

Cerebras

Homework

Run BERT example with different batch sizes like 512, 2048 and observe the performance difference, results on Polaris, it ran faster it fit into the cache more effectively, because data loading is expensive. (here is the result when I varied the context window instead from 128 to 512 and observed very interesting performance differences).

```
2024-04-10 01:46:33,156 INFO: Exploring floorplans
2024-04-10 01:46:46,450 INFO: Exploring data layouts
2024-04-10 01:47:06,543 INFO: Optimizing memory usage
2024-04-10 01:47:35,530 INFO: Gradient accumulation trying sub-batch size 32...
2024-04-10 01:47:42,095 INFO: Exploring floorplans
2024-04-10 01:47:50,057 INFO: Exploring data layouts
2024-04-10 01:48:05,776 INFO: Optimizing memory usage
2024-04-10 01:48:43,241 INFO: Gradient accumulation trying sub-batch size 256...
2024-04-10 01:48:48,832 INFO: Exploring floorplans
2024-04-10 01:49:05,056 INFO: Exploring data layouts
2024-04-10 01:49:26,298 INFO: Optimizing memory usage
2024-04-10 01:50:05,441 INFO: Gradient accumulation trying sub-batch size 64...
2024-04-10 01:50:11,219 INFO: Exploring floorplans
2024-04-10 01:50:19,380 INFO: Exploring data layouts
2024-04-10 01:50:37,702 INFO: Optimizing memory usage
2024-04-10 01:51:12,278 INFO: Gradient accumulation trying sub-batch size 512...
2024-04-10 01:51:18,195 INFO: Exploring floorplans
2024-04-10 01:51:22,695 INFO: Exploring data layouts
2024-04-10 01:52:00,759 INFO: Optimizing memory usage
2024-04-10 01:52:44,163 INFO: Exploring floorplans
2024-04-10 01:52:46,518 INFO: Exploring data layouts
2024-04-10 01:53:19,689 INFO: Optimizing memory usage
2024-04-10 01:53:42,378 INFO: No benefit from gradient accumulation expected. Compile will proceed at original per-box batch
24 with 9 lanes

2024-04-10 01:53:42,420 INFO: Post-layout optimizations...
2024-04-10 01:53:52,080 INFO: Allocating buffers...
2024-04-10 01:53:54,727 INFO: Code generation...
2024-04-10 01:54:16,591 INFO: Compiling image...
2024-04-10 01:54:16,597 INFO: Compiling kernels
2024-04-10 01:56:24,537 INFO: Compiling final image
2024-04-10 01:59:33,444 INFO: Compile artifacts successfully written to remote compile directory. Compile hash is: cs_94652
23743
2024-04-10 01:59:33,539 INFO: Heartbeat thread stopped for wsjob-nqtjvizsdc5tp5gqpnben.
2024-04-10 01:59:33,543 INFO: Compile was successful!
2024-04-10 01:59:33,549 INFO: Programming Cerebras Wafer Scale Cluster for execution. This may take a few minutes.
2024-04-10 01:59:36,010 INFO: Defaulted to use the job-operator namespace as the usernode config /opt/cerebras/config_v2 on
cess to that namespace.
2024-04-10 01:59:36,361 INFO: Initiating a new execute wsjob against the cluster server.
2024-04-10 01:59:36,494 INFO: execute job id: wsjob-ftykmgeffhvmgx6z54xgsh, remote log path: /nl/wsjob/workdir/job-operator
