Process to Install **TensorFlow 2.0 GPU** (and make it work)

Notes:

* All files I used following the [YouTube example](https://www.youtube.com/playlist?list=PLhA3b2k8R3t2Ng1WW_7MiXeh1pfQJQi_P) are on a github page: <https://github.com/baaggg/CNN-Attempt-from-example>
  + I used my 40 audio files from Capstone instead of his own 300 audio files
* Requires an “approved” NVIDIA GPU
  + See if your GPU is on this list somewhere: <https://developer.nvidia.com/cuda-gpus>
* Windows 10 (**64bit**)
* Python 3 (Specifically a **64bit Python version**, i.e. 3.7.6)

What I did in order:

1. Uninstalled Python 3.7.4 (32bit)
   1. “Add or Remove programs” 🡪 Search bar “Python” 🡪 Uninstall
2. Installed Python 3.7.6 (64bit)
   1. 64bit required for TensorFlow (TF)
   2. <https://www.python.org/downloads/release/python-376/>
3. Re-install and/or upgrade Python libraries
   1. In cmd: *python -m pip install --user numpy*
   2. Regular *pip install [library]* wouldn’t work for me, might work for you
4. **Important:** When I uninstalled Python 3.7.4 and downloaded 3.7.6, my PATH environment variable(s) changed

from:C:\Users\brand\AppData\**Roaming**\Python\Python37\site-packages

to: C:\Users\brand\AppData\**Local**\Programs\Python\Python37\Lib\site-packages

I tried adding the Roaming path to PATH but Python 3.7.6 would never access that directory. I had to manually uninstall every single package from the Roaming path specifically, and then reinstall every single package manually. Doing *pip uninstall tensorflow* in the specific Roaming directory did absolutely nothing, and unless I uninstalled every package individually they were never actually gone. If you don’t uninstall packages one-by-one, no matter where you *pip install* them Windows pulls up cached files and just places them back where they used to be which doesn’t help. I manually uninstalled the 2 or 3 dozen TensorFlow modules/packages manually to fix this stupid issue: <https://i.imgur.com/ZIT0seD.gifv>

1. When I did  *import tensorflow as tf*  in Python, it gave me a new error:
   1. *dynamic library not loaded* ***dlerror: cudart64\_100.dll*** *not found*
   2. This dude supplies the *cudart64\_100.dll file*, and you need an NVIDIA Toolkit
      * Article that helped me: <https://www.joe0.com/2019/10/19/how-resolve-tensorflow-2-0-error-could-not-load-dynamic-library-cudart64_100-dll-dlerror-cudart64_100-dll-not-found/>
   3. When downloading the NVIDIA CUDA Toolkit, get the version 10.0 to make it easier:
      * <https://developer.nvidia.com/cuda-10.0-download-archive>
      * TensorFlow 2 requires ***name64\_100.dll*** files, so just get this version from September 2018 to make it easier on yourself
2. Running this script to do the Fourier Transform of an audio file:
   1. [*https://i.imgur.com/h3NcLGx.png*](https://i.imgur.com/h3NcLGx.png)
   2. Tests *tensorflow.signal* module, looks like it works
3. Followed an example online for a Convolutional Neural Network and it creates a NN with TensorFlow 2 GPU with no issues
   1. Article I followed: <https://becominghuman.ai/building-an-image-classifier-using-deep-learning-in-python-totally-from-a-beginners-perspective-be8dbaf22dd8>
   2. Script: <https://i.imgur.com/Q9dOFnq.png>
   3. Output: <https://i.imgur.com/RESrMCb.png>
4. There is some dumb shit involved using TF with GPU so you need **cuDNN** which should have all the “dynamic libraries” for all the Neural Network computations and algorithms
   1. cudNN install page: <https://docs.nvidia.com/deeplearning/sdk/cudnn-install/index.html#install-windows>
   2. After following some steps, I got cuDNN v7.6.5 for CUDA 10.2 hoping it had the right .dll file, and it did: (*cudnn64\_7.dll*)
   3. Take [these files](https://i.imgur.com/pTMBrfe.png) from the *cuda* folder in the cuDNN zip file and copy/paste them into directly into your [GPU Computing Toolkit directory](https://i.imgur.com/BkLU0Dm.png):
      * *C:\Program Files\NVIDIA GPU Computing Toolkit\CUDA\v10.2*
5. Following [Seth Adam’s Deep Learning playlist on YouTube](https://www.youtube.com/playlist?list=PLhA3b2k8R3t2Ng1WW_7MiXeh1pfQJQi_P), I copied basically everything except a few very tiny parameters; he makes a Conv NN.
   1. My “version” of his code: <https://i.imgur.com/HFnrxlj.gifv>
   2. The output when I build and train my Conv Neural Net: <https://i.imgur.com/UdruPcG.gifv>
6. Wasn’t able to make predictions due to lack of files and a syntax error I couldn’t figure out, but everything up to that point worked.
   1. Some error in the  *predictions.py*  file

Summary:

* I nearly copy/pasted my entire NN example from the YouTube playlist, but I needed to see if TensorFlow libraries worked or not
* Couldn’t technically use the model, but it was successfully built, trained, and saved for later use
* I have no idea how much slower Neural Network development is on PCs without GPU usage, but for a big architecture with lots of computations it will probably make a large difference
  + Maybe get away with regular TensorFlow (not -gpu) and cut out some stuff with NVIDIA?