TensorFlow C++ Installation and Use with Keras ML Models

John Michaloski

3 February 2020

Contents

[Python Tensoflow 1](#_Toc31627100)

[Installing Tensorflow 2 on Ubuntu using Pycharm 1](#_Toc31627101)

[Setting up Pycharm with correct Tensorflow 2 Environment on Ubuntu 2](#_Toc31627102)

[C++ Tensorflow 5](#_Toc31627103)

[Compiling Tensorflow for C++ on Ubuntu 5](#_Toc31627104)

[Prerequisites 5](#_Toc31627105)

[Building Tensorflow From Source Via Clone and Bazel 6](#_Toc31627106)

# Python Tensoflow

## Installing Tensorflow 2 on Ubuntu using Pycharm

# Create a new anaconda3 python environment

conda create --name gzrcsplugin\_env python=3.6.10

# Activate the new anaconda3 python environment

source activate gzrcsplugin\_env

pip install pandas

pip install numpy

pip install matplotlib

pip install sklearn

pip install keras

pip install tensor\*

pip3 install --upgrade tensorflow

pip install --upgrade pandas

python -m pip install pandas

# Run pycharm in gzrcs environment (bash script to invoke although it can be installed with apt-get install and invoked without bash – but may not have correct environment variable.

/usr/local/pycharm-community-2017.3.3/bin/pycharm.sh

## Setting up Pycharm with correct Tensorflow 2 Environment on Ubuntu

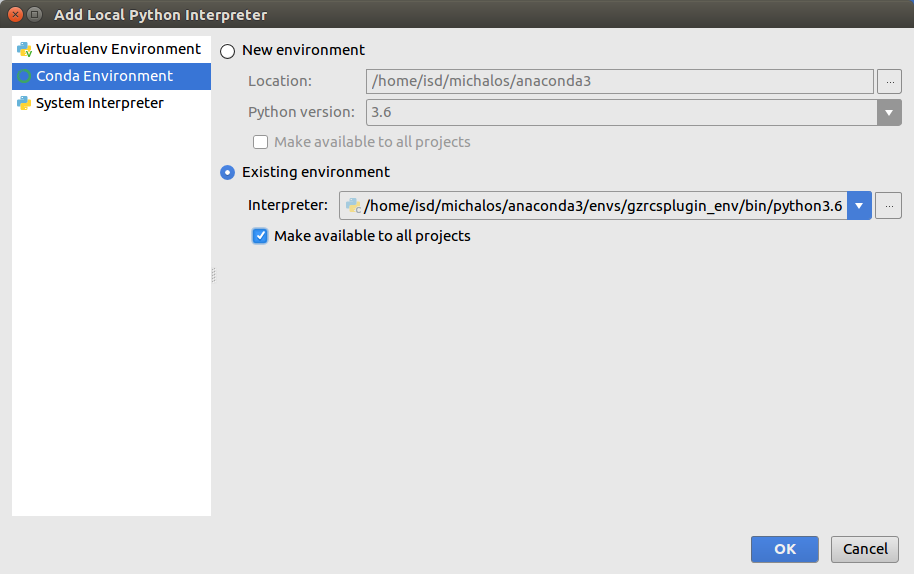
If you are using Pycharm (community) the following section describes getting the IDE up to speed to code in Tensorflow 2 (handling import, choosing correct Python interpreter, etc.)

Even if you set up the Python 3.6 correctly in Anaconda, Pycharm may not find it, or may not list it in its possible Python interpreters under the File->Settings:

