XNNC Quick Start

XIN LIU (samshin@xilinx.com)

The purpose of this document is to help users get familiar with the installation and basic usage of XNNC.

1 Prerequisites

Before the installation of XNNC, it is guaranteed that PyXir, a Python binding of XIR, has already installed in the system environment. Besides, Python 3.6.8 or above is required.

1.1 Environment Setup

In the root directory of XNNC4XIR source, run the command below to install required Python packages.

```
pip install –user –requirement requirements.txt
```

The following packages will be installed:

- marshmallow>=3.0.0rc5
- tqdm>=4.31.1
- numpy>=1.16.4
- graphviz>=0.11.1
- protobuf>=3.6.1

2 Installation Binary

To generate installation binary file (.whl), run the command below in the root directory of XNNC4XIR:

```
python3 setup.py sdist bdist_wheel
```

In the root directory, after that, a folder (named *dist*) is created, in which the installation binary file suffixed with *.whl* will be present.

3 Installation

To install XNNC application on your system, follow the steps below:

- In terminal, change current working directory to where XNNC installation file (for example, named *xnnc-0.0.1-py3-none-any.whl*) reside.
- Run command: `pip3 install xnnc-0.0.1-py3-none-any.whl`
- To check if the installation is successful, run command: `xnnc-run --help`

4 Basic Usages

For now, XNNC4XIR only supports converting TensorFlow and Caffe models (with customized fixed neuron layer) into XIR Graph. To achieve the conversion of TensorFlow ResNet50, for example, simply run the following command in terminal:

```
xnnc-run --model /path/to/resnet50.pb --type tensorflow --layout NHWC –png
```

The command options can be shown by running `xnnc-run --help`.