GroqFlow™ Installation Guide

Notes

```
python3.8 -m pip install --upgrade pip
#python3.8 -m pip install -r examples-requirements.txt
export PYTHONPATH="/opt/groq/runtime/site-packages:$PYTHONPATH"
# Replace /dev/groq[0-7.pci with /dev/groqA[0-7].pci if needed
python3.8 yourcode.py
```

Introduction

The following describes how to install **GroqFlow**. These instructions enable users to build models for **Groq** hardware, as well as execute those builds in systems that have **GroqCard™** accelerators physically installed.

Connect to Groq

From your development machine, connect to a Groq system, i.e., groq1, groq2, groq3, and groq4, using your CELS GCE account information.

```
ssh CELS-GCE-UserID@groq1.cels.anl.gov
CELS-GCE-UserID@groq1.cels.anl.gov's password:
```

Prerequisites

- Download and install the **GroqWare™** Suite version 0.9.0.

--For more information, see the **GroqWare Quick Start Guide** at support.groq.com.

--To compile your model for **Groq** hardware, **GroqFlow** requires the **Groq Developer Tools Package** (groq-devtools). To run your compiled model on hardware, **GroqFlow** requires the **Groq Runtime Package** (groq-runtime).

-Clone the **GroqFlow** GitHub repo using the following command:

```
cd ~
git clone https://github.com/groq/groqflow.git
```

Step 1: Create and Activate a Miniconda Virtual Environment

The following example demonstrates downloading, installing, and creating a Miniconda virtual environment.

```
cd ~
wget https://repo.anaconda.com/miniconda/Miniconda3-latest-Linux-x86_64.sh
bash Miniconda3-latest-Linux-x86_64.sh
source ~/.bashrc
conda create -n groqflow_env python=3.8.13
conda activate groqflow_env
```

Step 2: Pip install GroqFlow

Install the **groqflow** package into your conda virtual environment:

```
pip install --upgrade pip
cd groqflow
pip install -e .
```

where **groqflow** is the directory where you cloned the **GroqFlow** repo in the **prerequisites**.

Install PyNaCl

```
python -m pip install PyNaCl
```

Your output will look something like:

```
Collecting PyNaCl
  Using cached PyNaCl-1.5.0-cp36-abi3-
manylinux_2_17_x86_64.manylinux2014_x86_64.manylinux_2_24_x86_64.whl (856
kB)
Requirement already satisfied: cffi>=1.4.1 in
/home/wilsonb/.local/lib/python3.8/site-packages (from PyNaCl) (1.15.1)
Requirement already satisfied: pycparser in
/home/wilsonb/.local/lib/python3.8/site-packages (from cffi>=1.4.1->PyNaCl)
(2.21)
Installing collected packages: PyNaCl
ERROR: pip's dependency resolver does not currently take into account all
the packages that are installed. This behaviour is the source of the
following dependency conflicts.
groqflow 2.1.1 requires protobuf==3.19.4, but you have protobuf 3.20.1
which is incompatible.
groqflow 2.1.1 requires pyyaml==6.0, but you have pyyaml 5.4 which is
incompatible.
Successfully installed PyNaCl-1.5.0
```

Ignore the error message.

Step 3: Add GroqWare Suite to Python Path

This adds the **Groq** tools to your path:

```
export PYTHONPATH="/opt/groq/runtime/site-packages:$PYTHONPATH"
```

Step 4: Rock-It with groqit()

To confirm that you're setup correctly, navigate to the examples folder at **groqflow/examples/** and run the **hello_pytorch_world.py** example:

```
cd examples/
python hello_pytorch_world.py
```