Bachelor Pipeline Documentation

Mind Monitor

Version: 2.3.1

The latest version sends 2 AUX columns in addition to raw EEG data of Muse. The latest EEGEMO pipeline was fixed to match it.

Psychopy

Psychopy version 2022.2.5

Python version 3.8.16

Anaconda heavily recommended

Launch from console so that all proper modules are recognized

IP wanted by Psychopy and Mind Monitor is the IP of the network your pc is connected to. Get it from ipconfig command in cmd

EEGEMO

To run the pipeline with the psychopy you must run stream_live.py file. Before running it, start psychopy session and copy the path of the .log file it generated. Paste this log file into logs.config.

logs.config

```
[logger_stream]
level=INFO
handlers=fileHandler
qualname=file_logger


[logger_root]
level=DEBUG
handlers=screenHandler
qualname=screen_logger


[handler_fileHandler]
class=FileHandler
level=INFO
formatter=logFormatter
args=(f"C:\BachelorThesis\EEGEMO\PsychoPy_Setup\data\Lexo_study_english_2_part_2023-06-26_14h39.23.500.log",)


[handler_screenHandler]
class=StreamHandler
formatter=logFormatter
level=DEBUG
args=(sys.stdout,)


[formatter_logFormatter]
format=%(asctime)s | %(levelname)-8s | %(message)s

```

Now that your EEGEMO setup is connected to psychopy through the log file, you can run stream_live.py

You can run it without giving the existing model as an input. It will generate a model after learning from the first label given by user in Psychopy

```
python stream_live.py m 44 --sessionName FirstSession
```

Above is the command to run the live ssion for Participant 44 with session name FristSession. When terminated, it generates a dataset with recording EEG data and models for arousal and valence prediction in the Output folder.

You can use this generated model then to run the same script with it, initialising the session with an existing model

```
python stream_live.py m 44 --modelFileArousal "path to arousal model" --modelFileValence "path to valence model" --sessionName alex
```

stream_live.py generates not only the model files, but also gives as an output the .csv file with EEG values. You can run utils function on it the to make it fit for evaluation.

```
python clean_data.py "C:\BachelorThesis\EEGEMO\output_data\41_1_muse.csv" m
```

```
python label_data.py "C:\BachelorThesis\EEGEMO\output_data\41_1_muse_cleaned.csv" "C:\BachelorThesis\EEGEMO\PsychoPy_Setup\data\Alexander_s
```

Now that data is labelled, it is possible to run stream_csv.py on it to generate a confusion matrix, evaluating how well model performed. The naming and path of the file should correspond to this pattern:

```
C:\BachelorThesis\EEGEMO\input\subject_41\41_part1_muse_prep.csv
```

The command below run the evaluation on the dataset of participant 41 if such a dataset is present in the input folder with the naming like described above.

```
python stream_csv.py C:\BachelorThesis\EEGEMO\input M 41 m
```

The evaluation files should now be available at the following location:

```
C:\BachelorThesis\EEGEMO\output_data\mydata\41
```

It is also possible to run the stream_csv initialised with existing models. This way you can see how would the resulting model trained on given dataset be impoved when initialised with a certain model.

```
stream_csv.py C:\BachelorThesis\EEGEMO\input M 102 m --modelFileArousal "path to arousal model" --modelFileArousal "path to valence model"
```

Generalized models

To build a generalized user label model, one needs to first combine datasets of multiple participants into one. It is possible to do with a simple python script from the utils:

```
python combineFiles.py
```

It will prompt you to enter the path to the directory where the .csv files to combined are stored.

Running stream_csv with this new dataset will create a generalized user label model.

To create a generalized ground truth model, this combined dataset needs to be transfromed using another script from the utils

```
python changeGroundTruth.py
```

The script will prompt you to give the path to the dataset to be changed, without “”.