

# Run nemo-megatron-gpt-5B model with NVIDIA NeMo

## Introduction

NVIDIA NeMo is a powerful toolkit designed for researchers working on various conversational AI tasks, including automatic speech recognition (ASR), text-to-speech synthesis (TTS), large language models (LLMs), and natural language processing (NLP). It aims to facilitate the reuse of existing code and pretrained models while enabling the creation of new conversational AI models. In this tutorial, we will explore NeMo's capabilities and learn how to use the Megatron-GPT 5B language model for language modeling tasks.

## Model and Software References:

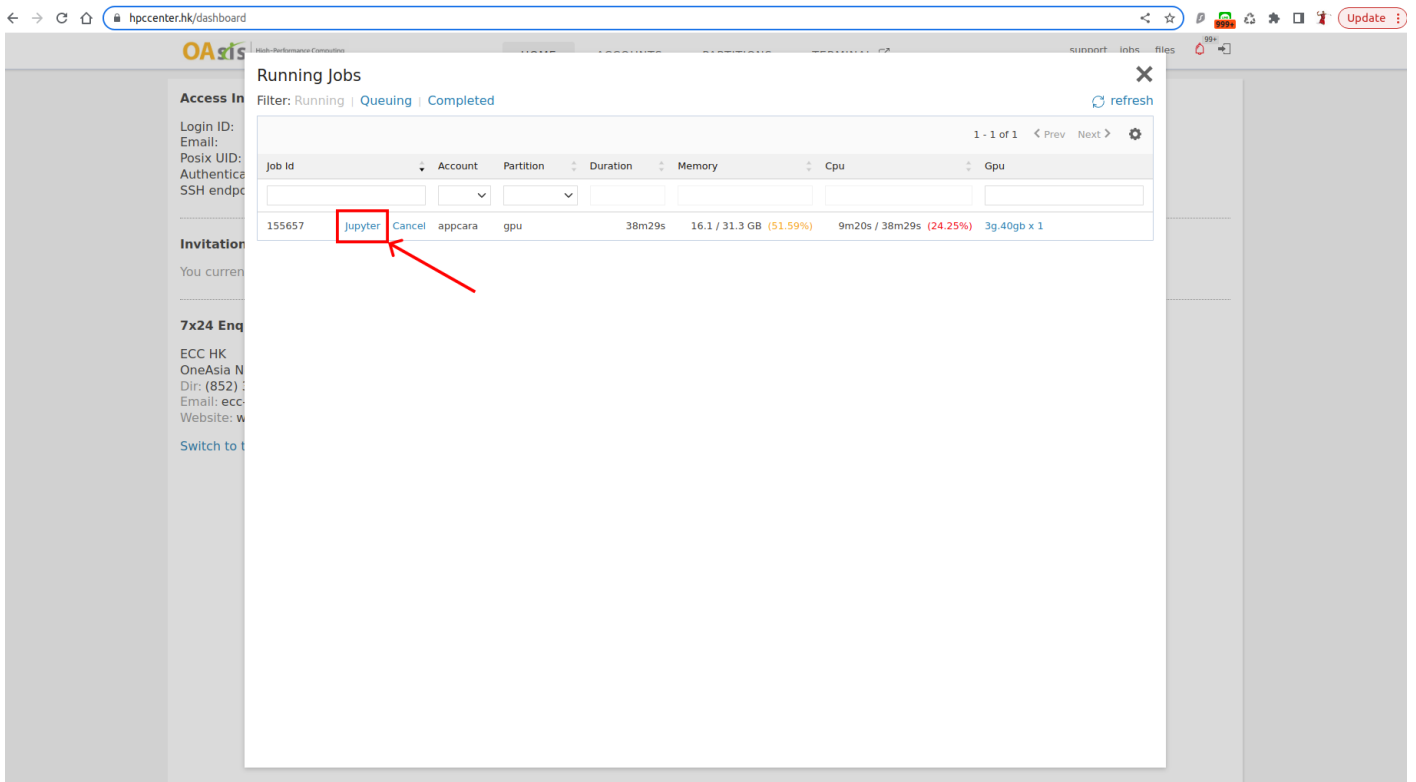
- NVIDIA NeMo: [<https://github.com/NVIDIA/NeMo>]
- nemo-megatron-gpt-5B: [<https://huggingface.co/nvidia/nemo-megatron-gpt-5B>]

