Install Jupyter Notebook

https://docs.anaconda.com/anaconda/install/windows/

Start Anaconda Navigator, click the Environments panel at the left and click the env you want to work with (you could create a new one here).

Go back to Home, and look for Jupyter Notebook, you will see either install or launch. Once you install it and you can launch Jupyter Notebook with this environment.

If you want to create an environment or install a package to the environment, do the following:

Open a windows Anaconda Prompt,

To create env: conda create myenv (you may also use Navigator to create a new environment)

To access env: conda activate myenv,

Now you can install packages to this evn: pip install

Install LLama3.1 locally on your computer

Windows:

1.Head to Ollama's download pageLinks to an external site. to download the Ollama installation file. Please refer the download pageLinks to an external site. for OS specific install instructions. You can confirm the Ollama server status by hitting the local URL http://localhost:11434/Links to an external site.,

2. Open a windows terminal (command-prompt) and execute the following Ollama command: ollama run llama3.1. Now you can have interactive conversations with LLama3.1, To create and access env: conda create --name myenv conda activate myenv, pip install

3. You can call the llama31 in Python

```
from langchain_community.llms import Ollama

llm = Ollama(model="llama3.1")

prompt = "Tell me a joke about llama"

result = llm.invoke(prompt)

print(result)
```

4. You can requests from URL, which means the llama 3.1 is a server and everyone can call it.

```
import requests

import json


# Define the URL and the data payload

url = "http://localhost:11434/api/generate"

payload = {

    "model": "llama3.1",

    "prompt": "who is the president of USA",

    "stream": False

}


# Set the headers

headers = {

    'Content-Type': 'application/json'

}
```

5. Make the POST request

response = requests.post(url, headers=headers, data=json.dumps(payload))

Print the response

print(response.json())

Unix:

install ollama

module load ollama/0.1.38

export OLLAMA_HOST='10.139.120.41:11434'

ollama serve

curl http://10.139.121.74:11434

At serve:

Request a jupyter session with a GPU.

Open a terminal tab on jupyter lab.

Load the ollama-0.1.38 module.

Run ollama serve.

Open a second terminal and download the llama3.1 model by running ollama pull llama3.1 .

Open a jupyter notebook and use the ollama-0.1.38 kernel.

Run the python example code below

import ollama

response = ollama.chat(model='llama3.1', messages=[{ 'role': 'user', 'content': 'Why is the sky blue?', },]) print(response['message']['content'])

Access Openai API

To access the OpenAI interface, you typically need to use the OpenAI API from OpenAILinks to an external site.

Sign Up and Get API Key: First, sign up on the OpenAI platform and get an API key.

Install OpenAI Python Library: pip install openai

Code:

import openai

```
# Replace 'your-api-key-here' with your actual OpenAI API key
openai.api_key = 'your-api-key-here'

# Example usage of the GPT model
response = openai.Completion.create(
  model="text-davinci-003",  # You can replace this with another model like "gpt-3.5-turbo"
  prompt="Translate the following English text to French: 'Hello, how are you?'",
  max_tokens=60
)

# Print the response
```

ASU Supercomputing Instructions

https://asurc.atlassian.net/wiki/spaces/KESC/pages/1914667538/Managing+Python+Modules+Through+the+Mamba+Environment+ManagerLinks to an external site.

From shell, enter: interactive

module load mamba/latest

copy from github: git clone https://github.com/pyg-team/pytorch_geometric.gitLinks to an external site.

cd

pip install -e .

This will create an environment called <new_environment_name> that is copied from <environment_to_copy>.

mamba create -n yourENVname

source activate yourENVname

source deactivate

mamba info --envs

remove env: rm -r /home/hwang49/.conda/envs/xxx

remove env in Juputer: rm -r   /home/hwang49/.local/share/jupyter/kernels/xxx

mkjupy xxx

mamba install -c conda-forge pandas

mamba install -c conda-forge pytorch torchvision torchaudio

mamba install -c conda-forge qiskit=1.0.1

mamba install -c conda-forge qiskit-algorithms=0.3.0

mamba install -c conda-forge qiskit-machine-learning=0.7.2

mamba install -c conda-forge qiskit-aer

mamba install -c conda-forge -c pytorch -c nvidia pytorch-gpu torchvision torchaudio pytorch-cuda=12

mamba install -c conda-forge -c pytorch pytorch torchvision torchaudio