = nn.Linear(128, 64)

        # Output layer
        self.output = nn.Linear(64, 23)

        # Batch normalization layers
        self.bn_input = nn.BatchNorm1d(512)
        self.bn_hidden1 = nn.BatchNorm1d(256)
        self.bn_hidden2 = nn.BatchNorm1d(256)
        self.bn_hidden3 = nn.BatchNorm1d(128)
        self.bn_hidden4 = nn.BatchNorm1d(64)

    def forward(self, x):
        # Input layer with activation and normalization
        x = F.relu(self.bn_input(self.input(x)))

        # Hidden layers with activation and normalization
        x = F.relu(self.bn_hidden1(self.hidden1(x)))
        x = F.relu(self.bn_hidden2(self.hidden2(x)))
        x = F.relu(self.bn_hidden3(self.hidden3(x)))
        x = F.relu(self.bn_hidden4(self.hidden4(x)))

        # Output layer
        x = self.output(x)
        return x
```