Instructions for running CIFAR10 experiments

File overview:

- `README_CIFAR10.md` - this readme file for CIFAR10.
- `spiking_ulils.py` - the functions of spiking convolution and linear.
- `figs` - visualization folder for SNN performance.
    - `accuracy_speed.py` - the accuracy versus speed script for `spiking CNN 1` and `spiking CNN 2` with different quantization precisions on CIFAR10.
    - `sops.py` - the computing operations script for `spiking CNN 1` and `spiking CNN 2` with different quantization precisions on CIFAR10.
    - `sparsity.py` - the spike sparsity script for `spiking CNN 1` and `spiking CNN 2` with different quantization precisions on CIFAR10.
- `CNN_1` - CNN 1 for CIFAR10.
    - `tensorlayer` - our provided tensorlayer package.
    - `Quant_CNN1_CIFAR10.py` - the training script for `CNN 1` with optional quantization precision *k* on CIFAR10.
    - `Spiking_CNN1_CIFAR10.py` - the evaluation script for `spiking CNN 1` with optional quantization precision *k* on CIFAR10.
    - `FP32_CNN1_CIFAR10.py` - the training script for `CNN 1` with `full precision (float32)` on CIFAR10.
- `CNN_2` - CNN 1 for CIFAR10.
    - `tensorlayer` - our provided tensorlayer package.
    - `Quant_CNN2_CIFAR10.py` - the training script for `CNN 2` with optional quantization precision *k* on CIFAR10.
    - `Spiking_CNN2_CIFAR10.py` - the evaluation script for `spiking CNN 2` with optional quantization precision *k* on CIFAR10.
    - `FP32_CNN2_CIFAR10.py` - the training script for `CNN 2` with `full precision (float32)` on CIFAR10.

ANN Training

Before running:

- Please note your default dataset folder will be `./data`
- please modify the command line parameters: `--resume True`, and `--learning_rate 0.0001` for another 100 epochs after the first 100 epochs. Totally, 200 training epoch need be performed.

Run the code:

Run the code:

for example (training, *k=0*, CNN1, CIFAR10):

```
$ cd CIFAR10/CNN_1
$ python Quant_CNN1_CIFAR10.py  --k 0 --resume False --learning_rate 0.001 --mode
'training'
```

finally, it will generate the corresponding model files including: `checkpoint`, `model_CIFAR10_advanced.ckpt.data-00000-of-00001`, `model_CIFAR10_advanced.ckpt.index`, `model_CIFAR10_advanced.ckpt.meta` and `model_cifar_10.npz`.

ANN Inference

Run the code:

for example (inference, *k=0*, CNN1, CIFAR10):

```
$ python Quant_CNN1_CIFAR10.py --k 0 --resume True  --mode 'inference'
```

Then, it will print the corresponding ANN test accuracy.

SNN inference

Run the code:

for example (inference, *k=0*, spiking CNN1, CIFAR10):

```
$ python Spiking_CNN1_CIFAR10.py --k 0
```

it will generate the corresponding log files including: `accuracy.txt`, `sop_num.txt`, `spike_collect.txt` and `spike_num.txt` in `./figs/k0/`.

Visualization

Accuracy versus speed:

```
$ cd figs
$ python accuracy_speed.py
```

Firing sparsity:

```
$ python sparsity.py
```

Computing operations:

```
$ python sops.py
```

Results

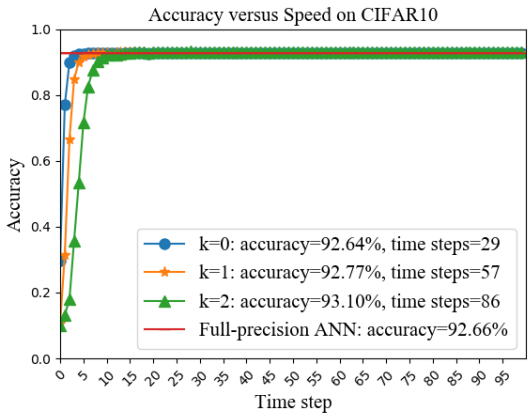
Our proposed method achieves the following performance on :

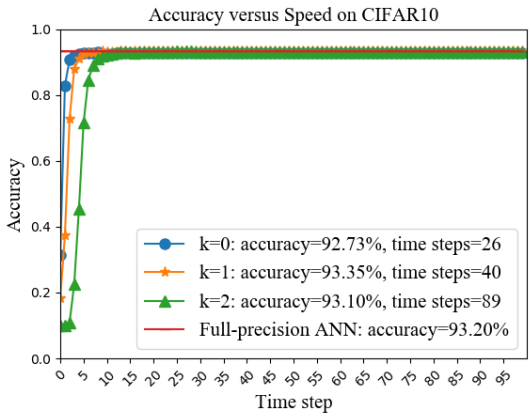
CIFAR10:

Quantization Level	Network Size	Epochs	ANN	SNN	Time Steps
full-precision	96C3-256C3-P2-384C3-P2-384C3-256C3-P2-1024-1024	200	92.66%	N/A	N/A
k=1	96C3-256C3-P2-384C3-P2-384C3-256C3-P2-1024-1024	200	92.77%	92.77%	57
full-precision	128C3-256C3-P2-512C3-P2-1024C3-512C3-P2-1024-512	200	93.20%	N/A	N/A
k=1	128C3-256C3-P2-512C3-P2-1024C3-512C3-P2-1024-512	200	93.35%	93.35%	40

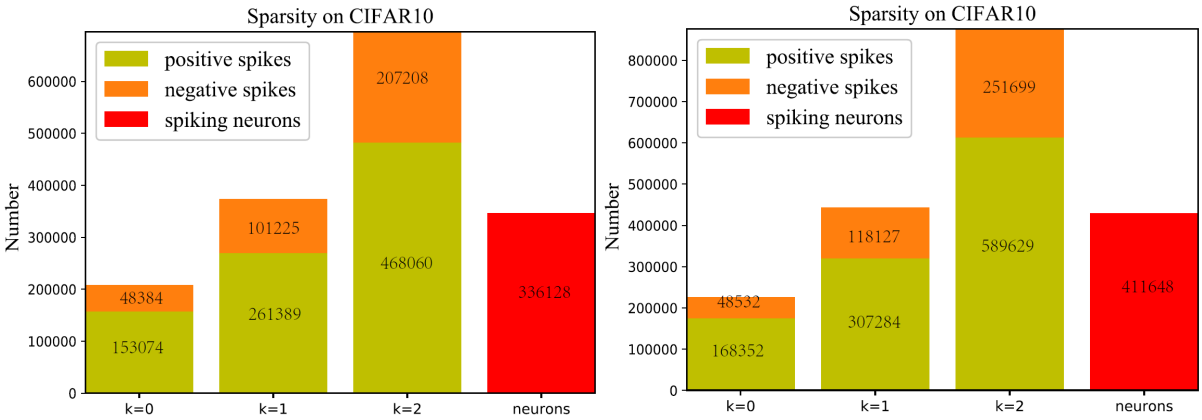
Accuracy versus speed:

CNN 1: 96C3-256C3-P2-384C3-P2-384C3-256C3-1024-1024
CNN 2: 128C3-256C3-P2-512C3-P2-1024C3-512C3-1024-512

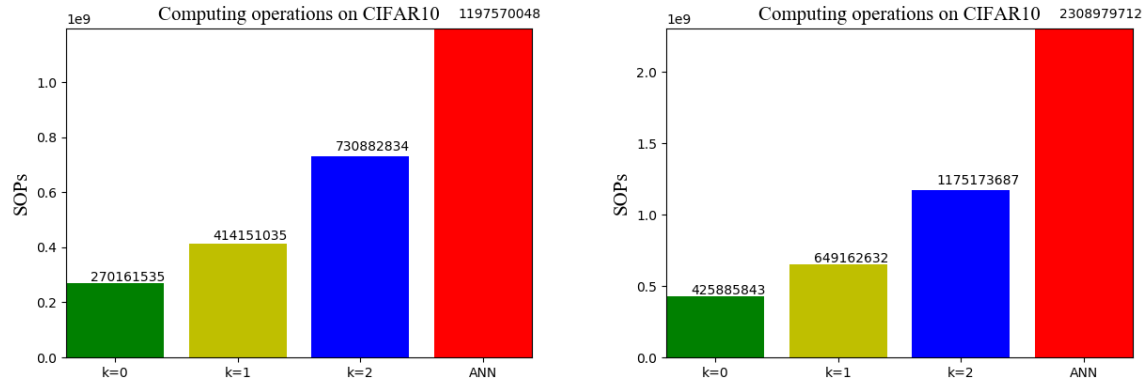




Firing sparsity:



Computing operations:



Notes

- We do not consider the synaptic operations in the input encoding layer and the spike outputs in the last classification layer (membrane potential accumulation instead) for both original ANN counterparts and converted SNNs.
- We also provide some scripts for visualization in `./figs`, please move to this folder and directly run the three scripts.

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.