Setting Up Caffe2 with SQL Python

[Caffe2](https://caffe2.ai/) is an open source deep learning framework from Facebook, built upon the original popular deep learning framework Caffe, with modularity and scalability in mind. In the recent release of SQL Server 2017 CTP 2.0, Python script is supported, which makes it possible to run Caffe2 models in SQL. However, Caffe2 has some limitations:

* It does not offer a pre-built binary for Windows,
* It supports Python 2.7, while SQL Python is Python 3.5

After exploration, we managed to build Caffe2 with SQL Python. This document will walk you through this process. The basic steps are as follows:

* Create an Azure VM
* Install SQL Server 2017 CTP2
* Install prerequisites for Caffe2
* Clone and Build Caffe2
* Set up Caffe2 in Python environment
* Test Caffe2 in SQL

1. Create an Azure VM:

Create a Windows Server 2016 Virtual Machine on Azure Portal. To enable GPU, will need to select HDD as disk type, and NC series infrastructure, which features NIVDIA Tesla Accelerator. Note, N-series VMs are more expensive than general purpose VMs.

2. Install SQL Server 2017 CTP2:

The public preview can be downloaded from: <https://www.microsoft.com/en-us/sql-server/sql-server-2017>. Make sure SQL Server Management Studio is also installed.

One key feature in CTP2.0 for SQL Server 2017 is Microsoft Machine Learning Services, which enables Python script to be run in-database directly.

3. Install prerequisites for Caffe2

* Install Visual Studio:
  + since CUDA doesn’t work with VS 2017, to enable GPU, must install VS 2015 (community version is fine)
  + for scenarios without GPU, install VS 2017 should work, make sure check C++/CLI support -> “Desktop Development with C++” in installation.
* Install CMAKE, add to system environment variable PATH
* Install Git for Windows
* Install NVIDIA graphics libraries:
  + GPU drivers: <https://docs.microsoft.com/en-us/azure/virtual-machines/windows/n-series-driver-setup>
  + CUDA8: <https://developer.nvidia.com/compute/cuda/8.0/Prod2/local_installers/cuda_8.0.61_win10-exe>
  + cuDNN: <https://developer.nvidia.com/cudnn> unzip the content to CUDA directory
* Install Python dependencies (pip install):
  + required packages: future, hypothesis, numpy, six
  + optional packages: flask, glog, graphviz, jupyter, matplotlib, pydot python-nvd3, pyyaml, requests, scikit-image, scipy, setuptools, tornado
  + check Caffe2 site <https://caffe2.ai/docs/getting-started.html?platform=windows&configuration=compile> for the latest requirements

4. Clone and build Caffe2:

* Clone Caffe2 source:
  + Open up a Developer Command Prompt for VS2015, find an appropriate place to clone the repo, e.g. C:\Caffe2Repo, run the command below:

git clone --recursive https://github.com/caffe2/caffe2.git

Make sure do not use a user-dependent folder (such as C:\Users\pythonadmin\...), as SQL Server will not be able to access it.

* Build protobuf:
  + Open up a Developer Command Prompt for VS. Go to C:\Caffe2Repo\caffe2\scripts\, edit file build\_host\_protoc.bat with the following line:

SET CMAKE\_GENERATOR="Visual Studio 14 2015 Win64"

* + Run build\_host\_protoc.bat to build protobuf
* Build Caffe2:
  + update build\_window.bat file with -DUSE\_CUDA=ON and SET CMAKE\_GENERATOR="Visual Studio 14 2015 Win64",
  + run build\_window.bat in Developer Command Prompt for VS2015. It will take about 50 min to complete.
  + There might be warnings when build finishes, since not all Python dependencies are installed/built beforehand. But if you encounter any error, the best place to get help is [Caffe2’s GitHub forum](https://github.com/caffe2/caffe2/issues).

5. Set up Caffe2 in Python environment

* Copy

caffe2\_pybind11\_state.pyb,

caffe2\_pybind11\_state\_gpu.pyb

from C:\Caffe2Repo\caffe2\build\caffe2\python to Python folder: C:\Program Files\Microsoft SQL Server\MSSQL14.MSSQLSERVER\PYTHON\_SERVICES\DLLs\

* ADD system environment variable PYTHONPATH = C:\Caffe2Repo\caffe2\build;C:\Caffe2Repo\caffe2\build\caffe2\python\Release

6. Test Caffe2 in SQL

* Open SQL Server Management Studio, connect to your SQL Server.
* Create a new query, then copy paste the script below to it:

EXECUTE sp\_execute\_external\_script

@language = N'Python',

@script = N'

from matplotlib import pyplot

import numpy as np

import time

from caffe2.python import core, workspace

from caffe2.proto import caffe2\_pb2

print("Current blobs in the workspace: {}".format(workspace.Blobs()))

print("Workspace has blob ''X''? {}".format(workspace.HasBlob("X")))

',

@input\_data\_1 = N'SELECT 1 AS hello;'

It will output messages like below:

