Assignment 2 - XAI

Data Science course

End goal for assignment 2

Deadline: **April 17**.

Task: Predict the diagnosis for given x-ray images.

1. Information on the task: https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia/
2. Information on the data: https://www.kaggle.com/datasets/paultimothymooney/chest-xray-pneumonia/data

End product: a report containing:

1. The task definition
2. Data description: results of the data exploration
3. Description of the baseline method
4. Description of the hyperparameter optimization
5. Results: a table with (a) Baseline results (replication of existing method); (b) Results of the hyperparameter optimization / alternative approach, and (c) an extensive analysis of the best model using XAI techniques.
6. A brief conclusion: which feature representation / model worked the best and why

This is a group assignment. I advise you to write your report in Overleaf.

Tasks

Dataset Exploration

- Download the dataset from the Kaggle website.
- Explore the dataset to understand the input features (images), classes (normal vs. pneumonia), and distribution of the data.
- Visualize sample images for both classes, along with their labels.
- You can use https://www.kaggle.com/code/madz2000/pneumonia-detection-using-cnn-92-6-accuracy as a reference (guide) for this part. Do not simply copy and run, but try to learn what is happening and why it is happening.

CNN Baseline

- Use the example CNN architecture provided in the Kaggle code as a base. Understand the role of key components like convolutional layers, pooling layers, and fully connected layers.
- Train the model on the training dataset.
- Track and report model performance metrics (accuracy, loss) during training and on the test dataset.

Performance Evaluation

- After training, evaluate the model on the test set.
- Present the confusion matrix, precision, recall, and F1 score for the baseline model.

Hyper-Parameter Optimization / alternative models

- Use an HPO method (SMAC3, Bayesian Optimization etc.) to try to improve the current model. Use a realistic budget (do not run experiments for more than a day).
- Try a different model (machine learning technique) for the given task, provide a motivation why you picked a particular model.
- Present the confusion matrix, precision, recall, and F1 score on train and test set for the final optimized CNN model and the alternative ML model.

Explainable AI

- Inspect misclassification errors and use explainable AI techniques to highlight what the models are focusing on.
- Feature Importance with Grad-CAM: (see for example https://keras.io/examples/vision/grad_cam/ or https://github.com/ismailuddin/gradcam-tensorflow-2 or https://deel-ai.github.io/xplique/v1.4.0/)
  - Use Grad-CAM (Gradient-weighted Class Activation Mapping) to visualize which parts of the input images contribute most to the model's predictions
  - Choose 5-10 misclassified samples and use Grad-CAM to inspect which regions the model is focusing on.
  - Interpret the results: Are the activations reasonable? Can you identify why the model made incorrect predictions for the chosen samples?
- Layer-wise Relevance Propagation (LRP) (optional for bonus points):
  - Apply LRP to further analyze how different layers of the network contribute to the final predictions.
  - Compare the insights from LRP with those from Grad-CAM. Are there any differences in how each method highlights the important regions of the images?

2. Report writing

Your report needs to contain the following structure:

1. Task definition
2. Data exploration
3. Baseline method
4. Hyperparameter optimization
5. In-depth explainable AI analysis
6. Results (as tables):
   a) Baseline results (replication of existing method);
   b) Results of the hyperparameter optimization;
   c) the results of XAI analysis.
7. conclusion

Keep it concise; maximum 8 pages. Don't add information that is not asked for (such as your struggles with the process, or how you downloaded and processed the data). Writing concise reports is an important exercise.

This is a team assignment. I advise you to write your report in Overleaf.

Please submit your report **as a single pdf (not zipped)**, together with your code zipfile, in Brightspace.

You can add additional images in the appendix of your pdf (the appendix does not count towards the page limit).

Tips and Tricks (setting up the environment)

To use the notebook from Kaggle (https://www.kaggle.com/code/madz2000/pneumonia-detection-using-cnn-92-6-accuracy/), you can

1. Run it directly within the Kaggle by clicking the “Copy & Edit”
    a. Disadvantage: this may be super slow or even halted halfway since it subjected to the computation resources allocated to you by the Kaggle platform.

    b. Advantage: you don’t need to care about the environment
2. Run it in Google Colab/notebook by clicking the three dot-menu button on top of the webpage and select “Open in Google Notebooks” (may require paid subscriptions to Google cloud) or “Open in Colab”
    a. It is even slower than Kaggle, but you don’t need to worry about the project being cancelled.

    b. you don’t need to care about the environment
3. Run it locally and install the environment via anaconda (Graphical UI):
    a. Create a python virtual environment usin
        1. **conda create -n GIVE_IT_A_NAME python==3.10,** for example **GIVE_IT_A_NAME is test**
        2. After it creates the environment, run: **conda activate test**
        3. Do: **pip install -U pip**

    b. **Install jupyter notebook or Jupyter Lab to your environment via Anaconda (optional).**

    c. **If you are on a Windows, Linux, or a MacBook with Intel chipsets user:**
        1. Download the “requirements_kaggle_x64.txt” or “requirements_windows.txt”
        2. If you prefer to have the same environment with Kaggle notebook, use the **first one**.
        3. If you prefer to have the (almost) latest supported one, use the **second one**.
        4. Open anaconda, install the package via anaconda terminal (ask ChatGPT, Claude, or Gemini if you don’t know how to do it):
        5. pip install -r requirements_kaggle_x64.txt –force
        6. Or: pip install -r requirements_ windows.txt –force
        7. Done

d. **If you are a mac user with a MacBook with Apple Silicon chipsets (click the apple logo on top left of your screen, click “About This Mac”, if the “chip” row says it is “Apple M…”)**

   1. Download the “requirements_macos.txt”
   2. Open anaconda, install the package via anaconda terminal (ask ChatGPT, Claude, or Gemini if you don’t know how to do it):
   3. pip install -r requirements_macos.txt –force
   4. Done

Tips for Assignment 2:

1. Hyperparameter tuning: if you are limited in computation resources, you can perturb just one hyperparameter. But please be clear **which and how you are tuning.**
2. The XAI part is important and interesting.
3. Do this assignment **as soon as possible** since the experiments **take longer** than you might expect.
4. Please come to the Lab sessions if you have questions or need help.