## Launch Jupyter Lab Job

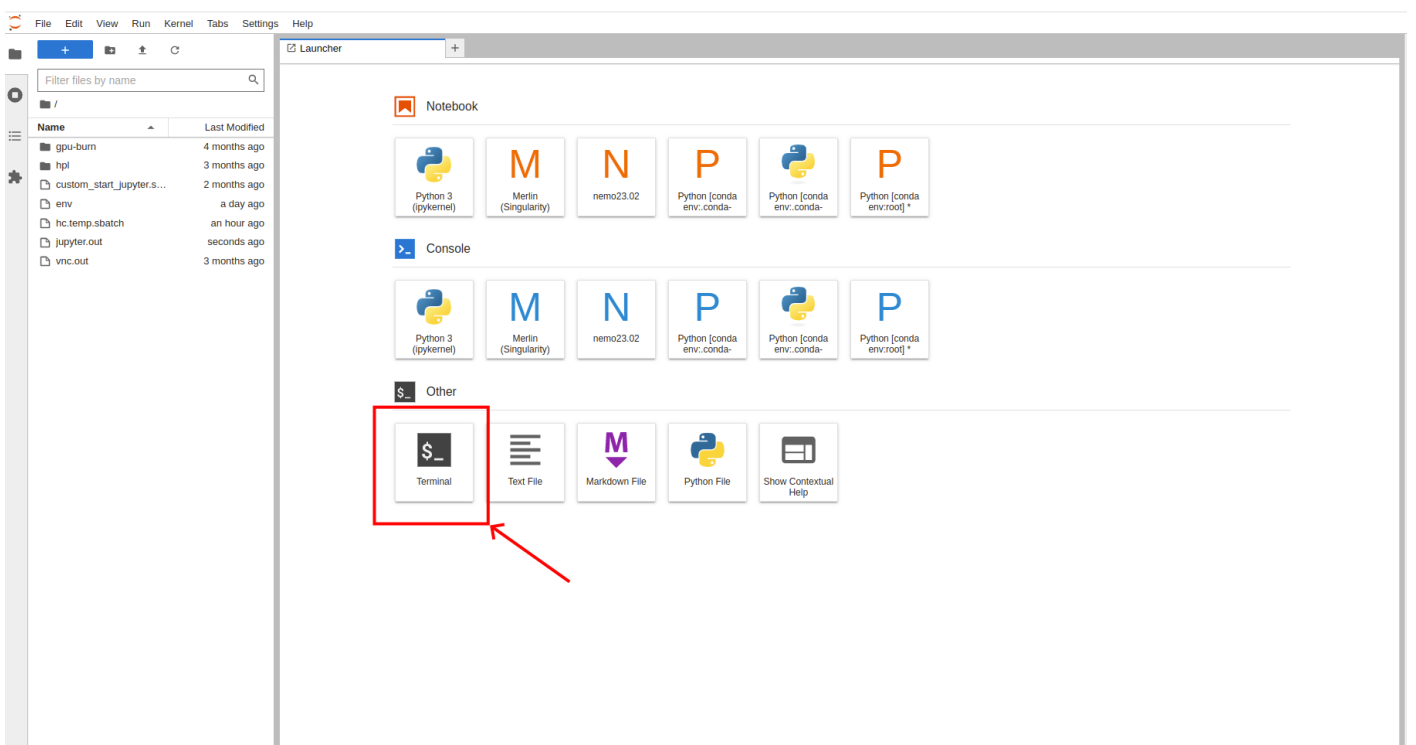
Create a Jupyter Lab job with the following specifications:

- CPU Cores: 4
- Memory: 64 GB
- GPU: 3g.40gb

Open your web browser and navigate to the Jupyter Lab web interface.



In the Jupyter Lab menu, open the Terminal.



