

Assignment II - Distributed Data Parallel Training

Group V

Badri Narayanan Murali Krishnan

Shreyas Subramanian

Rohan Purandhar

FNU Nevin John Selby

`bmuralikrish@wisc.edu, ss7@wisc.edu`

`purandhar@wisc.edu, nselby@wisc.edu`

Github Link: <https://github.com/MBadriNarayanan/CS744>

February 23, 2024

1 Part One: Training VGG-11 on CIFAR-10

1.1 Solution

We finalized the foundational training scripts. We accomplished the initial training process involving forward and backward passes, loss calculation, and optimizer setup in the *main.py* file. Additionally, we logged the loss of every 20 batch iteration. After running 40 mini-batch iterations and disregarding the initial one, we computed the average runtime which was found to be 1.477 seconds. The training loss was 4.652, the test loss was 3.529, and the test accuracy was 13.1%.

```
(base) cs744@dd5a98e55317:/CS744/AssignmentTwo$ python3 PartOne/main.py
-----
Starting Part One!

GPU not available!
Downloaded https://www.cs.toronto.edu/~kriz/cifar-10-python.tar.gz to ./data/cifar-10-python.tar.gz
100%[=====] 100.00M 178498071/178498071 [00:01<00:00, 98136199.27it/s]
Extracting ./data/cifar-10-python.tar.gz to ./data/cifar-10-python
Files already downloaded and verified
Created directory: Checkpoints!
Created directory: Checkpoints/PartOne!
Created directory: Logs!
Created directory: Logs/PartOne!
Checkpoints will be stored at: Checkpoints/PartOne!
Logs/PartOne will be stored at: Logs/PartOne/Logs.txt
Evaluation report will be stored at: Logs/PartOne/report.txt
Preparing model for training!
Parameters: 923114
Batch Idx: 0, Batch Loss: 2.531, Batch Accuracy: 0.195, Batch Duration: 1.893 seconds
Batch Idx: 20, Batch Loss: 3.867, Batch Accuracy: 0.117, Batch Duration: 1.452 seconds
Batch Idx: 48, Batch Loss: 2.642, Batch Accuracy: 0.133, Batch Duration: 1.455 seconds
-----
Successfully trained the model for 48 batches!
Epoch: 1, Train Loss: 4.652, Train Accuracy: 0.117, Avg Batch Duration: 1.477 seconds, Epoch Duration: 61.210 seconds
Avg Epoch Duration: 61.210 seconds
-----
Loaded checkpoint: Checkpoints/PartOne/EPOCH_1.pt for evaluation!
Metrics for the checkpoint: Checkpoints/PartOne/EPOCH_1.pt
Test Loss: 3.529, Test Accuracy: 0.131
Classification Report
precision    recall   f1-score   support
      0       0.000     0.000     0.000     1000
      1       0.167     0.001     0.002     1000
      2       0.000     0.000     0.000     1000
      3       0.000     0.000     0.000     1000
      4       0.000     0.000     0.000     1000
      5       0.000     0.000     0.000     1000
      6       0.016     0.005     0.008     1000
      7       0.115     0.924     0.264     1000
      8       0.000     0.251     0.274     1000
      9       0.000     0.000     0.000     1000
accuracy         0.053     0.131     0.056     10000
macro avg       0.053     0.131     0.056     10000
weighted avg    0.053     0.131     0.056     10000
Confusion Matrix
[[  0   4   0   0   0   0   48 427 521   0]
 [  0   1   0   0   0   0   36 821 142   0]
 [  0   1   0   0   0   0   16 833 158   0]
 [  0   0   0   0   0   0   10 100 143   0]
 [  0   0   0   0   0   0   12 742 46   0]
 [  0   0   0   0   0   0   20 898 85   0]
 [  0   0   0   0   0   0   15 978 25   0]
 [  0   0   0   0   0   0   19 100 100   0]
 [  0   0   0   0   0   0   82 537 381   0]
 [  0   0   0   0   0   0   54 814 132   0]]
-----
Part One complete!
-----
(base) cs744@dd5a98e55317:/CS744/AssignmentTwo$
```

Figure 1: Output snapshots for Part One.

2 Part Two: Distributed Data Parallel Training

2.1 Part 2a: Sync gradient with gather and scatter call using Gloo backend

2.1.1 Solution

The training process was completed successfully, producing the expected results. The final average train loss value was 5.784, the average test loss 2.753, while the test accuracy was 11.9% - slightly different from part 1 but still within the anticipated range. Table 1 denotes the avg time it took to run 40 mini batch iterations and the epoch duration across the different nodes with a batch size of 64. From the table, we see that the average batch duration was 1.55 seconds with an average epoch duration 47.894.

Table 1: Metrics across the different nodes for Part Two Task One.

