**llama-3.1-8b-instruct**

Step 1: Launch an EC2 instance with the NVIDIA GPU-optimized AMI(X86) from the AMAZON marketplace and use g5.2xlarge instances type

Step 2: connect to SSH with Key pair and follow the commands.

$ nvidia-smi

$ docker login nvcr.io

Username: $oauthtoken

Password: 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

export NGC\_API\_KEY= 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 export LOCAL\_NIM\_CACHE=~/.cache/nim

mkdir -p "$LOCAL\_NIM\_CACHE"

docker run -it --rm \

--gpus all \

--shm-size=16GB \

-e NGC\_API\_KEY \

-v "$LOCAL\_NIM\_CACHE:/opt/nim/.cache" \

-u $(id -u) \

-p 8000:8000 \

nvcr.io/nim/meta/llama-3.1-8b-instruct:latest

**You can now make a local API call using this curl command:**

curl -X 'POST' \

'http://0.0.0.0:8000/v1/chat/completions' \

-H 'accept: application/json' \

-H 'Content-Type: application/json' \

-d '{

"model": "meta/llama-3.1-8b-instruct",

"messages": [{"role":"user", "content":"Write a limerick about the wonders of GPU computing."}],

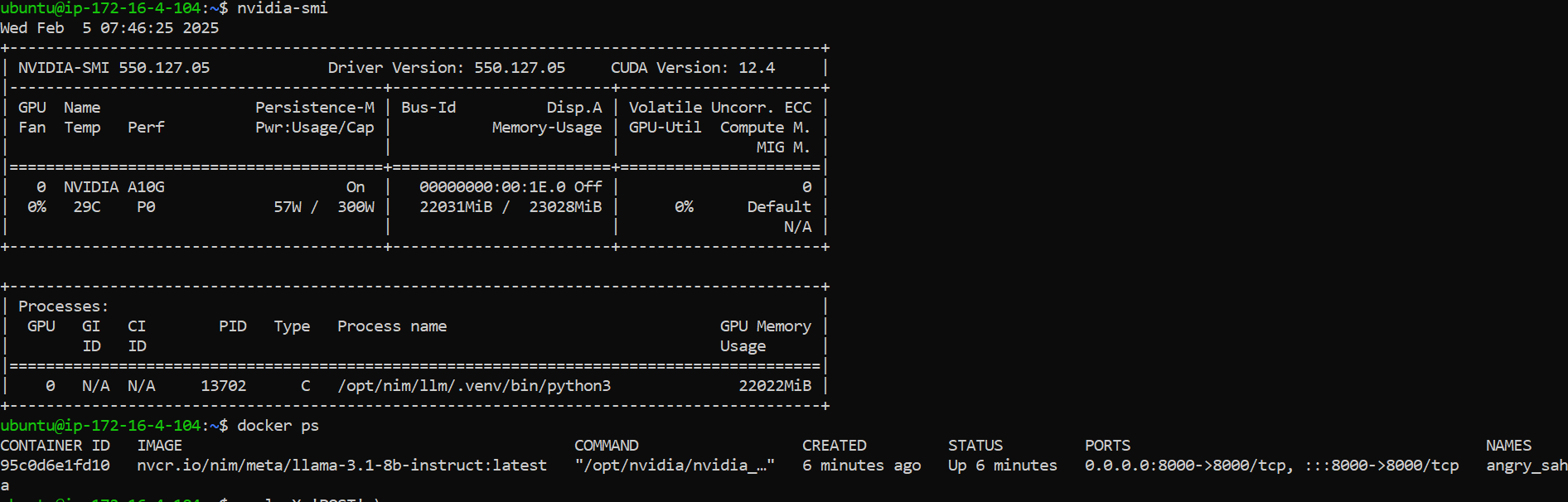
"max\_tokens": 64

}'

WEB URL: [llama-3.1-8b-instruct Model by Meta | NVIDIA NIM](https://build.nvidia.com/meta/llama-3_1-8b-instruct?snippet_tab=Docker)

[LLMA Model Deployment on EC2 with NVIDIA NIM](https://www.youtube.com/watch?v=WbFppAr2ItM)

[Function Calling — NVIDIA NIM for Large Language Models (LLMs)](https://docs.nvidia.com/nim/large-language-models/latest/function-calling.html)



A computer screen shot of text

Description automatically generated

A screenshot of a computer

Description automatically generated

**AI Virtual Assistant with NVIDIA Riva Contact**

Step 1: Launch an EC2 instance with the NVIDIA GPU-optimized AMI(X86) and use g5.xlarge instances type and then connect to SSH with Key pair

Step 2: Pull the Riva container from the NGC catalog

ngc config set

**KEY**: b3BvYzVhcjk2bWI0Mjl0MHFqb2lwMGozbmM6YTQ3OWQ1OTctN2FhZC00ZjRlLWI3NDMtZGVmYjc0NmExNWNk

LINK TO GENERATE KEY : [org.ngc.nvidia.com/setup/api-key](https://org.ngc.nvidia.com/setup/api-key)

Enter your NGC API Key from earlier, make sure that the CLI output is ASCIIor JSON, and follow the instructions using the Choicessection of the command line.

ngc registry resource download-version "nvidia/riva/riva\_quickstart:2.18.0"

cd riva\_quickstart\_v2.18.0

bash riva\_init.sh

bash riva\_start.sh

**NOTE: Before running step 3 come out of the riva\_quickstart\_v2.18.0 directory**

Cd ..

Step 3: Run the Riva ASR and TTS Hello Worldexamples

1. Clone Riva Sample Apps repository - <https://github.com/nvidia-riva/samples>

git clone https://github.com/nvidia-riva/sample-apps.git

1. Enter Riva and Rasa Virtual Assistant directory:

cd sample-apps/riva-contact

1. Install the package

npm install

1. set the URL of your running Riva AI Services in the config file [env.txt](https://docs.nvidia.com/deeplearning/riva/user-guide/docs/_downloads/0e45a2398c07f8bd0f4f43a7dc6d3e73/env.txt),

Step 4: Running the Service

npm run start

Using the Service

https://local ip address:8009/. ([Riva Contact](https://172.16.4.113:8009/))

WEB URL:

[Building a Speech-Enabled AI Virtual Assistant with NVIDIA Riva on Amazon EC2 | NVIDIA Technical Blog](https://developer.nvidia.com/blog/building-a-speech-enabled-ai-virtual-assistant-with-riva-on-amazon-ec2/)

[Riva Contact — NVIDIA Riva](https://docs.nvidia.com/deeplearning/riva/user-guide/docs/samples/sample-apps/riva-contact/README.html)



