AI 510 Artificial Intelligence of Cloud Computing
HOS09A Introduction to Anaconda and PyTorch

07/02/2024 Reviewed by Anh Nguyen

07/05/2024 Reviewed by Naveena Moddu

School of Technology and Computing (STC) @City University of Seattle (CityU)

Before You Start

- The directory path shown in the screenshots may be different from yours.
- Some steps are not explained in the tutorial. If you are not sure what to do:
    - Consult the resources listed below.
    - If you cannot solve the problem after a few tries, please contact the student worker through the MS Teams course channel.

Learning Outcomes

- Students will be able to learn:
    - Introduction to PyTorch
    - Setup conda environment for VSCode Jupyter Notebook
    - Installation to PyTorch
    - Installation to ONNX and ONNXRUNTIME

Resource

- Noah G., Alfredo D. (2021). Practical MLOps. O'Reilly Media, Inc.
- Learn the basics. (n.d.). PyTorch: https://pytorch.org/tutorials/beginner/basics/intro.html
- Installing Miniconda. (n.d.). Miniconda: https://docs.anaconda.com/free/miniconda/miniconda-install/
- Zivkovic, S. (2021, November 8). #017 PyTorch - How to apply batch normalization in PyTorch. Data Hacker. https://datahacker.rs/017-pytorch-how-to-apply-batch-normalization-in-pytorch/

PyTorch

PyTorch is one of the most popular open-source Python-based scientific computing packages for deep learning frameworks. PyTorch was initially developed by the Facebook Artificial Intelligence research group. It provides two broad services for a replacement for NumPy to use the power of GPUs, and an automatic differentiation library to implement neural networks.

PyTorch has three levels of abstraction:

- Tensor – n-dimensional array run on GPU
- Variable – Store data and Gradient
- Module – Neural Network layer that stores weights

Anaconda

Similar to the Python virtualenv library, Anaconda allows the creation of virtual environments and simplifies package management and deployment. Although the Conda project is heavier than the virtualenv, one of the good things about Conda is that you can switch between various Python versions.

Install miniconda

Anaconda setup can be as complicated as setting up a server with the necessary network configuration. However, the Anaconda also has a minimal installer called Miniconda for local development. We will set up and use Miniconda for this module.

Step 1) Visit https://docs.anaconda.com/free/miniconda/#latest-miniconda-installer-links to download the installer for your machine platform. The MacOS users should consider which CPU architecture to use before downloading the installer (ex. Intel or Arm-based).

Step 2) Follow the guide from the Miniconda page to set up Miniconda on your machine. https://docs.anaconda.com/free/miniconda/miniconda-install/

Step 3) After installing Miniconda, check if the "conda" command works in the newly opened terminal.

```
≥ conda
usage: conda [-h] [-v] [--no-plugins] [-V] COMMAND ...

conda is a tool for managing and deploying applications, environments and packages.

options:
  -h, --help          Show this help message and exit.
  -v, --verbose       Can be used multiple times. Once for detailed output, twice for
  --no-plugins        Disable all plugins that are not built into conda.
  -V, --version       Show the conda version number and exit.
```
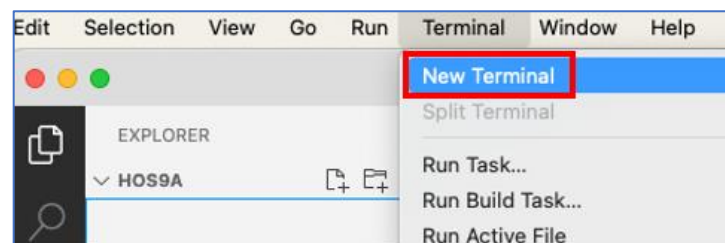
Setup Miniconda environment in VSCode

We will continue to create a conda environment and have it configured to VSCode.

Step 1) Create a folder and open the folder in VSCode for the HOS09A task and open a terminal.

```
Edit   Selection   View   Go   Run   Terminal   Window   Help

   ● ● ●                          New Terminal
                                  Split Terminal
        EXPLORER
        ∨ HOS9A                   Run Task...
                                  Run Build Task...
                                  Run Active File
```

Step 2) Use the following commands to create and activate a conda environment along with installing the necessary Python packages to the environment.

```
# 1. Creates virtual environment named ai510
# Type y and press the enter key whenever the prompt asks you
conda create --name ai510

# 2. Check the created conda environment
conda info --envs
```

```
# 3. Activate created environment
conda activate ai510

# 4. Install Python and Jupyter packages. You can also install other Python
versions.
# Type y and press the enter key whenever the prompt asks you
conda install python=3.11 jupyter
```

Note:

Here are some important command flows that are often used

- **Deactivate default base** environment start when terminal/command application starts

```
# If you want to disable conda being automatically activated on the
terminal open
conda config --set auto_activate_base false
```

- **Creating** environment (we used this to set our environment)

```
# 1. Creates virtual environment named ai510
conda create --name ai510

# 2. Check the created conda environment
conda info --envs

# 3. Activate created environment
conda activate ai510

# 4. Install Python and Jupyter packages. You can also install other
Python versions.
conda install python=3.11 jupyter
```

