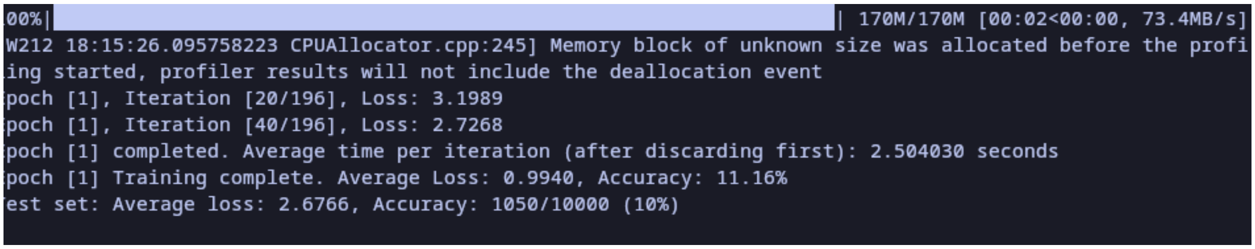
Runtimes -

Part 1: Metrics After Running the VGG11 for One Epoch (Batch Size: 256, 40 Iterations)

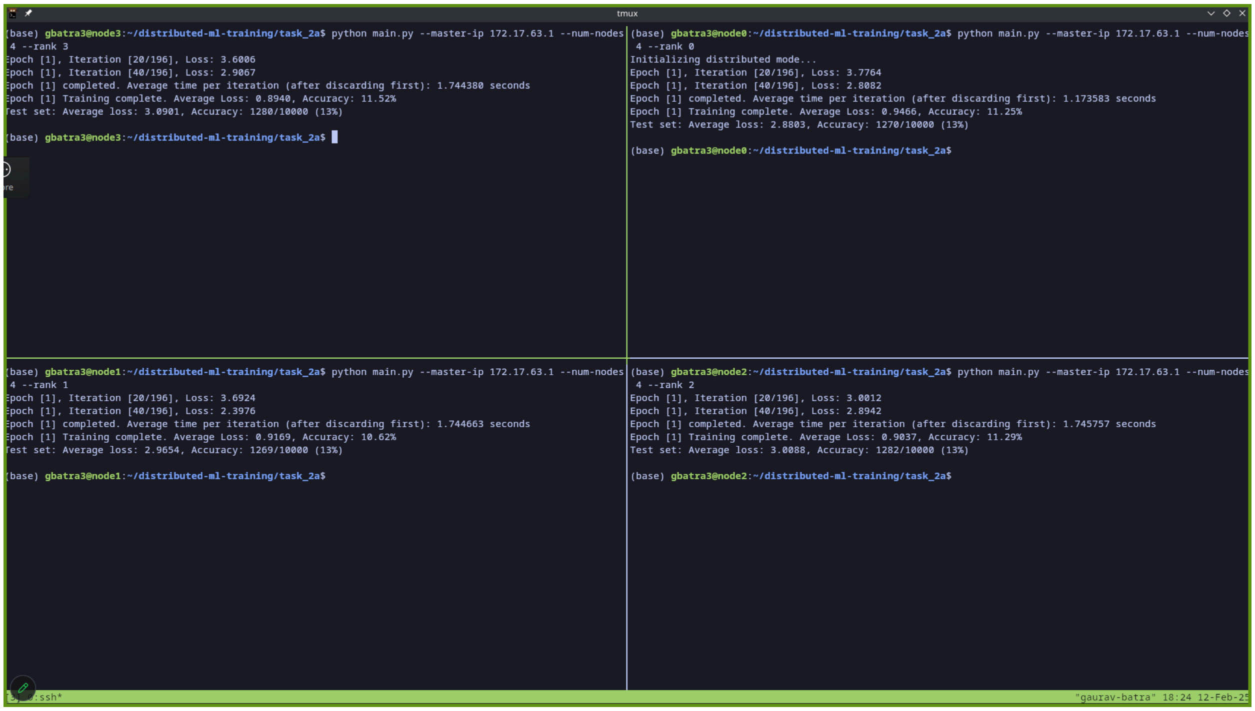


```
00%|████████████████████████████████████████| 170M/170M [00:02<00:00, 73.4MB/s]
W212 18:15:26.095758223 CPUAllocator.cpp:245] Memory block of unknown size was allocated before the profi
ing started, profiler results will not include the deallocation event
poch [1], Iteration [20/196], Loss: 3.1989
poch [1], Iteration [40/196], Loss: 2.7268
poch [1] completed. Average time per iteration (after discarding first): 2.504030 seconds
poch [1] Training complete. Average Loss: 0.9940, Accuracy: 11.16%
est set: Average loss: 2.6766, Accuracy: 1050/10000 (10%)
```

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.1989
1	40/196	2.7268

- Training Summary:
- Average Loss (Training): 0.9940
- Training Accuracy: 11.16%
- Test Accuracy: 10% (1050/10000)
- Average Time per Iteration (after discarding the first iteration): 2.504030 seconds

Part 2a: Metrics After Running the VGG11 for One Epoch (Total Batch Size: 256 (64*4), 40 Iterations)



```
(base) gbatra3@node3:~/distributed-ml-training/task_2a$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 3
poch [1], Iteration [20/196], Loss: 3.6006
poch [1], Iteration [40/196], Loss: 2.9067
poch [1] completed. Average time per iteration (after discarding first): 1.744380 seconds
poch [1] Training complete. Average Loss: 0.8940, Accuracy: 11.52%
est set: Average loss: 3.0901, Accuracy: 1280/10000 (13%)

(base) gbatra3@node3:~/distributed-ml-training/task_2a$ 
```

```
(base) gbatra3@node0:~/distributed-ml-training/task_2a$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 0
Initializing distributed mode...
Epoch [1], Iteration [20/196], Loss: 3.7764
Epoch [1], Iteration [40/196], Loss: 2.8082
Epoch [1] completed. Average time per iteration (after discarding first): 1.173583 seconds
Epoch [1] Training complete. Average Loss: 0.9466, Accuracy: 11.25%
Test set: Average loss: 2.8803, Accuracy: 1270/10000 (13%)

(base) gbatra3@node0:~/distributed-ml-training/task_2a$
```

```
(base) gbatra3@node1:~/distributed-ml-training/task_2a$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 1
poch [1], Iteration [20/196], Loss: 3.6924
poch [1], Iteration [40/196], Loss: 2.3976
poch [1] completed. Average time per iteration (after discarding first): 1.744663 seconds
poch [1] Training complete. Average Loss: 0.9169, Accuracy: 10.62%
est set: Average loss: 2.9654, Accuracy: 1269/10000 (13%)

(base) gbatra3@node1:~/distributed-ml-training/task_2a$
```

```
(base) gbatra3@node2:~/distributed-ml-training/task_2a$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 2
Epoch [1], Iteration [20/196], Loss: 3.0012
Epoch [1], Iteration [40/196], Loss: 2.8942
Epoch [1] completed. Average time per iteration (after discarding first): 1.745757 seconds
Epoch [1] Training complete. Average Loss: 0.9037, Accuracy: 11.29%
Test set: Average loss: 3.0088, Accuracy: 1282/10000 (13%)

(base) gbatra3@node2:~/distributed-ml-training/task_2a$
```

```
[3] 0:ssh*                                                    "gaurav-batra" 18:24 12-Feb-25
```

Node 0 (Rank 0)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.7764
1	40/196	2.8082

- Training Summary:

- Average Loss (Training): 0.9466
- Training Accuracy: 11.25%
- Test Accuracy: 13% (1270/10000)
- Average Time per Iteration (after discarding the first iteration): 1.173583 seconds

Node 1 (Rank 1)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.6924
1	40/196	2.3976

- Training Summary:

- Average Loss (Training): 0.9169
- Training Accuracy: 10.62%
- Test Accuracy: 13% (1269/10000)
- Average Time per Iteration (after discarding the first iteration): 1.744663 seconds

Node 2 (Rank 2)W

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.0012
1	40/196	2.8942

- Training Summary:

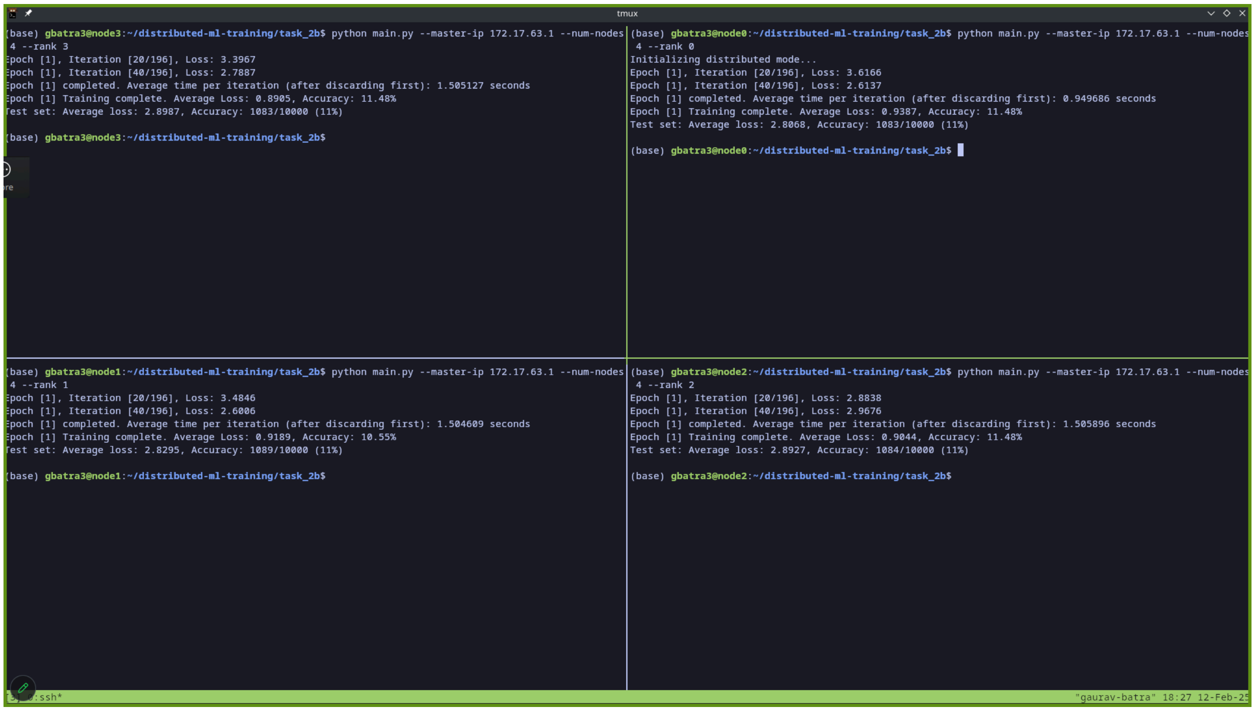
- Average Loss (Training): 0.9037
- Training Accuracy: 11.29%
- Test Accuracy: 13% (1282/10000)
- Average Time per Iteration (after discarding the first iteration): 1.745757 seconds

Node 3 (Rank 3)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.6006
1	40/196	2.9067

- Training Summary:
- Average Loss (Training): 0.8940
- Training Accuracy: 11.52%
- Test Accuracy: 13% (1280/10000)
- Average Time per Iteration (after discarding the first iteration): 1.744380 seconds

Part 2b: Metrics After Running the VGG11 for One Epoch (Total Batch Size: 256 (64*4), 40 Iterations)



```
(base) gbatra3@node3:~/distributed-ml-training/task_2b$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 3
Epoch [1], Iteration [20/196], Loss: 3.3967
Epoch [1], Iteration [40/196], Loss: 2.7887
Epoch [1] completed. Average time per iteration (after discarding first): 1.505127 seconds
Epoch [1] Training complete. Average Loss: 0.8905, Accuracy: 11.48%
Test set: Average loss: 2.8987, Accuracy: 1083/10000 (11%)

(base) gbatra3@node3:~/distributed-ml-training/task_2b$
```

```
(base) gbatra3@node0:~/distributed-ml-training/task_2b$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 0
Initializing distributed mode...
Epoch [1], Iteration [20/196], Loss: 3.6166
Epoch [1], Iteration [40/196], Loss: 2.6137
Epoch [1] completed. Average time per iteration (after discarding first): 0.949686 seconds
Epoch [1] Training complete. Average Loss: 0.9387, Accuracy: 11.48%
Test set: Average loss: 2.8068, Accuracy: 1083/10000 (11%)

(base) gbatra3@node0:~/distributed-ml-training/task_2b$ 
```

```
(base) gbatra3@node1:~/distributed-ml-training/task_2b$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 1
Epoch [1], Iteration [20/196], Loss: 3.4846
Epoch [1], Iteration [40/196], Loss: 2.6006
Epoch [1] completed. Average time per iteration (after discarding first): 1.504609 seconds
Epoch [1] Training complete. Average Loss: 0.9189, Accuracy: 10.55%
Test set: Average loss: 2.8295, Accuracy: 1089/10000 (11%)

(base) gbatra3@node1:~/distributed-ml-training/task_2b$
```

```
(base) gbatra3@node2:~/distributed-ml-training/task_2b$ python main.py --master-ip 172.17.63.1 --num-nodes
 4 --rank 2
Epoch [1], Iteration [20/196], Loss: 2.8838
Epoch [1], Iteration [40/196], Loss: 2.9676
Epoch [1] completed. Average time per iteration (after discarding first): 1.505896 seconds
Epoch [1] Training complete. Average Loss: 0.9044, Accuracy: 11.48%
Test set: Average loss: 2.8927, Accuracy: 1084/10000 (11%)

(base) gbatra3@node2:~/distributed-ml-training/task_2b$
```

```
[3] 0:ssh*                                                              "gaurav-batra" 18:27 12-Feb-25
```