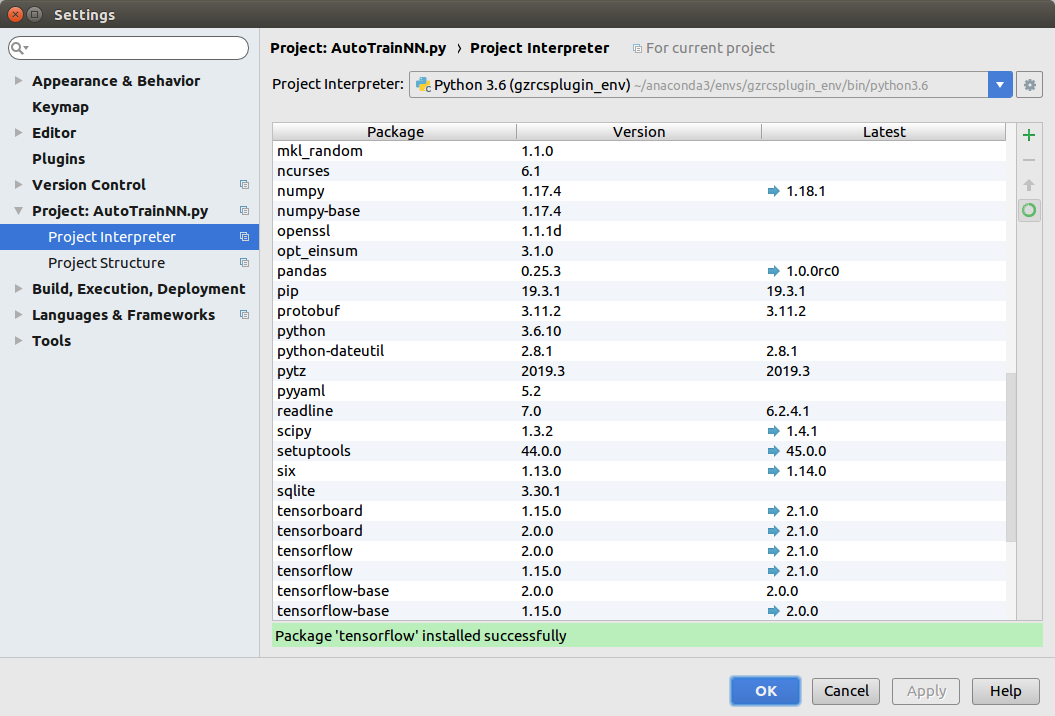
A screenshot of a computer

Description automatically generated

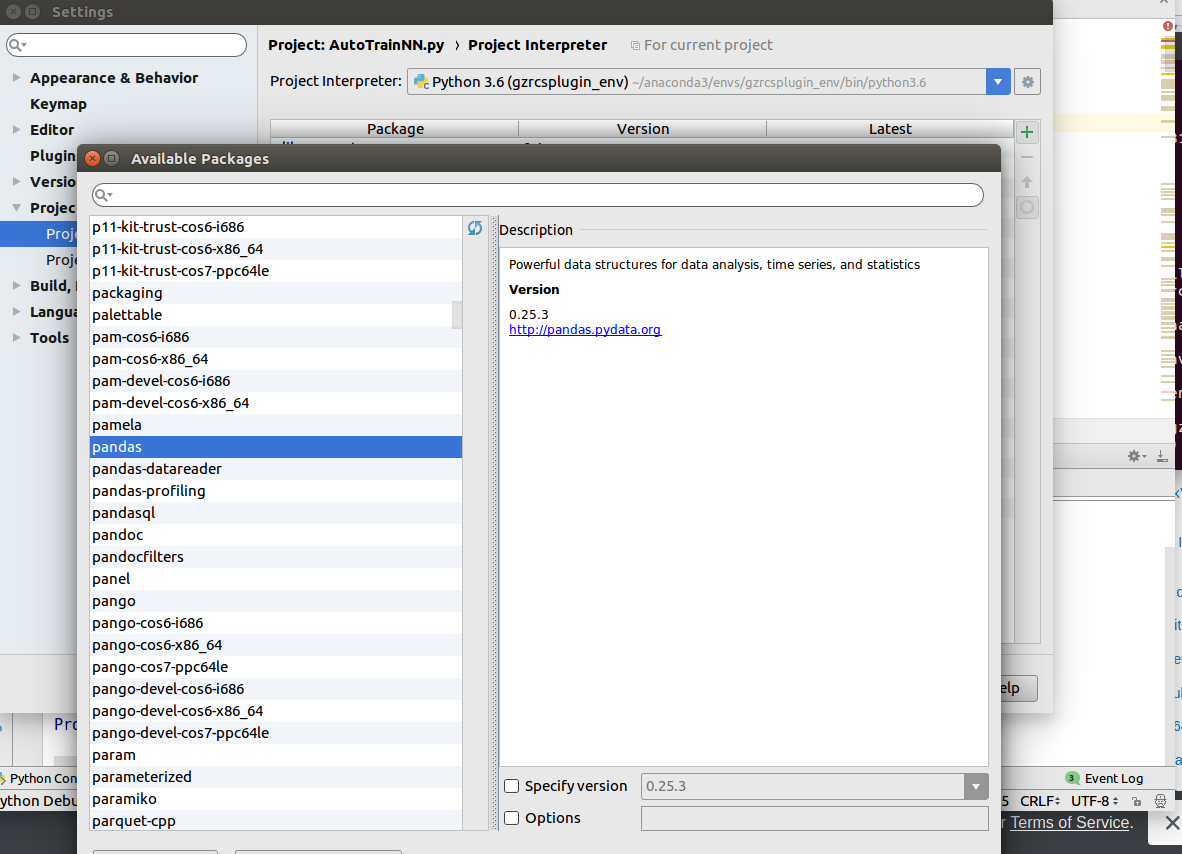
If the settings do not show the Python 3.6 interpreter, you can add it using the [+] sign above:



Now you can select the appropriate Python 3.6 interpreter and then add the required Python packages using the [+] button next to the packages list



For example, the pandas Python package is used so you need to install if as shown below:



Of note, you do not need to add the Keras packages as these are part of the

# C++ Tensorflow

## Compiling Tensorflow for C++ on Ubuntu

Several web sites offer explanations on how to build Tensorflow from source to be used by C++. In our case, the target platform was Ubuntu 16.04 (Xenial).

### Prerequisites

These were the prerequisites for building/compiling Tensorflow for C++/

1. No sudo right
2. Must install downgrade your bazel installation to version 0.26.1 or lower
3. Ubuntu 16.04 instsallation
4. Tensorflow 2.0

The flowing steps were taken:

1. install bazel
   1. Navigate to https://github.com/bazelbuild/bazel/releases?after=0.28.0
   2. Under assets download bazel-0.26.1-installer-linux-x86\_64.sh
2. cd Download
3. Added run permission of ./bazel-0.26.1-installer-linux-x86\_64.sh
4. ./bazel-0.26.1-installer-linux-x86\_64.sh
5. Installed in /usr/local

### Building Tensorflow From Source Via Clone and Bazel

https://tuanphuc.github.io/standalone-tensorflow-cpp/

1. cd /usr/local
2. git clone https://github.com/tensorflow/tensorflow.git
3. cd tensorflow
4. git checkout branch\_name # r1.9, r1.10, etc. I did: r2.0
5. ./configure

(YOU SHOULD HAVE PYTHON 3 INSTALLED: configure will find it.

answered default or no to all except for what?

/usr/local/anaconda3/envs/gzrcsplugin\_env/lib/python3.6/site-packages) as I had

previously installed python 3.6 using anaconda3 under virtual environment gzrcsplugin\_env)

1. bazel build //tensorflow:libtensorflow\_cc.so --config=v2 --config=nonccl

this did not include GPU support, or installation died otherwise.

1. /usr/local/tensorflow/tensorflow/contrib/makefile/build\_all\_linux.sh

Installing ADSL and PROTOBUF that are in makefile/contrib wtih bash shell will build:

/home/isd/michalos/build/TEstBuildTf/debug/TestBuildTf: error while loading shared libraries: libtensorflow\_framework.so.2: cannot open shared object file: No such file or directory

/home/isd/michalos/build/TEstBuildTf/debug/TestBuildTf: error while loading shared libraries: libtensorflow\_cc.so.2: cannot open shared object file: No such file or directory

2020-01-22 16:28:14.803482: I ../../src/TensorFlowCpp/TestBuildTf/neuralnet.cpp:34] Tensor<type: float shape: [18431,12] values: [-0.2927 -0.4952 -0.4956...]...>

2020-01-22 16:28:14.803678: I ../../src/TensorFlowCpp/TestBuildTf/neuralnet.cpp:35] Tensor<type: float shape: [18431,6] values: [0.3379 -0.0708 -0.9385...]...>

2020-01-22 16:28:14.857802: I tensorflow/core/platform/cpu\_feature\_guard.cc:142] Your CPU supports instructions that this TensorFlow binary was not compiled to use: SSE4.1 SSE4.2 AVX

2020-01-22 16:28:14.962889: F ../../src/TensorFlowCpp/TestBuildTf/neuralnet.cpp:157] Non-OK-status: session.Run({{x, x\_data}, {y, y\_data}}, {loss}, &outputs) status:

Invalid argument: Matrix size-incompatible: In[0]: [18431,12], In[1]: [3,3]

[[{{node MatMul}}]]