- **Daily** work

```
# Activates previously created conda env
conda activate ai510

# After some amount of work …
# you can deactivate conda env to create new or delete env afterward
conda deactivate
```

- **Delete** the environment to **refresh** (Execute this if you want to clean up the Conda environment **after** this module.)

```
# need to deactivate environment before deleting env
conda deactivate

# remove(delete) env along with all of the packages in the
environment folder
conda remove --name ai510 --all
```

```
# check if the environment is correctly deleted by not seeing the env
name in the env list
conda info --envs
```

Step 3) Click the top VSCode search bar, and type "python: Select Interpreter" to click into the Interpreter selection list. When the list shows, select just configured conda env python installation. The VSCode finds the Python executables when we set up conda environments.

a.

b.

c.
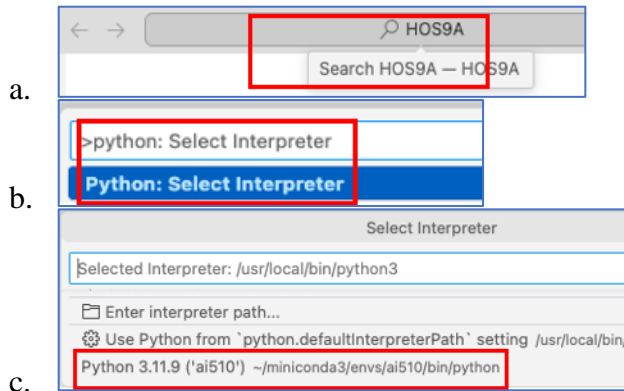
Setup PyTorch and run inference on the ONNX model

We will continue to explore how the PyTorch library can be configured and use the existing model to create the ONNX model as well as run inference on it.

Step 1) Open Command Palette (Ctrl + Shift + P) > Python Create terminal to open a new terminal.
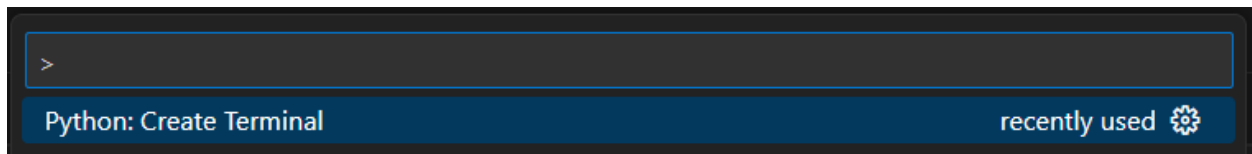
Activate the "ai510" environment if you deactivate the environment, and run the following commands to install the necessary packages.

```
conda activate ai510

pip3 install onnx onnxruntime pillow torchvision
```
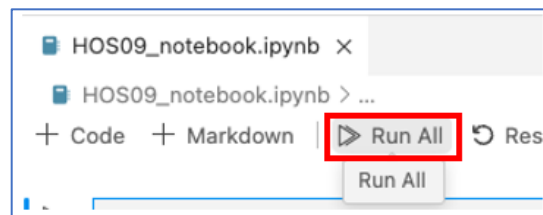
Step 2) Copy & Paste the provided files to the VSCode opened HOS09A folder
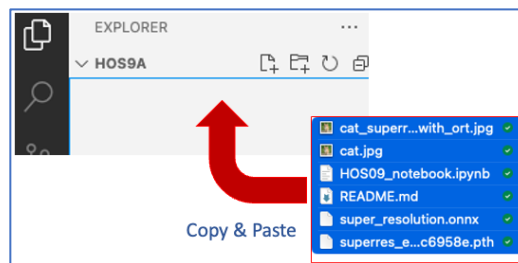
Step 3) Open "HOS09_notebook.ipynb" file and on the top right of the notebook, click "Select Kernel" to select the previously created conda environment.

a.

b.
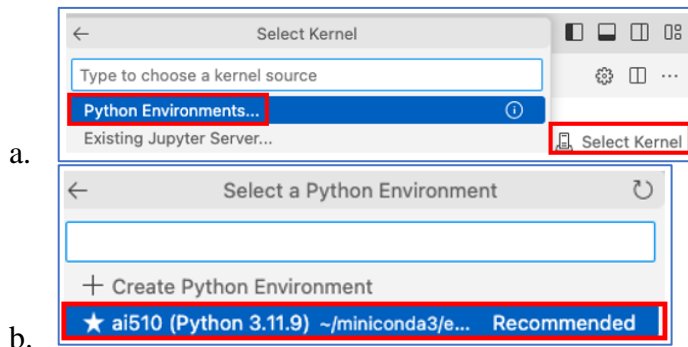
Step 4) Click the "Run All" button at the top center of Jupyter Notebook to execute all code cells.

Step 5) Save the executed repository and submit the entire HOS09A folder for this HOS.

HOS submission instructions:
1. Please install the GitHub Desktop: https://cityuseattle.github.io/docs/git/github_desktop/
2. Clone, organize, and submit your work through GitHub Desktop:
https://cityuseattle.github.io/docs/hoporhos