Node 0 (Rank 0)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.6166
1	40/196	2.6137

- Training Summary:
- Average Loss (Training): 0.9387
- Training Accuracy: 11.48%
- Test Accuracy: 11% (1083/10000)
- Average Time per Iteration (after discarding the first iteration): 0.949686 seconds

Node 1 (Rank 1)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.4846
1	40/196	2.6006

- Training Summary:
- Average Loss (Training): 0.9189
- Training Accuracy: 10.55%
- Test Accuracy: 11% (1089/10000)
- Average Time per Iteration (after discarding the first iteration): 1.504609 seconds

Node 2 (Rank 2)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	2.8838
1	40/196	2.9676

- Training Summary:
- Average Loss (Training): 0.9044
- Training Accuracy: 11.48%
- Test Accuracy: 11% (1084/10000)
- Average Time per Iteration (after discarding the first iteration): 1.505896 seconds

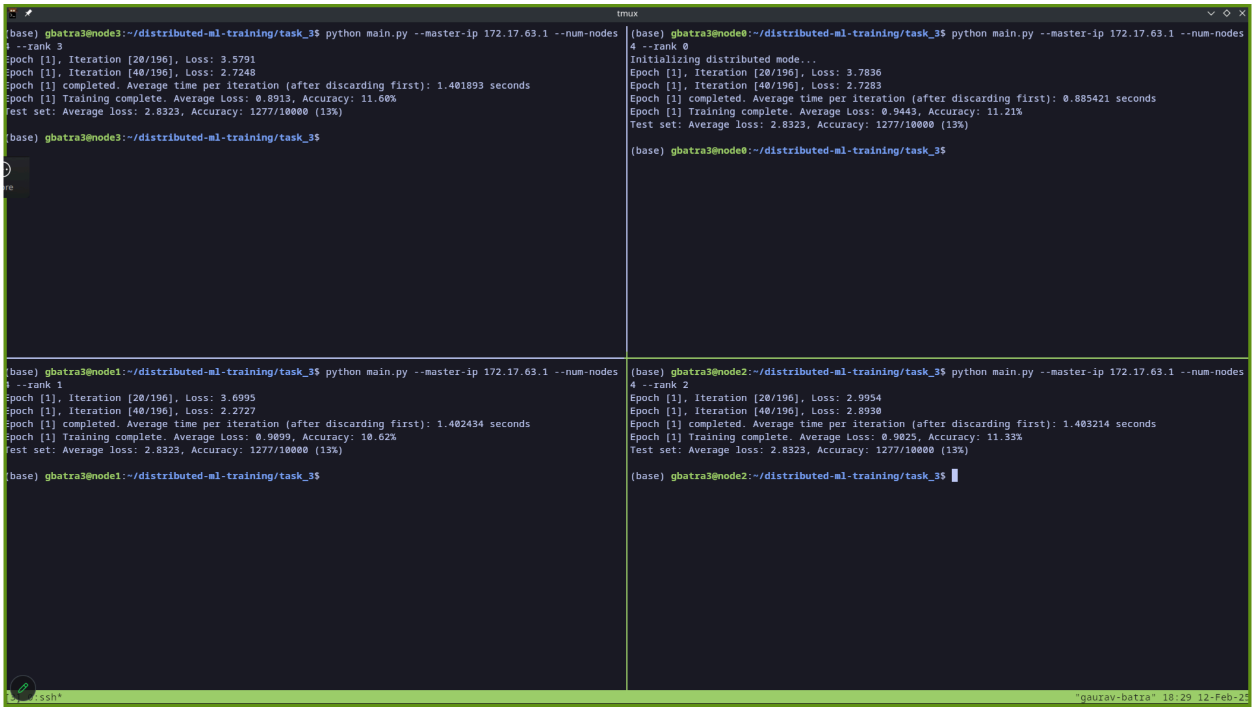
Node 3 (Rank 3)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.3967
1	40/196	2.7887

- Training Summary:

- Average Loss (Training): 0.8905
- Training Accuracy: 11.48%
- Test Accuracy: 11% (1083/10000)
- Average Time per Iteration (after discarding the first iteration): 1.505127 seconds

Part 3: Metrics After Running the VGG11 for One Epoch (Total Batch Size: 256 (64*4), 40 Iterations)



```
(base) gbatra3@node3:~/distributed-ml-training/task_3$ python main.py --master-ip 172.17.63.1 --num-nodes
4 --rank 3
Epoch [1], Iteration [20/196], Loss: 3.5791
Epoch [1], Iteration [40/196], Loss: 2.7248
Epoch [1] completed. Average time per iteration (after discarding first): 1.401893 seconds
Epoch [1] Training complete. Average Loss: 0.8913, Accuracy: 11.60%
Test set: Average loss: 2.8323, Accuracy: 1277/10000 (13%)

(base) gbatra3@node3:~/distributed-ml-training/task_3$
```

```
(base) gbatra3@node0:~/distributed-ml-training/task_3$ python main.py --master-ip 172.17.63.1 --num-nodes
4 --rank 0
Initializing distributed mode...
Epoch [1], Iteration [20/196], Loss: 3.7836
Epoch [1], Iteration [40/196], Loss: 2.7283
Epoch [1] completed. Average time per iteration (after discarding first): 0.885421 seconds
Epoch [1] Training complete. Average Loss: 0.9443, Accuracy: 11.21%
Test set: Average loss: 2.8323, Accuracy: 1277/10000 (13%)

(base) gbatra3@node0:~/distributed-ml-training/task_3$
```

```
(base) gbatra3@node1:~/distributed-ml-training/task_3$ python main.py --master-ip 172.17.63.1 --num-nodes
4 --rank 1
Epoch [1], Iteration [20/196], Loss: 3.6995
Epoch [1], Iteration [40/196], Loss: 2.2727
Epoch [1] completed. Average time per iteration (after discarding first): 1.402434 seconds
Epoch [1] Training complete. Average Loss: 0.9099, Accuracy: 10.62%
Test set: Average loss: 2.8323, Accuracy: 1277/10000 (13%)

(base) gbatra3@node1:~/distributed-ml-training/task_3$
```

```
(base) gbatra3@node2:~/distributed-ml-training/task_3$ python main.py --master-ip 172.17.63.1 --num-nodes
4 --rank 2
Epoch [1], Iteration [20/196], Loss: 2.9954
Epoch [1], Iteration [40/196], Loss: 2.8930
Epoch [1] completed. Average time per iteration (after discarding first): 1.403214 seconds
Epoch [1] Training complete. Average Loss: 0.9025, Accuracy: 11.33%
Test set: Average loss: 2.8323, Accuracy: 1277/10000 (13%)

(base) gbatra3@node2:~/distributed-ml-training/task_3$
```

```
[3] 0:ssh*                                                                 "gaurav-batra" 18:29 12-Feb-25
```