/mkgeffhvmgx6z54xgsh
2024-04-10 01:59:46,539 INFO: Poll ingress status: Waiting for job running, current job status: Scheduled, msg: job is sche
2024-04-10 01:59:56,523 INFO: Poll ingress status: Waiting for job service readiness.
2024-04-10 02:00:16,565 INFO: Ingress is ready: Job ingress ready, poll ingress success.
2024-04-10 02:00:16,777 INFO: Preparing to execute using 1 CSX
2024-04-10 02:00:45,693 INFO: About to send initial weights
2024-04-10 02:01:26,624 INFO: Finished sending initial weights
2024-04-10 02:01:26,629 INFO: Finalizing appliance staging for the run
2024-04-10 02:01:26,665 INFO: Waiting for device programming to complete
2024-04-10 02:03:40,725 INFO: Device programming is complete
2024-04-10 02:03:41,652 INFO: Using network type: ROCE
2024-04-10 02:03:41,653 INFO: Waiting for input workers to prime the data pipeline and begin streaming ...
2024-04-10 02:03:41,694 INFO: Input workers have begun streaming input data
2024-04-10 02:03:58,415 INFO: Appliance staging is complete
2024-04-10 02:03:58,421 INFO: Beginning appliance run
2024-04-10 02:04:19,191 INFO: | Train Device=CSX, Step=100, Loss=9.48438, Rate=4963.97 samples/sec, GlobalRate=4963.97 samp
2024-04-10 02:04:40,180 INFO: | Train Device=CSX, Step=200, Loss=8.35938, Rate=4912.80 samples/sec, GlobalRate=4920.96 samp
2024-04-10 02:05:01,266 INFO: | Train Device=CSX, Step=300, Loss=7.91406, Rate=4878.94 samples/sec, GlobalRate=4899.24 samp
2024-04-10 02:05:22,426 INFO: | Train Device=CSX, Step=400, Loss=7.54688, Rate=4855.14 samples/sec, GlobalRate=4884.11 samp
2024-04-10 02:05:43,439 INFO: | Train Device=CSX, Step=500, Loss=7.46875, Rate=4865.99 samples/sec, GlobalRate=4881.93 samp
2024-04-10 02:06:04,386 INFO: | Train Device=CSX, Step=600, Loss=7.39844, Rate=4879.45 samples/sec, GlobalRate=4883.01 samp
2024-04-10 02:06:25,734 INFO: | Train Device=CSX, Step=700, Loss=7.35156, Rate=4829.78 samples/sec, GlobalRate=4870.49 samp
2024-04-10 02:06:47,018 INFO: | Train Device=CSX, Step=800, Loss=7.25000, Rate=4818.66 samples/sec, GlobalRate=4863.00 samp
2024-04-10 02:07:08,403 INFO: | Train Device=CSX, Step=900, Loss=7.21094, Rate=4800.43 samples/sec, GlobalRate=4854.58 samp
2024-04-10 02:07:29,724 INFO: | Train Device=CSX, Step=1000, Loss=7.07812, Rate=4801.83 samples/sec, GlobalRate=4849.35 sam
2024-04-10 02:07:29,725 INFO: Saving checkpoint at step 1000
2024-04-10 02:08:07,141 INFO: Saved checkpoint model_dir_bert_large_pytorch/checkpoint_1000.mdl
2024-04-10 02:08:50,322 INFO: Heartbeat thread stopped for wsjob-ftykmgeffhvmgx6z54xgsh.
2024-04-10 02:08:50,330 INFO: Training completed successfully!
2024-04-10 02:08:50,330 INFO: Processed 1024000 sample(s) in 211.16228866 seconds.
(venv_cerebras_pt) [clarexie@cer-login-03 bert]$ |
```