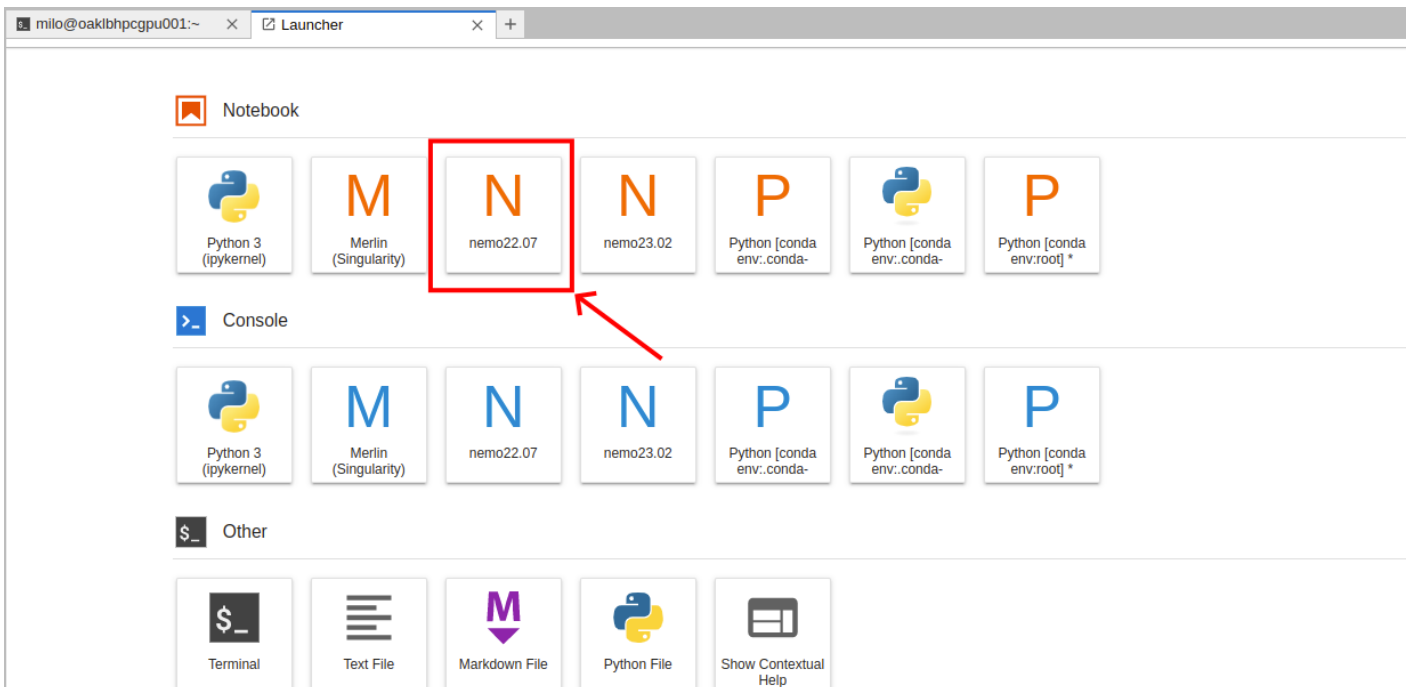
## Enabling the NeMo Container Kernel in Jupyter Lab

Execute the following commands in the Terminal:

```
cd $HOME
mkdir -p .local/share/jupyter/kernels/ngc.nemo.22.07
```

```
echo '{
  "language": "python",
  "argv": ["/usr/bin/singularity",
    "exec",
    "--nv",
    "-B",
    "/run/user:/run/user",
    "/pfss/containers/ngc.nemo.22.07.sif",
    "python",
    "-m",
    "ipykernel",
    "-f",
    "{connection_file}"
  ],
  "display_name": "nemo22.07"
}' > .local/share/jupyter/kernels/ngc.nemo.22.07/kernel.json
```

After adding the content to the kernel.json file, refresh your browser by pressing F5. You should now see "nemo22.07" under the Notebook section in Jupyter Lab Launcher.



# Launch eval server

Execute the following command in the Terminal:

```
# set the TMPDIR environment variable
export TMPDIR=/pfss/scratch02/appcara/nlp/tmp

# start eval server with nemo-megatron-gpt-5B model by nemo container
singularity run --nv /pfss/containers/ngc.nemo.22.07.sif python
/pfss/scratch02/appcara/nlp/NeMo/examples/nlp/language_modeling/megatron_gpt_eval.py
gpt_model_file=/pfss/scratch02/appcara/nlp/nemo_gpt5B_fp16_tp1.nemo server=true
tensor_model_parallel_size=1 trainer.devices=1 port=5556
```

## Send prompts to the model

Copying the Jupyter Lab File:

```
# copy the jupyter example file into your home folder
cp $SCRATCH_APPCARA/nlp/nemo-megatron-gpt-template.ipynb $HOME
```

In the "File Browser" section of Jupyter Lab, locate the copied file and open it. Also change the kernel to **nemo22.07**.