Node 0 (Rank 0)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.7836
1	40/196	2.7283

- Training Summary:

- Average Loss (Training): 0.9443
- Training Accuracy: 11.21%
- Test Accuracy: 13% (1277/10000)
- Average Time per Iteration (after discarding the first iteration): 0.885421 seconds

Node 1 (Rank 1)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.6995
1	40/196	2.2727

- Training Summary:

- Average Loss (Training): 0.9099
- Training Accuracy: 10.62%
- Test Accuracy: 13% (1277/10000)
- Average Time per Iteration (after discarding the first iteration): 1.402434 seconds

Node 2 (Rank 2)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	2.9954
1	40/196	2.8930

- Training Summary:

- Average Loss (Training): 0.9025
- Training Accuracy: 11.33%
- Test Accuracy: 13% (1277/10000)
- Average Time per Iteration (after discarding the first iteration): 1.403214 seconds

Node 3 (Rank 3)

Epoch	Iteration	Loss (Cross-Entropy)
1	20/196	3.5791
1	40/196	2.7248

- Training Summary:

- Average Loss (Training): 0.8913
- Training Accuracy: 11.60%
- Test Accuracy: 13% (1277/10000)
- Average Time per Iteration (after discarding the first iteration): 1.401893 seconds

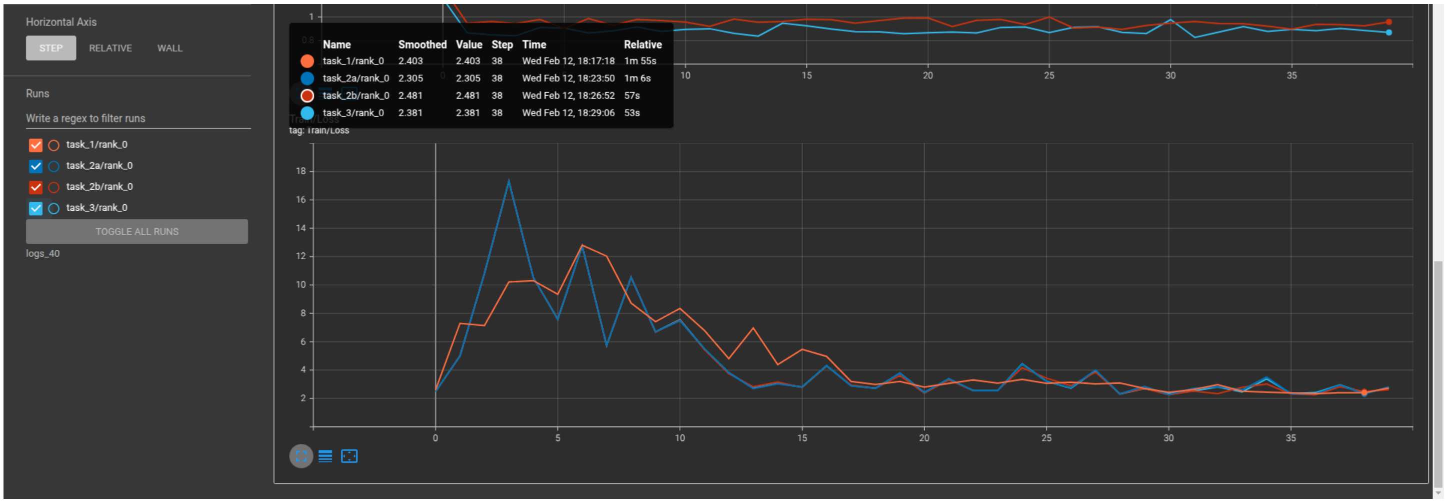
Comparison of different setups -

- Part 1 is purely a single node implementation of the NN where the entire model with all of its data is trained on a single worker node
- From Part 2 onwards, we do distributed processing (data parallelism) and training of the NN by dividing data among worker nodes
- In Part 2a, the data is divided on the 4 worker nodes which individually train on a mini-batch of data (64 samples) and gradient synchronization is done manually using a gather and scatter protocol on Rank 0 node
- In Part 2b, we follow the same data distribution and training as in Part 2a with the only difference being the AllReduce protocol for gradient synchronization, which distributes communication costs amongst all worker nodes
- Part 3 leverages the DistributedDataParallel provided by Pytorch to achieve Data Parallel training of the model (as described in the paper). It internally uses bucketing and other clever optimizations to do gradient synchronization and achieve parallel communication and computation

Scalability -

- Part 1 (Single-node training)
  - This approach is expected to be the slowest due to obvious reasons as the whole computation is bottlenecked by the compute limitations of a single GPU and it’s memory.
  - As expected this was the slowest and also the least scalable as it does not leverage multiple nodes for scaling
- Part 2a (Gradient synchronization with gather-scatter)
  - While this approach is better than single node implementation, there is still a scalability issue due to communication overhead at rank 0 worker
  - Each worker sends its gradients to rank 0 (gather), which averages them and sends them back (scatter).
  - This introduces communication overhead at rank 0, making it potentially inefficient at scale.
- Part 2b (Gradient synchronization with allreduce)
  - This approach is more scalable as compared to gather and scatter since the communication load is distributed amongst all the workers
  - Instead of gathering at rank 0, all nodes participate in averaging gradients in a ring-reduce fashion
  - Hence as expected this approach is faster and more scalable than 2a, especially as world size increases
- Part 3 (DDP)
  - This is the most optimized and scalable approach since it implements an optimized allreduce and overlaps computation with communication
  - It also uses bucketization to minimize synchronization overhead (as mentioned in the paper)

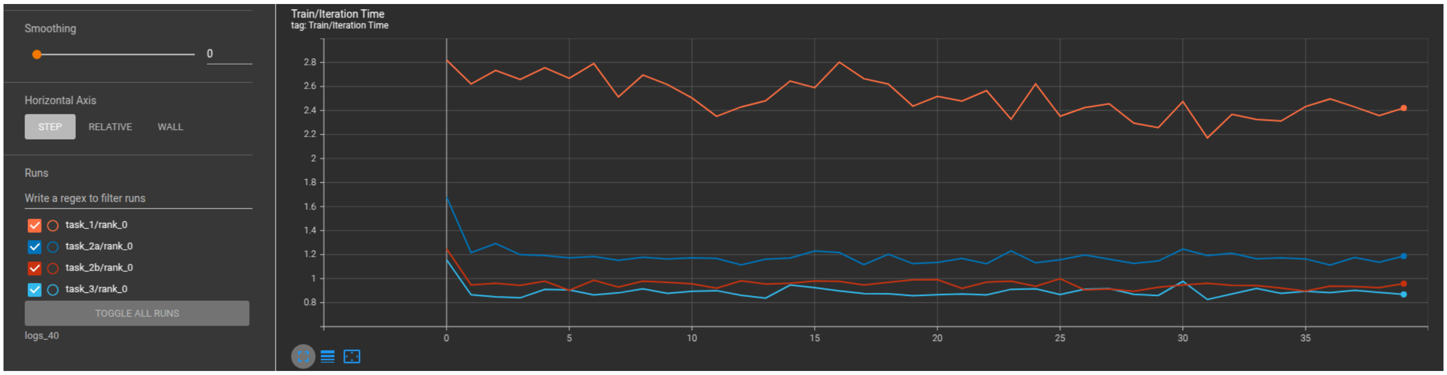
Training Loss vs Iterations for all the implementations:



