



Hackathon #2: Training and Evaluation

Maike Theis

Summer School for Young Researchers in Radiology & AI 16. - 20. September 2024





Tutors



M.Sc. Priya
Priya.Priya@iais.fraunhofer.de
Fraunhofer-Institut für Intelligente Analyse- und
Informationssysteme IAIS, Universität Bonn



M.Sc. Maurice Günder

Maurice.Guender@iais.fraunhofer.de

Fraunhofer-Institut für Intelligente Analyse- und
Informationssysteme IAIS, Universität Bonn



M.Sc. Maike Theis

Maike.Theis@ukbonn.de

Klinik für Diagnostische und Interventionelle

Radiologie, Universitätsklinikum Bonn



M.Sc. Christoph Fürböck
christoph.fuerboeck@meduniwien.ac.at
Universitätsklinik für Radiologie und
Nuklearmedizin, Medizinische Universität Wien



M.Sc. Konstantin Miloserdov konstantin.miloserdov@meduniwien.ac.at Universitätsklinik für Radiologie und Nuklearmedizin, Medizinische Universität Wien









UNIVERSITÄT BONN

Learning Outcome



• Main task: Train and evaluate a classification model for detection of pneumonia

- In the process:
 - Define training and model parameters
 - Implement a training loop
 - Evaluate and visualize model performance







Tasks



- Use the following notebook as a basis: train_pneumonia_baseline.ipynb
- Load dataset:
- Load the PneumoniaMNIST dataset from MedMnist (https://medmnist.com/)
- Use the code from Hackathon #1
- 2. Implement classification model:
 - Define training and model parameters (learning rate, optimizer, epochs, ...)
 - Define classification model (see e.g. https://docs.monai.io/en/stable/networks.html#nets)
 - Vary the training and model parameters and document your observations regarding model performance, training time, etc.







Tasks



- 3. Define performance metrics
 - Which metrics are suitable for evaluating your classification model? (see e.g. https://scikit-learn.org/stable/modules/model_evaluation.html)
- What exactly do the individual metrics mean? How can you interpret the results?
- 4. Training loop
 - Iteratively pass batches through your network, calculate loss and update model weights
 - Evaluate model performance on your validation data
 - Save your model
- 5. Final model evaluation
 - Visualize model performance (loss, evaluation metrics)
 - Model inference on the test set







Optional Tasks



Task 1: Multi-label classification

- Use the ChestMNIST dataset from MedMNIST
- Transfer the code to solve a multi-label problem
- How can you handle class imbalances?

■ Task 2: Large Language Models (LLMs)

- Use LLMs to generate radiological reports for the PneumoniaMNIST dataset
- See https://huggingface.co/microsoft/Phi-3-vision-128k-instruct

Task 3: Generative data augmentation

- Train a Generative Adversarial Network (GAN) to enlarge your dataset with new generated healthy and pneumonia images
- Consider to use a pre-trained GAN (see https://github.com/Project-MONAI/tutorials/blob/main/modules/mednist GAN tutorial.ipynb)



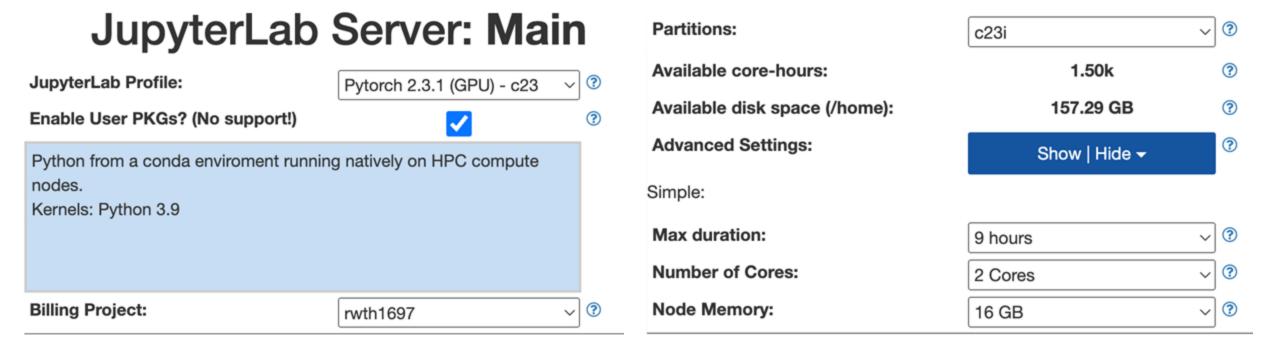




Getting started



- 1. Log into Jupyter Hub: https://jupyterhub.hpc.itc.rwth-aachen.de:9651/
- 2. Start a Jupyter server with the following settings









Getting started



1. open the terminal

2. access the folder:

cd SummerSchoolUKB

3. Pull new scripts from git repository:

git stash
git pull
git stash pop

→ Use train_pneumonia_baseline.ipynb as a basis









Happy coding and feel free to ask questions!



In case you need WiFi:

name: FhG-Gast

password: fhg-0512