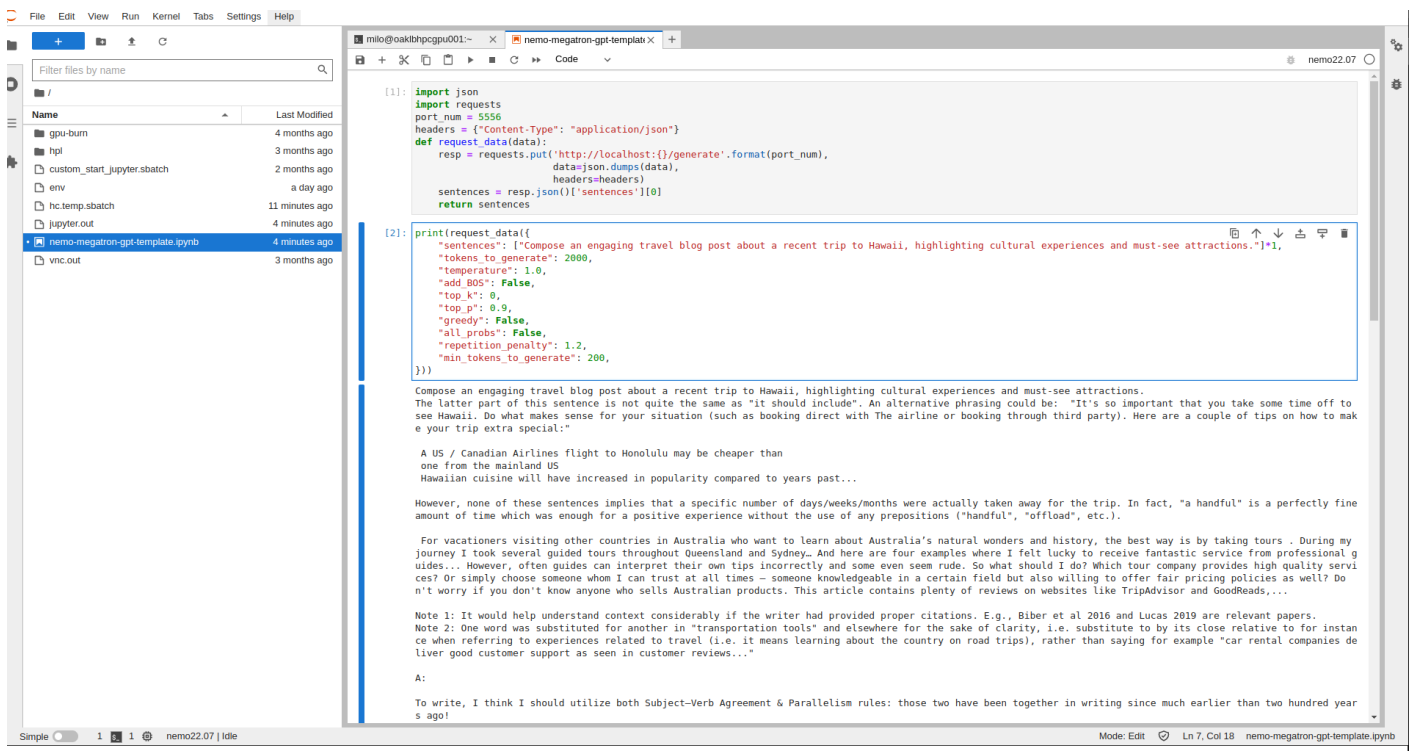
The screenshot shows the Jupyter Lab interface. On the left is the 'File Browser' panel with a list of files. The file 'nemo-megatron-gpt-template.ipynb' is selected. The main panel shows the code editor with Python code for sending prompts to the model. A 'Select Kernel' dialog box is open, showing a list of kernels. The kernel 'nemo22.07' is selected. The code in the editor is as follows:

```
[1]: import json
import requests
port_num = 5556
headers = {'Content-Type': 'application/json'}
def request_data(data):
    resp = requests.put('http://localhost:{}/generate'.format(port_num),
                        data=json.dumps(data),
                        headers=headers)
    sentences = resp.json()['sentences'][0]
    return sentences

[2]: print(request_data({
    "sentences": ["Compose an engaging travel blog post about a recent trip to Hawaii, highlighting cultural experiences and must-see attractions."],
    "tokens_to_generate": 2000,
    "temperature": 1.0,
    "add_BOS": False,
    "top_k": 0,
    "top_p": 0.9,
    "greedy": False,
    "all_probs": False,
    "repetition_penalty": 1.0,
    "min_tokens_to_generate": 10
}))
```

The code in the editor is for sending prompts to the model. The prompt is: "Compose an engaging travel blog post about a recent trip to Hawaii, highlighting cultural experiences and must-see attractions." The code sends this prompt to the model and returns the generated sentences.

Edit what you want to talk with chatbot in the "**sentences**" section of the file. Run the program.



### Revision #3

Created 9 June 2023 04:14:47 by Milo Cheung

Updated 9 June 2023 06:43:42 by Milo Cheung