	Name	Smoothed	Value	Step	Time	Relative
●	task_1/rank_0	2.403	2.403	38	Wed Feb 12, 18:17:18	1m 55s
●	task_2a/rank_0	2.305	2.305	38	Wed Feb 12, 18:23:50	1m 6s
●	task_2b/rank_0	2.481	2.481	38	Wed Feb 12, 18:26:52	57s
●	task_3/rank_0	2.381	2.381	38	Wed Feb 12, 18:29:06	53s

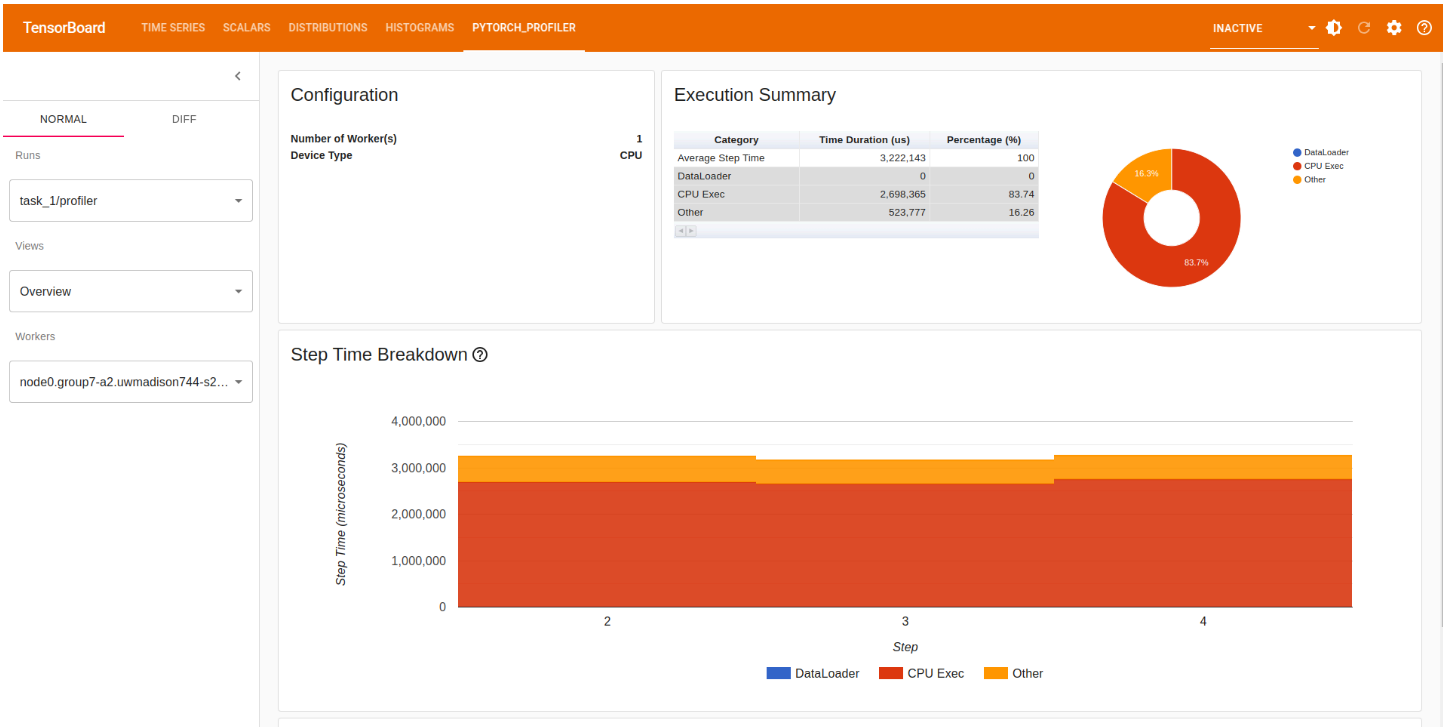
- As expected the training loss decreases with the number of iterations.

Time per iteration vs Iterations for all the implementations



- The time per iteration is highest for Part 1 (single node) followed by Part 2a, 2b and lowest for Part 3 (as explained before)
- The time per iteration in distributed modes is comparable as we are running for 40 iterations on a small model. With larger model size and data volumes the difference within these setups will be significant.

Part 1: Single CPU training:



Runs: task_1/profiler

Views: Overview

Workers: node0.group7-a2.uwmadison744-s2...

Configuration

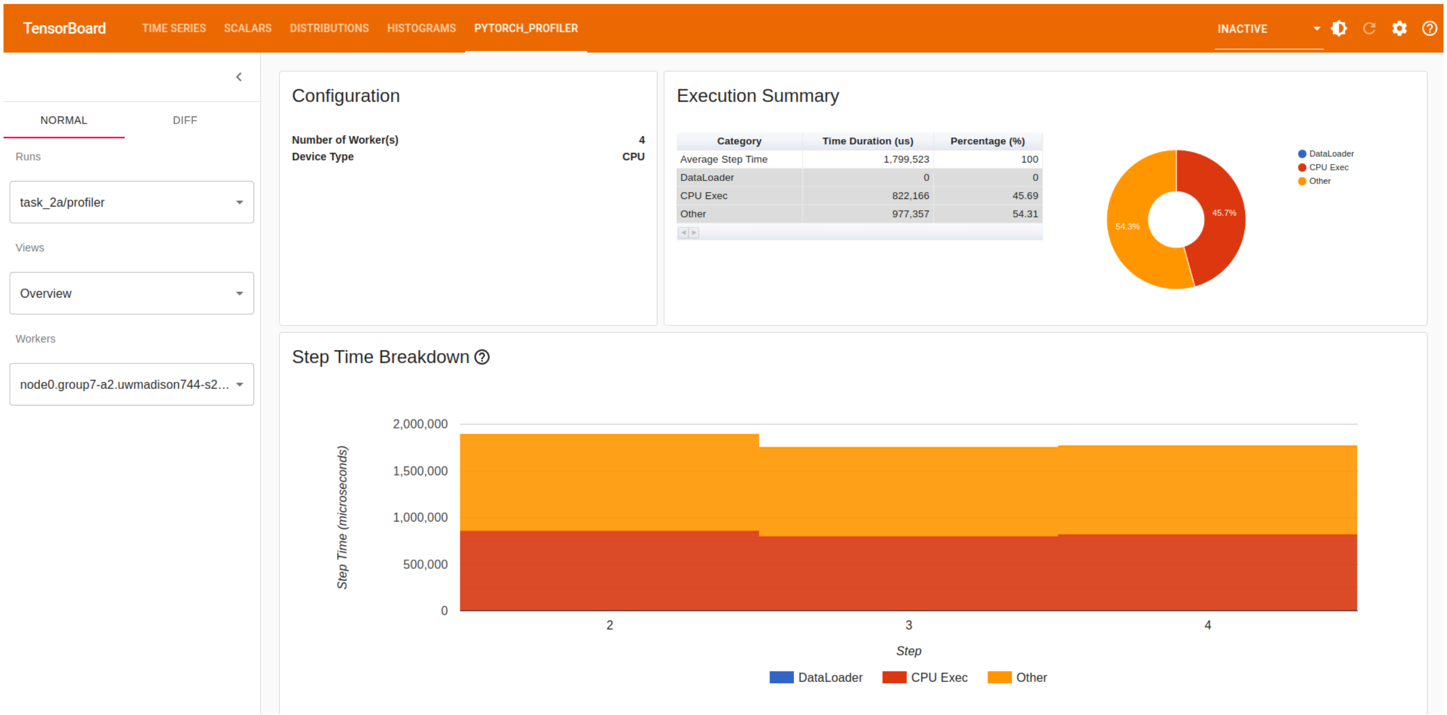
Number of Worker(s) **1**
Device Type **CPU**

