

Homework (Voice Anti-spoofing)

Task

Implement and train a Countermeasure (CM) system on the Logical Access partition of the [ASVSpooof 2019 Dataset \(Kaggle Link\)](#). You may find the [ASVspooof 2019 evaluation plan](#) useful.

Countermeasure systems

This time, we restrict our solution to the LCNN architecture.

You cannot use implementations available on the internet in any way (including looking at them).

LightCNN

Implement [LightCNN \(LCCN\)](#) following the Speech Technology Center [paper](#).

Hints:

1. Take training recipe and data preparation scheme from [this paper](#). Also, read the comparative study and think whether you should use A-Softmax or Cross-Entropy loss function.
2. Use STFT (FFT in the paper) as front-end. (Though others may work too.)
3. Put dropout layer as it is done in [this paper](#).

Code and logs

You have two options for code:

1. Write your code in a clone of [PyTorch Project Template](#): \$grade_coef=1.0\$
2. Write your code in a [ipynb](#) notebook: \$grade_coef=0.8\$

[!IMPORTANT] We do not accept option 2 if it looks like a mess. Use markdown to split your notebook in meaningful sections. Add some comments.

Besides, you need to prove that you indeed trained a model by showing logs via [W&B](#). Add all the required plots: losses on train/eval; metrics on eval, etc.

Report

HSE requires you to write a short report at the end of the practice. Ask your Program advisors about the specific rules for your program. In general, you should include the following information.

- Introduction: What the mini-course was about and what task are you solving at the end (Deepfake Detection).
- Description of the task: Introduce metrics and main terminology. Describe LCNN model.
- Methodology: Describe how have you trained the model. What learning rate, batch size, what input features, how many steps, etc.
- Results: Show results and some plots.

- Conclusion: Describe if the model is working and what have you learnt during the mini-course.

Do not mention LauzHack. The report should be mostly about the course content (general-wise) and the task for the homework.

Grade

Your mark (\$M\$) will depend on your model performance (\$P\$) the \$grade_coef\$ (based on the code style):

$$M = grade_coef * P$$

Wandb logs are not graded but their absence or if they look suspicious will result in zero grade (your EER plot should slowly go closer and closer to your final EER). Report is not graded but required by HSE to pass the practice.

Performance metrics (must be calculated on the evaluation set of the LA partition of the ASVspoof 2019 Dataset):

- Equal Error Rate (EER).

Code for metrics is provided in this repo (see `compute_eer` function). You should provide metrics in your report and achieve the following performance:

- EER range:
 - $9.5 < EER$: $P=0$.
 - $5.3 \leq EER < 9.5$: $P=4$ (linearly scales from 4 at $EER=9.5$, to 10 at 5.3)
 - $EER < 5.3$: $P=10$

Note: requested EERs are much higher than the ones described in the paper to save your time.

Time to achieve full \$P\$ grade in Kaggle for the teacher's solution (may vary a bit if the random seed is bad):

Model	Time (h)
LCNN	4.5-5

Submission

To submit your solution, use the following [Google Form](#).

- Full Name: In Russian, full official name as in HSE.
- HSE email address: email address.
- TG: your telegram.
- Official: say if you are an official student.
- Submission URL:
 - If your solution is an IPython notebook, submit a link to Google Colab.
 - If your solution is a PyTorch project, upload it on GitHub and submit the link to GitHub repo.
- Predictions `csv`: save your model predictions for the `eval` set into `hse_email_username.csv` file and upload it to the form. **You want receive the grade if the filename is wrong.**

Important: Check your solution by running `grading.py` script, described below. It also explains the expected formatting of `hse_email_username.csv`

`grading.py` expects to have a directory `students_solutions` with the following format:

```
students_solutions
├─ iiiivanov.csv # example hse username
├─ ...
└─ pppetrov.csv # example hse username
```

To run the script, do:

```
# protocol is located next to grading.py
mkdir students_solutions
mv MyName_MySurname.csv students_solutions/
python3 grading.py
```

We will use the output of this script to grade your solutions.