29/08/2023

Action detection, final project:

Dataset information:

- This is aprivate dataset of Schiller's lab at the Technion (no approval for sharing this dataset). The dataset is of mice that conduct hand-reach task that are trained on and rewarded for. The end goal of this dataset is to correlate behaviors with brain activity that is simultaneously recorded.
- Total number of videos (that are trials) = 202, with FPS = 200 and length of 2400 frames.
- The data is sorted for annotator and animal (each annotator has one animal).
- Each session has ~14 videos, with animal success and failure balanced.
- Please take into consideration that the different animals have high level of variance. Training the described tasks below can be also conducted per animal and not across all videos.
- You are also supplied with an optional data set that is with fps = 150 and total frames = 1800.
- List of behaviors of the animals that are labeled:
    - **['background', 'Perch', 'Lift', 'Reach', 'Grab_nonPellet', 'Grab', 'Sup', 'AtMouth', 'AtMouth_nonPellet', 'BackPerch', 'Table']**
- Download the dataset from the link: Dataset_Final_Project.rar

Following is the dataset structure:

```
Annotator (A, B, C, D, E, F)
|
└── Animal (1, 2, 3,..)
  |
  ├── Session (1, 2, …)
  |  |
  |  └── Trail
  |
  ├── Session
  |  |
  |  └── Trail
  |
  └── ...
```

29/08/2023

Code information:

- For installation: https://github.com/jbohnslav/deepethogram/blob/master/docs/installation.md , And my supplied environment.yml file: environment.yml

- To start running code, use the main code we supply: main_runner.py

Tasks and questions:

1. Read the paper "DeepEthogram, a machine learning pipeline for supervised behavior classification from raw pixels" (2021) and explain the following:
    a. What are the learning models (that are supplied as default) architecture and why were these architectures chosen?
    b. What are the loss functions per each architecture (that is supplied as default) and what are the pros and cons to choose these functions and to calculate separately?
    c. Describe the output of each final (output) layer of each architecture.

2. Conduct the following experiments with the dataset of the hand-reach task that we provided:
    a. Create a benchmark for the default run of the model, use the supplied metrics of F1 and confusion matrix to compare and discuss (you can add metrics as needed). Reminder: you can train separately per annotator and not across, provide results accordingly.
        i. You can experiment with different hyperparameter values, as detailed in conf.model_name.yaml (i.e., num_ephocs, dropout etc.) and optimize, not mandatory though can be highly interesting.
    b. Experiment with different architectures provided in conf.model_name[arch] and discuss what architectures better perform and why, use minimum of 3 chosen models that are provided. For example, in feature extractor, architectures provided are: resnet18 (default), resnet50 and resnet3d_34. There is a list of options in the conf files. Reminder: you can train separately per annotator and not across, provide results accordingly.
    c. Choose one of the architectures (suggested is the first generator) and change it to other architecture that will improve model results predictions (suggested to experiment with transformer), do not choose one of the architectures in part b that are already provided. Reminder: you can train separately per annotator and not across, provide results accordingly.

29/08/2023

Submission must include:

- Detailed report of all tasks
- Code with a readme file

For any questions, please feel free to contact Sivan Schwartz: sivan.s@technion.ac.il

I will not be available at 1-15/9, you can discuss with dr. Chaim Baskin for questions at this period