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	3,222,143	100
DataLoader	0	0
CPU Exec	2,698,365	83.74
Other	523,777	16.26

- In this implementation majority of the time is spent in computation as it’s a single CPU (no communication)
- However, the average time step is higher as compared to other implementations.

Part 2a: Multi CPU training with Gloo as backend using gather and scatter primitives



Configuration

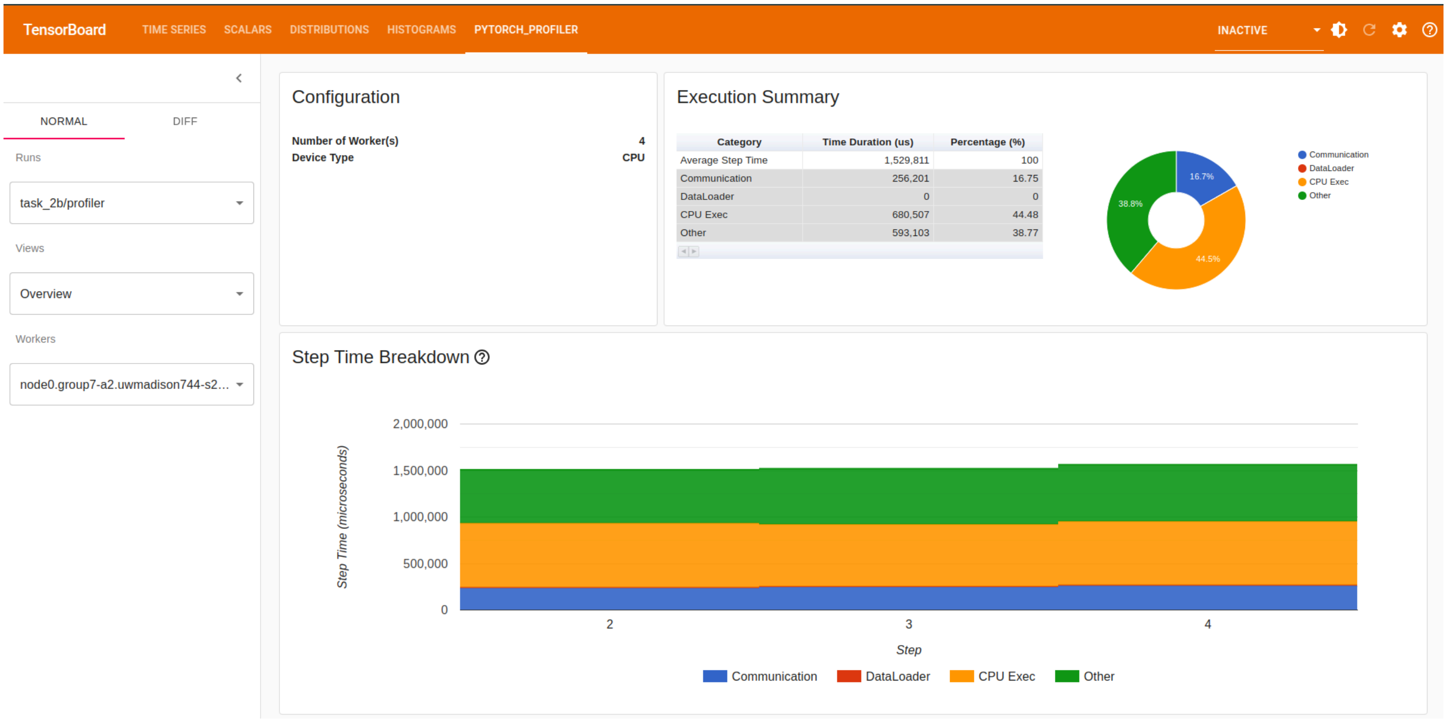
Number of Worker(s)	**4**
Device Type	**CPU**

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	1,799,523	100
DataLoader	0	0
CPU Exec	822,166	45.69
Other	977,357	54.31

- In this implementation, time spent in computation decreases (~85% -> 46%) as now there is additional overhead of communication between the CPUs.
- However, the average time step is lower as compared to single CPU training.

