Phase 2: EEG Analysis

In this second part of the P&O, we will look into **auditory attention detection (AAD)** based on EEG recordings. The goal is to build the second component of the system, that is capable of determining which of the two voices a subject is listening to. Combined with the component developed in phase 1, which allows to 'zoom in' on one of the two speakers using beam forming, this will then be integrated in an online demo in the final project phase.

AAD based on EEG recordings is a relatively new field, and comes with significant challenges. Indeed, the input data is high-dimensional and very noisy. Based on visual inspection, it is usually impossible to tell which speaker a subject is attending to.

Whereas the first phase focused on a signal processing-based approach, for the AAD part we will rely on **machine learning** techniques to analyze the signals. That is, starting from labeled training data, we will learn the system to map the inputs (EEG-analysis and corresponding audio signals) to the correct output. We will explore not just one, but two alternative schemes to do so:

- first, we will experiment with a scheme based on **linear regression** that tries to reconstruct the spatial envelope of the attended audio signal from the EEG signals, compares this with the two original audio signals and selects the one that shows the highest correlation.
- Next, we will explore using a **deep learning** solution that directly classifies the inputs into two cases: those where the subject is listening to the speaker on the left vs. those where he/she pays attention to the speaker on the right.

The goal of this phase is to implement both approaches and compare them. For the linear regression-based approach, only a high-level description will be given, together with some references to the literature. That should suffice to implement this approach from scratch.

For the deep learning solution, we will start with example code of the most standard application of convolutional neural networks, namely image classification for digit recognition. Once you master this code, you will move on to the classification of the EEG signals, for which a basic framework will be given, but the details of the architecture will need to be filled in by you.

The work in this second phase is spread over 10 sessions, and consists of the following aspects (with a rough indication of the time dedicated to each):

- Making your own EEG recordings (1 session)
- Preprocessing the data (1 session)
- AAD based on linear regression (1-2 sessions)
- Tutorial on training neural nets (1 session)
- AAD based on a neural network (2-3 sessions)
- Further optimization (2 sessions)
- Evaluation and reporting (1 session)

Documentation

On Toledo -> “Course Documents” -> “Phase 2: EEG Analysis” -> “Documentation” you can find the following documents to get you started (in the order you should look into them):

- *1 Phase 2 top level.pdf*: this document, which is the starting point for this phase and has all practical information. Read this first.
- *2 EEG document.pdf*: a short introduction to EEG signals and EEG recording, including information on the two EEG recording devices used. Read this before starting implementing anything, so you are familiar with the task and the data.
- *3 Preparation EEG experiments.pdf*: instructions on how to prepare the audio for the actual recordings.
- *4 Preprocessing.pdf*: data recorded via EEG cannot be used directly, but first needs to be processed. The data you get directly from us has already been preprocessed. For data you record yourself, you have to do the preprocessing yourself. This document tells you how to do this, and gives details on the format in which the data is provided.
- *5 Linear_regression*.pdf: basic information on how to solve the problem with linear regression. From this description, and the references in the document, you should be able to implement an AAD system.
- *6 CNN document.pdf*: a document to get you started with the neural network based approach.

Data

Apart from the data you collect yourself, we also make available data we recorded previously, so you have sufficient data for training and evaluating your models. This data can be found on Toledo -> “Course Documents” -> “Phase 2: EEG Analysis” -> Data.
We will start with data recorded using a 64-channel device, which is different from the 24-channel device used for your own recordings. This will require some additional adaptations.

In particular, here’s an overview of the provided data:

- *Audio for EEG-recordings*
- *64 channel Biosemi EEG data* (preprocessed, see document *4 Preprocessing.pdf*)
- *24 channel Smarting EEG data* (not yet preprocessed, see *4 Preprocessing.pdf*)

Tools

You do not have to start the implementations from scratch. We provide code for you to get started with. For the preprocessing and linear regression, you will be working in matlab. For the deep learning part, we will use python – in particular Jupyter notebooks with Keras running on top of Tensorflow.

In particular, you can find the following code/tools on Toledo:

- *deadroom_ite_hrtfs.mat*: head-related-transfer-functions, needed to prepare the audio for the EEG recordings.
- *audioprepro_constants.mat:* parameters to be used for audio preprocessing
- *eegread.m*: function to read EEG data from the experiment
- *jupyter notebooks for EEG analysis with keras*

Practical organization

- The **recordings** take some time (roughly 3 hours, including setup). Therefore, they will be performed in larger groups consisting of 4 or 5 students (2 teams together) and will be spread over the sessions of the first two weeks.
- Apart from that, you work independently as a team, based on the provided documentation. The TAs will be around during the first 2.5 hours of each session to assist you. Keep track of your progress relative to the rough time indication given above, to make sure you finish in time.
- For the deep learning part, it is probably easiest to make use of **Google Colab**, which allows you to work in the cloud and use GPU resources provided by Google. If you do not have a Google account, or feel uncomfortable with using it for this purpose**, let us know in time so we can set up an alternative.**

Evaluation

Evaluation of phase 2 is similar to the evaluation of phase 2, and consists of:

- A demo of your system during the last session,
- A report describing your work in roughly 5 pages.

Information we want to find back in this report includes:

- A description of your solution based on linear regression
- An evaluation of your solution based on linear regression, both on the 64 and 24 channel data, and discussion of those results
- A description of your experiments with deep learning alternatives
- An evaluation of your final solution based on deep learning, both on the 64 and 24 channel data, and discussion of those results
- A comparison of the two approaches and final conclusion