The above is 128

Next we will do 512:

```
2024-04-10 02:30:03,658 INFO: Gradient accumulation trying sub-batch size 128...
2024-04-10 02:30:10,341 INFO: Exploring floorplans
2024-04-10 02:30:21,874 INFO: Exploring data layouts
2024-04-10 02:30:55,002 INFO: Optimizing memory usage
2024-04-10 02:32:10,138 INFO: Gradient accumulation trying sub-batch size 32...
2024-04-10 02:32:16,668 INFO: Exploring floorplans
2024-04-10 02:32:23,780 INFO: Exploring data layouts
2024-04-10 02:32:41,871 INFO: Optimizing memory usage
2024-04-10 02:33:10,583 INFO: Gradient accumulation trying sub-batch size 256...
2024-04-10 02:33:17,494 INFO: Exploring floorplans
2024-04-10 02:33:35,254 INFO: Exploring data layouts
2024-04-10 02:34:05,529 INFO: Optimizing memory usage
2024-04-10 02:35:05,254 INFO: Gradient accumulation trying sub-batch size 64...
2024-04-10 02:35:11,330 INFO: Exploring floorplans
2024-04-10 02:35:19,597 INFO: Exploring data layouts
2024-04-10 02:35:38,426 INFO: Optimizing memory usage
2024-04-10 02:36:23,703 INFO: Gradient accumulation trying sub-batch size 512...
2024-04-10 02:36:29,658 INFO: Exploring floorplans
2024-04-10 02:36:33,619 INFO: Exploring data layouts
2024-04-10 02:37:12,202 INFO: Optimizing memory usage
2024-04-10 02:38:10,893 INFO: Exploring floorplans
2024-04-10 02:38:12,928 INFO: Exploring data layouts
2024-04-10 02:38:53,426 INFO: Optimizing memory usage
2024-04-10 02:39:46,356 INFO: Gradient accumulation picked sub-batch size 512 with 11 lanes

2024-04-10 02:39:46,423 INFO: Post-layout optimizations...
2024-04-10 02:39:56,924 INFO: Allocating buffers...
2024-04-10 02:39:59,608 INFO: Code generation...
2024-04-10 02:40:12,420 INFO: Compiling image...
2024-04-10 02:40:12,425 INFO: Compiling kernels
2024-04-10 02:42:31,832 INFO: Compiling final image
2024-04-10 02:45:14,838 INFO: Compile artifacts successfully written to remote compile directory. Compile hash is: cs_253019740038351140

2024-04-10 02:45:14,907 INFO: Heartbeat thread stopped for wsjob-j3v7kj7w9tzyibxkt9ynqp.
2024-04-10 02:45:14,911 INFO: Compile was successful!
2024-04-10 02:45:14,917 INFO: Programming Cerebras Wafer Scale Cluster for execution. This may take a few minutes.
2024-04-10 02:45:17,247 INFO: Defaulted to use the job-operator namespace as the usernode config /opt/cerebras/config_v2 only has access to that namespace.

2024-04-10 02:45:17,637 INFO: Initiating a new execute wsjob against the cluster server.
2024-04-10 02:45:17,785 INFO: execute job id: wsjob-62bb2sg873frutuunbw3s, remote log path: /n1/wsjob/workdir/job-operator/wsjob-62bb2sg873frutuunbw3s

2024-04-10 02:45:27,852 INFO: Poll ingress status: Waiting for job running, current job status: Scheduled, msg: job is scheduled.
2024-04-10 02:45:37,872 INFO: Poll ingress status: Waiting for job service readiness.
2024-04-10 02:45:57,916 INFO: Ingress is ready: Job ingress ready, poll ingress success.
2024-04-10 02:45:58,098 INFO: Preparing to execute using 1 CSX
2024-04-10 02:46:24,816 INFO: About to send initial weights
2024-04-10 02:47:06,288 INFO: Finished sending initial weights
2024-04-10 02:47:06,292 INFO: Finalizing appliance staging for the run
2024-04-10 02:47:06,343 INFO: Waiting for device programming to complete
2024-04-10 02:49:28,922 INFO: Device programming is complete
2024-04-10 02:49:29,916 INFO: Using network type: ROCE
2024-04-10 02:49:29,917 INFO: Waiting for input workers to prime the data pipeline and begin streaming ...
2024-04-10 02:49:29,966 INFO: Input workers have begun streaming input data
2024-04-10 02:49:46,885 INFO: Appliance staging is complete
2024-04-10 02:49:46,889 INFO: Beginning appliance run
2024-04-10 02:50:53,525 INFO: | Train Device=CSX, Step=100, Loss=10.17188, Rate=1538.38 samples/sec, GlobalRate=1538.38 samples/sec
2024-04-10 02:52:00,479 INFO: | Train Device=CSX, Step=200, Loss=9.14062, Rate=1533.00 samples/sec, GlobalRate=1533.88 samples/sec
2024-04-10 02:53:07,531 INFO: | Train Device=CSX, Step=300, Loss=8.39062, Rate=1529.51 samples/sec, GlobalRate=1531.65 samples/sec
2024-04-10 02:54:14,446 INFO: | Train Device=CSX, Step=400, Loss=8.15625, Rate=1529.97 samples/sec, GlobalRate=1531.30 samples/sec
2024-04-10 02:55:21,168 INFO: | Train Device=CSX, Step=500, Loss=7.90625, Rate=1532.83 samples/sec, GlobalRate=1531.99 samples/sec
2024-04-10 02:56:28,011 INFO: | Train Device=CSX, Step=600, Loss=7.89062, Rate=1532.30 samples/sec, GlobalRate=1531.98 samples/sec
2024-04-10 02:57:34,929 INFO: | Train Device=CSX, Step=700, Loss=7.84375, Rate=1531.06 samples/sec, GlobalRate=1531.73 samples/sec
2024-04-10 02:58:41,828 INFO: | Train Device=CSX, Step=800, Loss=7.77344, Rate=1530.83 samples/sec, GlobalRate=1531.60 samples/sec
2024-04-10 02:59:48,490 INFO: | Train Device=CSX, Step=900, Loss=7.67188, Rate=1533.99 samples/sec, GlobalRate=1532.10 samples/sec
2024-04-10 03:00:55,599 INFO: | Train Device=CSX, Step=1000, Loss=7.46875, Rate=1529.12 samples/sec, GlobalRate=1531.47 samples/sec
2024-04-10 03:00:55,600 INFO: Saving checkpoint at step 1000
2024-04-10 03:01:33,909 INFO: Saved checkpoint model_dir_bert_large_pytorch_512/checkpoint_1000.mdl
2024-04-10 03:02:52,525 INFO: Heartbeat thread stopped for wsjob-62bb2sg873frutuunbw3s.
2024-04-10 03:02:52,540 INFO: Training completed successfully!
2024-04-10 03:02:52,541 INFO: Processed 1024000 sample(s) in 668.637653252 seconds.
```