Part 2b: Multi CPU training with Gloo as backend using all reduce



Runs: task_2b/profiler

Views: Overview

Workers: node0.group7-a2.uwmadison744-s2…

Configuration

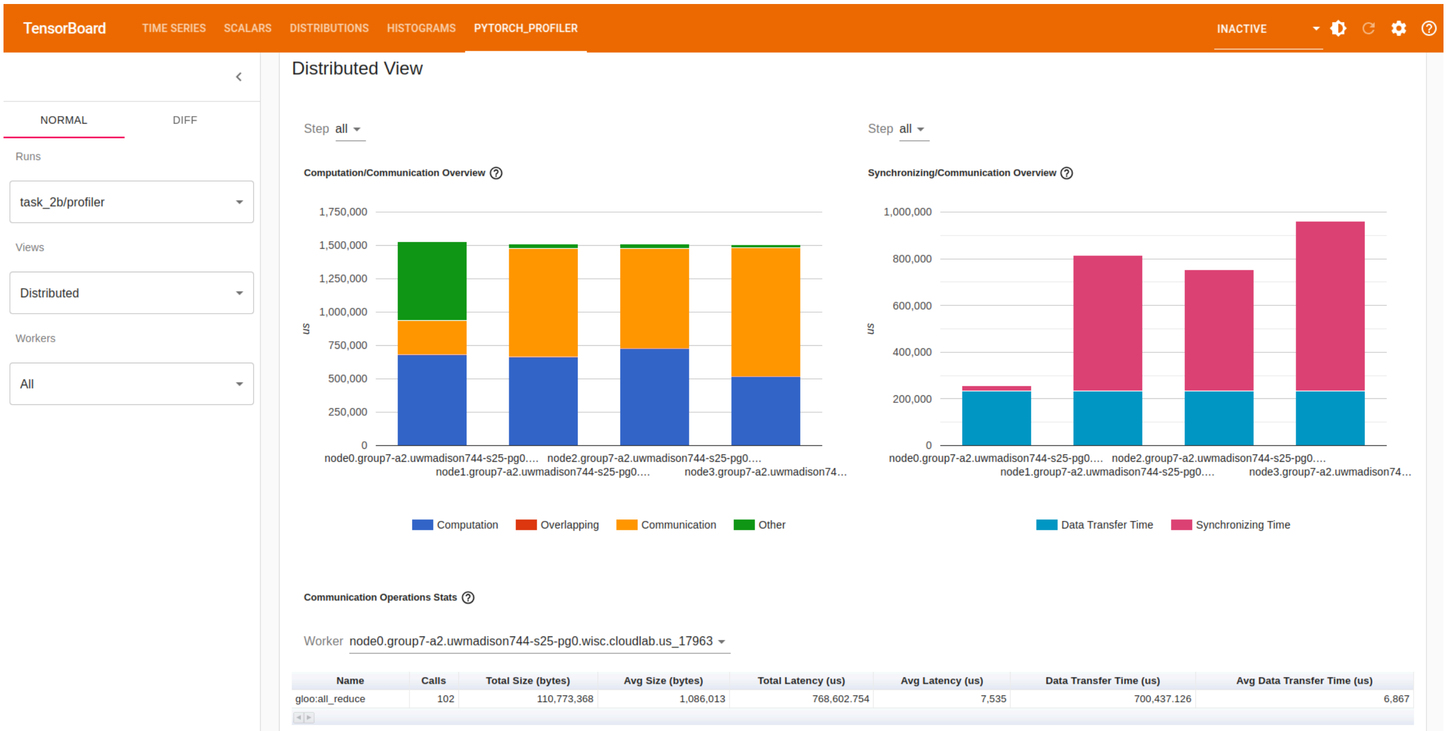
Number of Worker(s) 4

Device Type CPU

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	1,529,811	100
Communication	256,201	16.75
DataLoader	0	0
CPU Exec	680,507	44.48
Other	593,103	38.77

Step Time Breakdown


TensorBoard
TIME SERIES
SCALARS
DISTRIBUTIONS
HISTOGRAMS
PYTORCH_PROFILER
INACTIVE
NORMAL
DIFF

Runs: task_2b/profiler

Views: Distributed

Workers: All

Distributed View

Step all

Computation/Communication Overview

1,750,000
1,500,000
1,250,000
1,000,000
750,000
500,000
250,000
0
us
node0.group7-a2.uwmadison744-s25-pg0.…
node1.group7-a2.uwmadison744-s25-pg0.…
node2.group7-a2.uwmadison744-s25-pg0.…
node3.group7-a2.uwmadison74…
Computation
Overlapping
Communication
Other

Step all

Synchronizing/Communication Overview

1,000,000
800,000
600,000
400,000
200,000
0
us
node0.group7-a2.uwmadison744-s25-pg0.…
node1.group7-a2.uwmadison744-s25-pg0.…
node2.group7-a2.uwmadison744-s25-pg0.…
node3.group7-a2.uwmadison74…
Data Transfer Time
Synchronizing Time

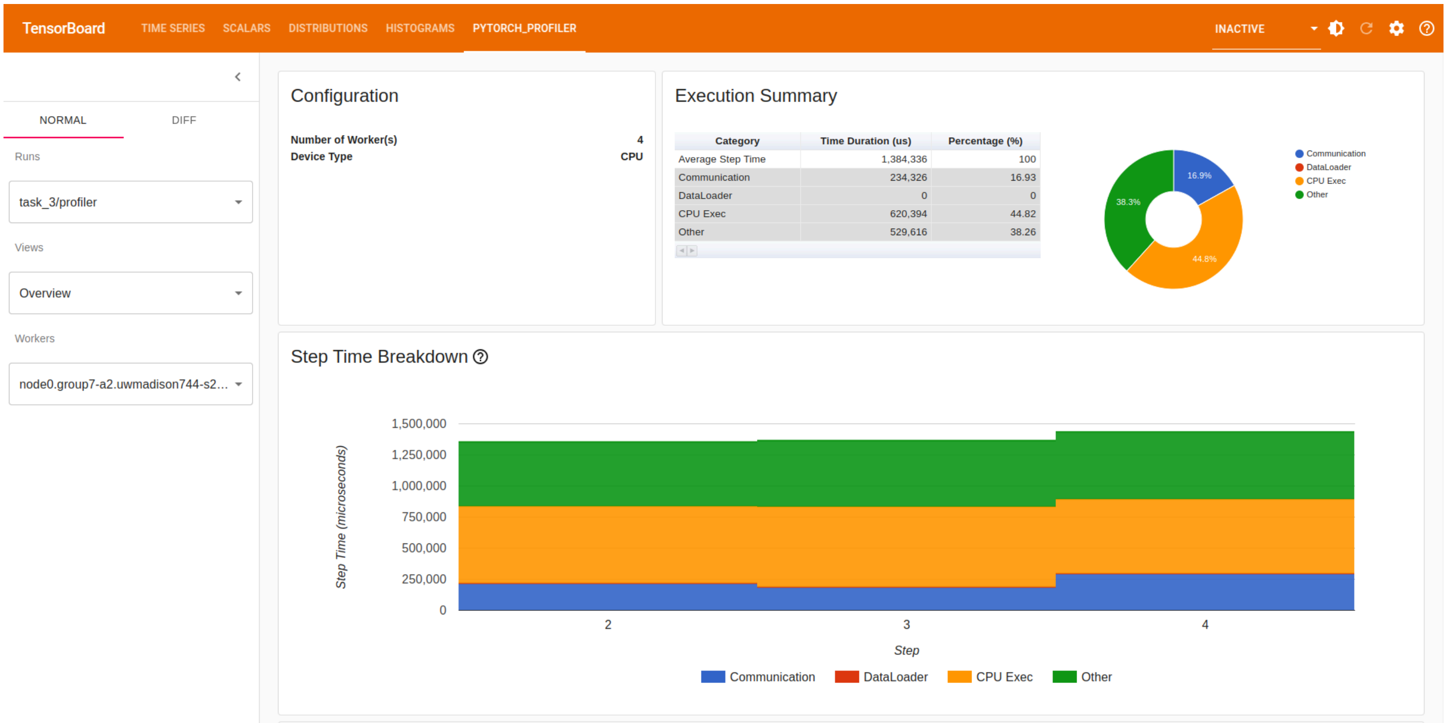
Communication Operations Stats

Worker node0.group7-a2.uwmadison744-s25-pg0.wisc.cloudlab.us_17963

Name	Calls	Total Size (bytes)	Avg Size (bytes)	Total Latency (us)	Avg Latency (us)	Data Transfer Time (us)	Avg Data Transfer Time (us)
gloo:all_reduce	102	110,773,368	1,086,013	768,602.754	7,535	700,437.126	6,867

- In this implementation, the time spent in computation is similar to the previous implementation (part 2a). [~45% for computation and ~17% for communication]
- The average time step is also lower to the previous implementation (part 2a).
- There is no overlapping of communication/computation between the nodes.

