Attention Model using SNN
UCF REU 2022 Week 4

Advisor: Dr. Gonzalo Vaca-Castano
Student: Connor Cabrera
Weekly Overview

- I was given two code bases that convert ANNs to SNNs
- SNN Tool Box
  - Automated Conversion Tool
  - Can take Keras & PyTorch Models as input
  - Can simulate the execution of the SNN since we don't possess a neuromorphic processor
- Spiking YOLO v3
  - Trains and Tests a regular YOLO Object Detection Model then converts it to an SNN
Difficulties

SNN Tool Box

- Not in active development
- Library is quite extensive and examining the code from scratch is a slow process
- Encountered problems in implementing SNNTB, but found an excellent tutorial from University of South Carolina:
  
  https://www.youtube.com/watch?v=FxTMH9VRsgk

Spiking YOLO

- Encountered Error: Passthrough is not supported, GL is disabled, ANGLE is
SNN Tool Box Results
## Supported features

<table>
<thead>
<tr>
<th>Supported features</th>
<th>Input model</th>
<th>Output model</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fully-connected</td>
<td>Keras</td>
<td>PyTorch</td>
</tr>
<tr>
<td>Convolution</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Depthwise separable convolution</td>
<td>✔️</td>
<td></td>
</tr>
<tr>
<td>Max-Pooling</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Average-Pooling</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Batch-Normalization</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Dropout</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Flatten</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Merge / Concatenate (Inception modules)</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Linear activation</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>ReLU activation</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Softmax activation</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Binary activation {-1, 1} or {0, 1}</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Binary weights {-1, 1}</td>
<td>✔️</td>
<td>✔️</td>
</tr>
<tr>
<td>Non-zero biases</td>
<td>✔️</td>
<td>✔️</td>
</tr>
</tbody>
</table>

### Output model

- Absorbed into prev. layer
- Removed for inference
- Replaced by ReLU
- Repl. by ReLU

Source: SNN Tool Box Documentation
LeNet MNIST Conversion
ANN acc: 98.96% | SNN acc: 98.96%
LeNet MNIST Conversion Example Output

Input image (class: 2)

Spiketrains of layer 7Dense_10

neuron index vs. time [ms]
Assignment 1: CIFAR-100 Conversion | $T = 18$

ANN acc: 66.67% | SNN acc: 38.54%

Correlation between ANN activations and SNN spikerates, averaged over 32 samples.
Assignment 1: CIFAR-100 Conversion Example Output | T = 18

Input image (class: 28)

Spiketrains of layer 11Dense_100

neuron index

time [ms]
Assignment 1: CIFAR-100 Conversion | T = 50
ANN acc: 66.67% | SNN acc: 66.67%

Correlation between ANN activations and SNN spikerates, averaged over 32 samples
Assignment 1: CIFAR-100 Conversion Example Output | T = 50
What’s Next For Me?

- Fixing the Error and getting Spiking YOLO to work
- Analysis into the inner workings of SNN Tool Box
  - At this point I can convert an ANN to an SNN, but I need to deepen my understanding on how that is being done