ECE 408 Final Project

applied_indexing

Nishant Dash, dash3
Rohan Tikmany, tikmany2
Zizhen Liu, zliu99

Index

Milestone 1

On keeping the yml file as it is, the output generated is:

```
New Inference
Loading fashion-mnist data... done
Loading model... done
EvalMetric: {'accuracy': 0.8673}
```

On modifying the yml file to run `m1.2.py`, the output generated is:

```
New Inference
Loading fashion-mnist data... done
Loading model...[22:06:14] src/operator/./../cudnn_algoreg-inl.h:112: Runni
```

```
ng performance tests to find the best convolution algorithm, this can take a while... (setting env variable MXNET_CUDNN_AUTOTUNE_DEFAULT to 0 to disable)
done
EvalMetric: {'accuracy': 0.8673}
```

And finally, modifying the yml file to generate an **NVPROF** Profile, from which we can see that some of the most time consuming kernels are,

Kernel	*Time*	*% of total time used*
void cudnn::detail::implicit_convolve_sgemm	50.460 ms	37.08%
void cudnn::detail::activation_fw_4d_kernel	19.383 ms	14.24%
void cudnn::detail::pooling_fw_4d_kernel	14.502 ms	10.66%

The remaining calls all fall under 1% of time used.

The **NVPROF** profile generated shows us under one section, which kernel call is spending how much time in absolute value and how much % of total execution time, and under another section, the same statistics for CUDA API calls.

Milestone 2

- Running the code on the large dataset, we see the following NVPROF profile. The time spent on the convolution layer is 0.115775s

```
* Running nvprof python m2.1.py
New Inference
Loading fashion-mnist data... done
Loading model... done
```

```
Op Time: 11.643823
Correctness: 0.8562 Model: ece408-high
```

- Running the code on the small dataset, we see the following NVPROF profile. The time spent on the convolution layer is 0.115775s

```
* Running nvprof python m2.1.py ece408-low 100
New Inference
Loading fashion-mnist data... done
Loading model... done
Op Time: 0.115775
Correctness: 0.63 Model: ece408-low
```

As seen above, the expected correctness for the datasets was obtained.

Milestone 3

- Running the base GPU implementation on the ***ece408-high*** dataset of size 10000 yielded the following results with the expected *correctness* of 0.8562 and an *op time* of 0.492485s:

Kernel	***Time***	***Number of calls***	***% of total time used***
forward_kernel	472.05 ms	1	81.56%
sgemm_sm35_ldg_tn_128x8x256x16x32	38.754 ms	1	6.70%

However our fastest recorded instance of *op time* on the *high* dataset is 0.405591s and on the *low* dataset is 0.321678s (both of default size 10000).

```
* Running /usr/bin/time -f "%Uuser %Ssystem %eelapsed" python /eval-scripts/m3.1.py ece408-high
New Inference
Loading fashion-mnist data... done
```

```
Loading model... done
Op Time: 0.405591
Correctness: 0.8562 Model: ece408-high
2.00user 1.12system 2.64elapsed
* Running /usr/bin/time -f "%Uuser %Ssystem %eelapsed" python /eval-scripts/m3.1.py ece408-low
New Inference
Loading fashion-mnist data... done
Loading model... done
Op Time: 0.321678
Correctness: 0.629 Model: ece408-low
1.58user 0.98system 2.10elapsed
```

Division of Labor

Task	***Nishant***	***Rohan***	***Zizhen***
Understanding Task	•	•	•
Implementing simple GPU forward convolution	•	•	•
Report	•	•	•