Graphcore

Homework

Run MNIST example by changing values of the input parameters like batch-size, learning rate and number of epochs trained and observe and report the performance implications.

Unfortunately, I've been trying forever. This job just is in queue forever:

```
L->torchvision==0.15.2->-r requirements.txt (line 1)) (3.13.4)
Requirement already satisfied: networkx in /home/clarexie/venvs/graphcore/poptorch33
L->torchvision==0.15.2->-r requirements.txt (line 1)) (3.1)
Requirement already satisfied: typing-extensions in /home/clarexie/venvs/graphcore/poptorch33
L->torch==2.0.1->torchvision==0.15.2->-r requirements.txt (line 1)) (4.11.0)
Requirement already satisfied: sympy in /home/clarexie/venvs/graphcore/poptorch33
L->torchvision==0.15.2->-r requirements.txt (line 1)) (1.12)
Requirement already satisfied: MarkupSafe>=2.0 in /home/clarexie/venvs/graphcore/poptorch33
L->torch==2.0.1->torchvision==0.15.2->-r requirements.txt (line 1)) (2.1.5)
Requirement already satisfied: mpmath>=0.19 in /home/clarexie/venvs/graphcore/poptorch33
L->torch==2.0.1->torchvision==0.15.2->-r requirements.txt (line 1)) (1.3.0)
Installing collected packages: tqdm
  Attempting uninstall: tqdm
    Found existing installation: tqdm 4.66.2
    Uninstalling tqdm-4.66.2:
      Successfully uninstalled tqdm-4.66.2
Successfully installed tqdm-4.46.1
(poptorch33_env) clarexie@gc-login-01:~/graphcore/examples/tutorials/simple_applications
(poptorch33_env) clarexie@gc-login-01:~/graphcore/examples/tutorials/simple_applications
(poptorch33_env) clarexie@gc-login-01:~/graphcore/examples/tutorials/simple_applications
python mnist_poptorch.py
srn: job 21014 queued and waiting for resources
```

Groq

Homework

Run BERT example with custom input instead of dummy input.