Part 3: Multi CPU training with Gloo as backend using Pytorch's Distributed Data-Parallel Module

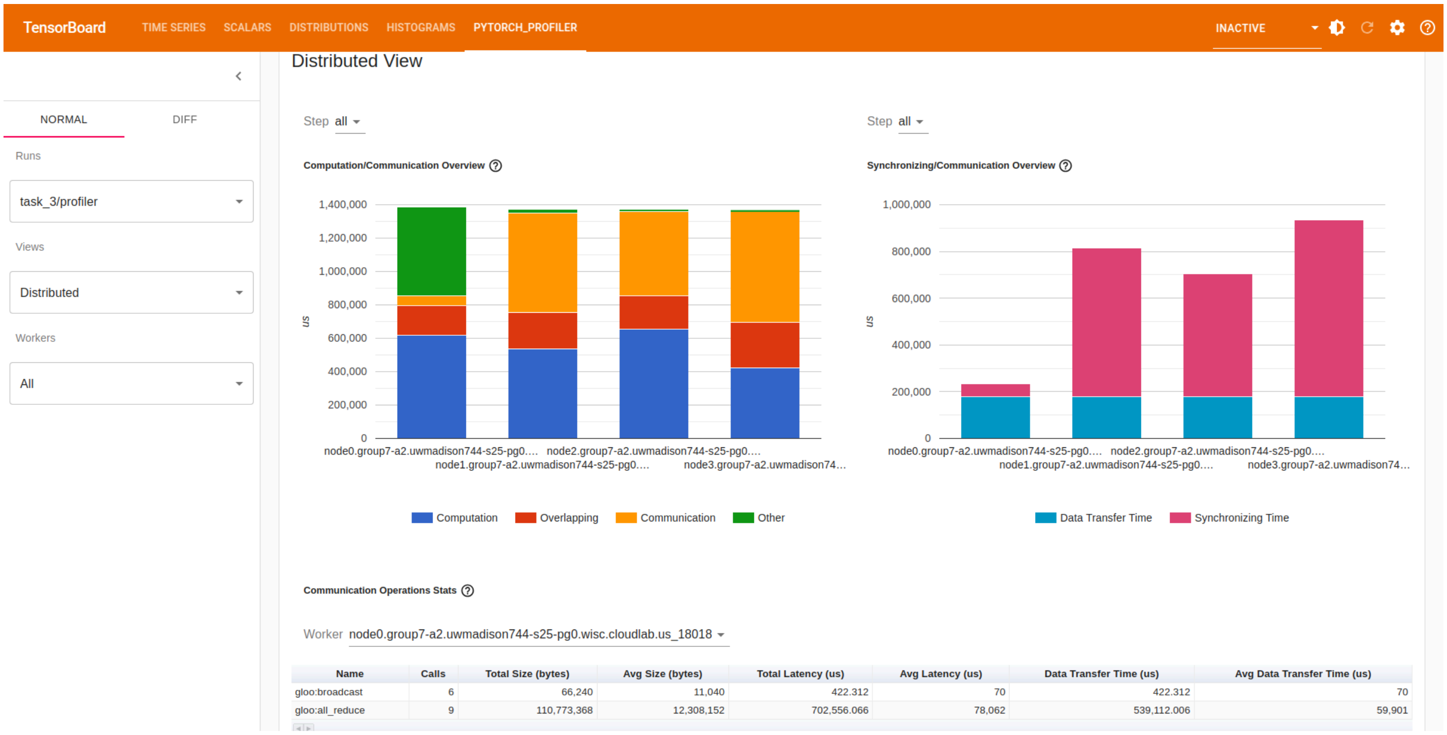


Configuration

Number of Worker(s)	**4**
Device Type	**CPU**

Execution Summary

Category	Time Duration (us)	Percentage (%)
Average Step Time	1,384,336	100
Communication	234,326	16.93
DataLoader	0	0
CPU Exec	620,394	44.82
Other	529,616	38.26


TensorBoard
TIME SERIES
SCALARS
DISTRIBUTIONS
HISTOGRAMS
PYTORCH_PROFILER
INACTIVE
NORMAL
DIFF
Runs
task_3/profiler
Views
Distributed
Workers
All
Distributed View
Step all
Computation/Communication Overview
1,400,000
1,200,000
1,000,000
800,000
600,000
400,000
200,000
0
us
node0.group7-a2.uwmadison744-s25-pg0.…
node1.group7-a2.uwmadison744-s25-pg0.…
node2.group7-a2.uwmadison744-s25-pg0.…
node3.group7-a2.uwmadison74…
Computation
Overlapping
Communication
Other
Step all
Synchronizing/Communication Overview
1,000,000
800,000
600,000
400,000
200,000
0
us
node0.group7-a2.uwmadison744-s25-pg0.…
node1.group7-a2.uwmadison744-s25-pg0.…
node2.group7-a2.uwmadison744-s25-pg0.…
node3.group7-a2.uwmadison74…
Data Transfer Time
Synchronizing Time

Communication Operations Stats

Worker node0.group7-a2.uwmadison744-s25-pg0.wisc.cloudlab.us_18018

Name	Calls	Total Size (bytes)	Avg Size (bytes)	Total Latency (us)	Avg Latency (us)	Data Transfer Time (us)	Avg Data Transfer Time (us)
gloo:broadcast	6	66,240	11,040	422.312	70	422.312	70
gloo:all_reduce	9	110,773,368	12,308,152	702,556.066	78,062	539,112.006	59,901

- In this implementation, the time spent in computation is similar to the previous implementations (part 2a and part 2b). [~45% for computation and ~17% for communication]
- The average time step is lower than the previous implementations (part 2a and part 2b).
- There is an overlapping of communication/computation between the nodes.

Contribution

- Team Members - Chirag Jain, Sarthak Khattar, Gaurav Batra
- Each one of us has worked on the assignment equally. We sat together and set up the whole cluster and performed all parts together
- The codes for the different parts were reasoned and implemented together and the results as presented in this document were collectively discussed and documented