

Instructions for running ImageNet experiments

This code can be used as the supplemental material for the paper: "Towards a Lossless Conversion for Spiking Neural Networks with Negative Spike Dynamics". (Submitted to *Advanced Intelligent Systems*, Wiley, July, 2023) .

Citation:

To be completed.

Features:

- To verify the effect of our conversion algorithm on MobileNet, we reproduce the [MobileNet](#) structure and train it on the [ImageNet dataset](#). Similarly, we set the quantization level k to the choice of $\{0,1,2\}$, and record the experimental results including accuracies and simulation time steps. The training details can be found in the original paper and our codes. It should be noted the training procedure is accomplished for 60 epochs and the simulation of SNN-based mobilenet is really slow (both are time-consuming). We only test it on the part of complete imagenet dataset. Our conversion algorithm also achieved a lossless conversion from the ANN-version MobileNet to the SNN-version one. To our knowledge, to perform SNN simulations on the complete imagenet dataset is very rare, because it need quite long simulation time and vast computing resources. Therefore, we only add these results in this open-source github for readers, but choose not to discuss it in our paper.

File overview:

- [README.md](#) - this readme file.

[Other files](#) include several scripts which support imagenet dataset parse and distributed training.

Requirements

Dependencies and Libraries:

- python 3.5 (<https://www.python.org/> or <https://www.anaconda.com/>)
- tensorflow_gpu 1.2.1 (<https://github.com/tensorflow>)
- tensorlayer 1.8.5 (<https://github.com/tensorlayer>)
- CPU: Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz
- GPU: Tesla V100

Please refer to the original MobileNet paper for more information.

Installation:

To install requirements,

```
pip install -r requirements.txt
```

Datasets:

- ImageNet: [dataset](#), [preprocessing](#)
- [MobileNet](#)

Run the code:

Firstly, you need to download the ImageNet dataset personally.

Then, for example (training, $k=0$, MobileNet, imagenet):

```
$ cd MobileNet_on_Imagenet
$ python running_distributed_mobilenet.py --k 0 --resume False --learning_rate
0.01 --mode 'training'
```

Results

Please check these results in our paper.

More question:

- There might be a little difference of results for multiple training repetitions, because of the randomization.
- Please feel free to reach out here or email: xxx@xxx, if you have any questions or difficulties. I'm happy to help guide you.