Assignment 6

MLP Group

Task Description

torchserve is a flexible and easy-to-use tool for serving pytorch models in production environments. In this assignment, you'll deploy a pre-trained Mask R-CNN object detection model using *torchserve* on an edge device (Jetson Nano or Raspberry Pi) and use it to detect objects in images.

mar format

A .mar file is a specialized archive format used by *torchserve* that bundles all components needed to serve a model. It contains:

- The model weights and architecture
- Handler scripts that define how to:
  - Pre-process input data
  - Run inference
  - Post-process model outputs
- Model metadata and dependencies
- Model version information
- Custom files needed for inference

The .mar format ensures portability and reproducibility when deploying pytorch models. Think of it as a self-contained package that includes everything *torchserve* needs to serve your model.

Task steps

1. Environment setup (Same as last assignment):
   1. Download Anaconda (What is Anaconda?)on your computer - link to ARM version with Python 3.12
   2. Copy the anaconda installer to the Jetson nano/Raspberry PI via scp

    3. Install Anaconda. It is highly recommended to install via terminal by just executing the downloaded file.
        - During installation you can decide if you want to have anaconda at startup - I would recommend to don't do that to avoid system breaks
    4. Download pytorch installer from your computer at this url: http://download.pytorch.org/whl/cpu/torch/. You have to download the version 2.2.2 for the python version installed in your device and for the right architecture (arm64). Then transfer it to the Jetson nano/Raspberry PI
    5. Download torchvision installer from your computer at this url: http://download.pytorch.org/whl/cpu/torch/. You have to download the version 0.17 for the python version installed in your device and for the right architecture (arm64). Then transfer it to the Jetson nano/Raspberry Pi
2. Download *torchserve* package version 0.12 from pypi and move to the Jetson Nano/Raspberry Pi. Then install it with pip
3. Download captum package version 0.7.0 from pypi and move to the Jetson Nano/Raspberry Pi. Then install it with pip
4. Download the Java 17 conda package from here, move to the edge device and install it with the command `conda install`
5. Download the .mar file from studon (rename it by removing .sec extention) or in alternative, build the .mar file which is an archive containing all the information to predict with the object detector. Follow the instruction here to build the mar file. Note: in your computer you have to install `torchserve torch-model-archiver torch-workflow-archiver` to build the mar package
6. Move the mar package to the Jetson Nano/Raspberry pi
7. Use *torchserve* to serve the mar package (Check here for command example)
8. Use the image on studon, under Assignment 6 folder to make the api request
9. [Optional] Once you get the coordinates, apply bounding boxes to the image as here. One way to apply bounding boxes is through *matplotlib*