This guideline shows how to use/install Miniconda and PyTorch on rwcpu8.cse.ust.hk

Using Global Miniconda and PyTorch

If you don't want to install Miniconda and PyTorch yourself, you can use the global Miniconda and PyTorch installed at `/export/data/miniconda3`.

1. Initialize Miniconda:

   ```
   source "/export/data/miniconda3/etc/profile.d/conda.csh"
   ```

2. If you want to use PyTorch, activate the `pytorch` `conda` environment:

   ```
   conda activate pytorch
   ```

3. There is also a `conda` environment for TensorFlow 2:

   ```
   conda activate tf2
   ```

4. After you activate the corresponding environment, you should be able to run Python scripts that uses PyTorch/TensorFlow by the `python` command:

   ```
   python python_script.py
   ```

Installing Your Own Miniconda

1. Download Miniconda installer.

   ```
   wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
   ```

2. Run the installer. The argument `-p` specifies the path to install Miniconda. You may not be able to install Miniconda to your home directory because there is a space limit for your home directory. Choose another directory that you can access and that does not have a space limit, such as `/rwproject/kdd-db/your_username`.

   ```
   sh Miniconda3-latest-Linux-x86_64.sh -b \
     -p /rwproject/kdd-db/`whoami`/miniconda3
   ```

   Since `/rwproject/kdd-db/` is a remote folder, it may take several minutes for the installation to finish.

3. Add the code that initializes Miniconda to your shell initialization script. Suppose you use the default shell `tcsh`:

   ```
   /rwproject/kdd-db/`whoami`/miniconda3/bin/conda init tcsh
   ```

   The code will be written to `~/.tcshrc`. But the default shell initialization script set by cssystem is `~/.cshrc_user`, so you should write the content in `~/.tcshrc` to `~/.cshrc_user`:

```
cat ~/.tcshrc >> ~/.cshrc_user
```

Since if `~/.tchsrc` exists, `~/.cshrc_user` won't be loaded, so you need to remove `~/.tcshrc`:

```
rm ~/.tcshrc
```

4. Log out and log in again. If Miniconda is successfully installed, you should be able to see the usage of `conda` using the following command:

```
conda --help
```

Installing Your Own PyTorch

You can install PyTorch to the default environment (i.e., the `base` environment) or a new environment. If you want to install PyTorch to the default environment, skip Steps 1.

1. Create a new `conda` environment.

```
conda create -n pytorch
```

`pytorch` is the name of the environment to be created. You may specify a different name.

2. Activate the environment that you want to install PyTorch to.

```
conda activate pytorch
```

Replace `pytorch` with `base` if you use the default environment.
You should see a prefix in your prompt showing the name of the activated environment, e.g.:

```
(pytorch) rwcpu8.cse.ust.hk:your_username:101>
```

3. Install PyTorch

```
conda install pytorch torchvision cudatoolkit=10.2 -c pytorch
```

It may be very slow to download the `pytorch` package, but that's not because you're installing PyTorch to a remote folder. It is a known problem that sometimes the download speed of pytorch is slow.

4. If PyTorch is successfully installed, then you could see the version of PyTorch by the following command:

```
python -c 'import torch; print(torch.__version__)'
```

5. Verify PyTorch is able to use GPUs.

```
python -c 'import torch; print(torch.cuda.is_available())'
```

The output should be `True` if PyTorch is able to use GPUs.

Useful Links

- Miniconda Documentation
- PyTorch: Getting Started
- Install TensorFlow