Notes: I found that there's some issue with requirements.txt that results in a bunch of extra work. They need to specify the pandas version to be 1.4.4 or something close to that because the required numpy version doesn't work with the most recent pandas version that requirements installs.

```

e of the follow@groq-r01-gn-08:~/groqflow/proof_points/natural_language_processing/bert$ python bert_tiny.py
tokenizer_config.json: 100%|          | 346/346 [00:00<00:00, 1.81MB/s]p
vocab.txt: 100%|          | 232k/232k [00:00<00:00, 18.0MB/s]
special_tokens_map.json: 100%|        | 112/112 [00:00<00:00, 1.25MB/s]
config.json: 100%|          | 760/760 [00:00<00:00, 8.19MB/s]
pytorch_model.bin: 100%|          | 17.6M/17.6M [00:00<00:00, 88.5MB/s]
/home/clarexie/miniconda3/envs/groqflow/lib/python3.10/site-packages/torch/_utils.py:831: UserWarning: TypedStorage is deprecated. It
will be removed in the future and UntypedStorage will be the only storage class. This should only matter to you if you are using stora
ges directly. To access UntypedStorage directly, use tensor.untyped_storage() instead of tensor.storage()
  return self.fget.__get__(instance, owner)()

Building "bert_tiny"
  ✓ Exporting PyTorch to ONNX
  ✓ Optimizing ONNX file
  ✓ Checking for Op support
  ✓ Converting to FP16
  ✓ Compiling model
  ✓ Assembling model

Woohoo! Saved to ~/.cache/groqflow/bert_tiny
Preprocessing data.
/home/clarexie/miniconda3/envs/groqflow/lib/python3.10/site-packages/datasets/load.py:1461: FutureWarning: The repository for sst cont
ains custom code which must be executed to correctly load the dataset. You can inspect the repository content at https://hf.co/dataset
s/sst
You can avoid this message in future by passing the argument 'trust_remote_code=True'.
Passing 'trust_remote_code=True' will be mandatory to load this dataset from the next major release of 'datasets'.
  warnings.warn(
Downloading builder script: 100%|          | 9.13k/9.13k [00:00<00:00, 49.8MB/s]
Downloading readme: 100%|          | 6.68k/6.68k [00:00<00:00, 34.2MB/s]
Downloading data: 100%|          | 6.37M/6.37M [00:01<00:00, 5.39MB/s]
Downloading data: 100%|          | 790k/790k [00:00<00:00, 1.85MB/s]
Generating train split: 100%|          | 8544/8544 [00:00<00:00, 11437.46 examples/s]
Generating validation split: 100%|        | 1101/1101 [00:00<00:00, 2010.44 examples/s]
Generating test split: 100%|          | 2210/2210 [00:00<00:00, 3951.02 examples/s]

Info: No inputs received for benchmark. Using the inputs provided during model compilation.
Running inference on GroqChip.
Running inference using PyTorch model (CPU).
100%|          | 2210/2210 [00:04<00:00, 457.12it/s]

+-----+-----+-----+-----+-----+-----+
| Source | Accuracy | end-to-end latency (ms) | end-to-end IPS | on-chip latency (ms) | on-chip IPS |
+-----+-----+-----+-----+-----+-----+
| cpu    | 77.47%   | 2.19                   | 457.05         | --                   | --          |
| groq   | 77.47%   | 0.06                   | 15690.22       | 0.03                 | 37576.72    |
+-----+-----+-----+-----+-----+-----+

Proof point /home/clarexie/groqflow/proof_points/natural_language_processing/bert/bert_tiny.py finished!
(groqflow) clarexie@groq-r01-gn-08:~/groqflow/proof_points/natural_language_processing/bert$

```

this is what I switched the inputs to be:

```

inputs = {
    "input_ids": torch.arange(max_seq_length, dtype=torch.long).expand(batch_size, max_seq_length),
    "attention_mask": torch.eye(batch_size, max_seq_length, dtype=torch.bool),
}

```

still pretty arbitrary, but at least this varies the index and move the mask.

Sambanova

Homework

For BERT example, understand flags used in the script. Change values for flag `--ntasks` and measure its effect on performance.

Answer:

using 16 task we only got an incremental speed up:

| | | | |
|--------------------------|-----|---------|--------|
| /NODE/XRDU_3/SW_0/PORT_8 | N/A | Present | Online |
| /NODE/HOST/HIC_0/DPORT | N/A | Present | Online |
| /NODE/HOST/HIC_1/DPORT | N/A | Present | Online |
| /NODE/HOST/HIC_2/DPORT | N/A | Present | Online |
| /NODE/HOST/HIC_3/DPORT | N/A | Present | Online |
| Duration: 790 | | | |

this is likely because of all the communication necessary

here is the sambnova log file path ~/ai-science-training-

series/07_AItestbeds/Sambanova/bert, we call them task8 and task16 respectively