

Machine Learning Workshop 2022

November 25th 2022

Due date: Monday December 19th 2022 23.59h.

Goals:

- This is an exploratory workshop intended for those students who have no or little practical experience with deep neural networks applied to image and/or audio classification tasks.
- To research novel ideas for representing timeseries data as images, such that DNN-based methods originally designed for the image domain can also be used for the audio domain and other time-series data-domains (Assignment 2b)).

Grading: 0-10, where Assignment 2b constitutes 60% of the grade.

Introduction

Deep Neural Networks can be used for Audio Classification problems in many different ways. During this workshop we will explore some examples of Convolutional Neural Networks (CNN's) used for image classification and study an example of a CNN that has been adapted and used for an audio classification task.

Preparations:

1. Download the .zip file containing all data, scripts and notebooks:

```
wget <link will be given on Brightspace>  
unzip <file_name>.zip
```

Do not distribute the data and scripts any further! It is for this workshop only.

2. Make your virtual environment:

```
virtualenv ml --python=python3.9  
source ./ml/bin/activate
```

Note: to exit the virtual environment issue the command: *deactivate*

3. Install some further required packages:

```
pip install --upgrade pip  
pip install -r requirements.txt
```

NB This does the following pip installs:

```
pip install tensorflow==2.10  
pip install jupyter  
pip install pillow  
pip install numpy scipy matplotlib ipython  
pip install scikit-image  
pip install opencv-python
```

4. Start the jupyter notebook:

jupyter notebook

Start Firefox or another browser with page: *localhost:8888* (Note the port may vary.)

Assignment 1:

NB This is an exploratory workshop and should not take you more than an hour or two in total! If the code in any of the notebooks fails on your computer, instead of mentioning the results, mention the error (1 sentence per error).

The notebooks themselves are a very nice read from the highly recommended book by Francois Chollet. Deep Learning with Python [1].

NB. Be sure to close down previous notebooks before starting the next one.

- a) In the Jupyter Notebook opened in Firefox browse to the *cats_and_dogs*-directory and select the Notebook 5.1. This is an example of a neural network trained on the MNIST dataset of written characters.
Click the <run>-button for each of the pieces of python script, while reading the explanations that are given. Report the final accuracy.
- b) Open Notebook 5.2, this shows how a convolutional network can be trained on a small dataset of cats and dogs. You do not have to download that dataset as it is already available in the 2nd *cats_and_dogs*-subdirectory.
Note: you may want to skip the training of the network using data augmentation and dropout, as this takes around 15 minutes using a GPU. Instead, you can skip this step and use a previously stored trained model *cats_and_dogs_small_2.h5* to plot the results. Report the final accuracies you obtained.
- c) Notebook 5.3 shows how to use existing CNNs as feature extractors. These features can then be used to train and use a dense classifier network. Subsequently it is shown how you can fine tune a CNN for your new task. Note that training can take a long time (15 minutes on a GPU). You may want to skip that step and use the previously finetuned and model *cats_and_dogs_small_4.h5*. Report the accuracies you obtained.
- d) Notebook 5.4 shows how to visualize the activation areas of a CNN when used for inference.

Assignment 2:

- a) Activate the virtual environment *ml* from Assignment 1 and browse to the *audio_classification*-directory.

Note you *may* have to install the following components on your system.

```
sudo apt-get install libasound-dev
sudo apt-get install portaudio19-dev
sudo apt-get install python-pyaudio
sudo apt-get install python3-pyaudio
```

While the virtual environment *ml* from Assignment 1 is still active execute the following pip installs:

```
pip install pyaudio
pip install soundfile
```

Now execute:

```
python3 fetch.py
```

Report the performance. What do you notice?

NB If you have to execute *fetch.py* a second time, be sure to first delete the generated *.jpg* images in the *<testrecs>* directory. You also may want to use your own recordings of pronounced digits.

- b) Study the script that converts the audio file to an image (i.e., the function *graph_spectrogram* in the file *fetch.py*). Propose and describe your own method to convert the audio file into an image that you expect to be effective in audio classification tasks using CNNs. Use pseudo-code to define your method and motivate your proposal.

NB Write your findings in a pdf-file. Submit your answers in a single *.zip* file using Brightspace before Monday December 19th 2022 23.59h.

References and Links:

- [1] The Jupyter Notebooks are slightly adapted from Chapter 5 from the book by Francois Chollet. Deep Learning with Python, Manning Publication, December 2017. (Highly recommended book. See also 2nd Edition.)
- [2] The Cats and Dogs small data set is extracted from the original data set: <https://www.kaggle.com/c/dogs-vs-cats/data> , please adhere to the Kaggle-rules, i.e., use the provided data for this workshop only, and do not distribute any further.
- [3] Assignment 2 uses adapted code from [Rishi Sidhu](#), (22-3 2019) and processed data from [Zohar Jackson](#) (2019).