Node	Avg Batch Duration	Epoch Duration
Master Node	1.155	47.894
Node 1	1.155	47.893
Node 2	1.155	47.892
Node 3	1.155	47.896
Avg	1.155	47.894

```
(base) cs744@dd5a90e55317:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 0
-----
Starting Part Two Task Two!
-----
Running Rank: 0!
GPU not available!
Files already uploaded and verified
Files already downloaded and verified
Directory: Checkpoints already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Creating checkpoints stored at Checkpoints/PartTwo/TaskTwo/
Training log will be stored at Logs/PartTwo/TaskTwo/logs/rank0.txt!
Evaluation report will be stored at Logs/PartTwo/TaskTwo/report_rank0.txt!
Prepare model for training!
Parameters: 9231114
-----
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.121 seconds
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.966 seconds
Batch Idx: 40, Batch Loss: 2.443, Batch Accuracy: 0.031, Batch Duration: 0.963 seconds
Successfully trained the model for 40 batches!
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.065 seconds
Avg Epoch Duration: 40.065 seconds
-----
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/Epoch_1.pt
Test Loss: 2.763, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
0            0.120    0.721    0.212    1000
1            0.064    0.849    0.056    1000
2            0.000    0.000    0.000    1000
3            0.000    0.000    0.000    1000
4            0.000    0.000    0.000    1000
5            0.153    0.214    0.178    1000
6            0.000    0.000    0.000    1000
7            0.000    0.000    0.000    1000
8            0.053    0.000    0.014    1000
9            0.000    0.000    0.000    1000
accuracy          0.119    10000
macro avg       0.039    0.119    0.046    10000
weighted avg    0.039    0.119    0.046    10000
Confusion Matrix
[[921  0  0  0  0  49  0  0  0  0]
 [854  49  0  0  0  87  0  0  10  0]
 [748  94  0  0  0  145  0  0  13  0]
 [675  106  0  0  0  195  0  0  24  0]
 [68  112  0  0  0  214  0  0  51  0]
 [1443  2  0  0  0  255  0  0  23  0]
 [1682  120  0  0  0  255  0  0  23  0]
 [731  86  0  0  0  171  0  0  12  0]
 [1940  20  0  0  0  32  0  0  8  0]
 [868  51  0  0  0  66  0  0  15  0]]
-----
Part Two Task Two complete!
-----
(base) cs744@dd5a90e55317:/CS744/AssignmentTwo$
```

Figure 2: Output snapshots on Master Node for Part Two Task One.

```
(base) cs744@clifbee@clcs53:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 1
-----
Starting Part Two Task Two!
-----
Running Rank: 1
GPU not available!
Files already downloaded and verified
Files already downloaded and verified
Directory: Checkpoints/PartTwo already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Logs already exists!
Directory: Logs/PartTwo already exists!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Checkpoints will be stored at: Checkpoints/PartTwo/TaskTwo!
Training logs will be stored at: Logs/PartTwo/TaskTwo/logs_rank1.txt!
Evaluation report will be stored at: Logs/PartTwo/TaskTwo/report_rank1.txt!
Prepared model for training!
Parameters: 923114
-----
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.096 seconds
-----
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.958 seconds
-----
Batch Idx: 48, Batch Loss: 2.461, Batch Accuracy: 0.031, Batch Duration: 0.968 seconds
-----
Successfully trained the model for 48 batches!
-----
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.066 seconds
Avg Epoch Duration: 40.066 seconds
-----
```



```
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/Epoch_1.pt
Test Loss: 2.753, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
      0       0.120     0.721     0.212     1000
      1       0.064     0.049     0.056     1000
      2       0.080     0.080     0.080     1000
      3       0.080     0.080     0.080     1000
      4       0.080     0.080     0.080     1000
      5       0.153     0.214     0.178     1000
      6       0.080     0.080     0.080     1000
      7       0.080     0.080     0.080     1000
      8       0.053     0.088     0.071     1000
      9       0.080     0.080     0.080     1000
accuracy                           0.119   10000
macro avg       0.089     0.119     0.086   10000
weighted avg    0.089     0.119     0.086   10000
Confusion Matrix
[[92  26  0  0  0  42  0  0  4  0]
 [894  0  0  0  57  0  0  28  0  0]
 [748  9  0  0  145  0  0  13  0  1]
 [675  106  0  0  0  195  0  0  24  0]
 [689  112  0  0  0  184  0  0  35  0]
 [656  9  0  0  0  141  0  0  0  0]
 [1492  128  0  0  0  265  0  0  23  0]
 [731  86  0  0  0  171  0  0  12  0]
 [940  20  0  0  0  32  0  0  8  0]
 [868  51  0  0  0  66  0  0  15  0]]
```



```
-----
Part Two Task Two complete!
-----
(base) cs744@clifbee@clcs53:/CS744/AssignmentTwo$
```

Figure 3: Output snapshots on Node 1 for Part Two Task One.

```
(base) cs744@885fb17e8de6:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 2
-----
Starting Part Two Task Two!
-----
Running Rank: 2
GPU not available!
Files already uploaded and verified
Files already downloaded and verified
Directory: Checkpoints already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Logs/PartTwo already exists!
Directory: Logs/PartTwo already exists!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Checkpoints will be stored at Checkpoints/PartTwo/TaskTwo/
Logs will be stored at Logs/PartTwo/TaskTwo/logs_rank2.txt!
Evaluation report will be stored at Logs/PartTwo/TaskTwo/report_rank2.txt!
Prepare model for training!
Parameters: 923114
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.100 seconds
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.967 seconds
Batch Idx: 40, Batch Loss: 2.461, Batch Accuracy: 0.031, Batch Duration: 0.968 seconds
Successfully trained the model for 40 batches!
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.065 seconds
Avg Epoch Duration: 40.065 seconds
-----
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/EPOCH_1.pt
Test Loss: 2.753, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
0           0.120     0.221     0.212      1000
1           0.064     0.049     0.056      1000
2           0.000     0.000     0.000      1000
3           0.000     0.000     0.000      1000
4           0.000     0.000     0.000      1000
5           0.153     0.214     0.178      1000
6           0.000     0.000     0.000      1000
7           0.000     0.000     0.000      1000
8           0.053     0.000     0.014      1000
9           0.000     0.000     0.000      1000
accuracy          0.119      10000
macro avg       0.039     0.119     0.046      10000
weighted avg    0.039     0.119     0.046      10000
Confusion Matrix
[[921 26 0 0 0 47 0 0 4 0]
 [1884 49 0 0 0 87 0 0 18 0]
 [748 94 0 0 0 145 0 0 13 0]
 [675 106 0 0 0 195 0 0 24 0]
 [1689 112 0 0 0 184 0 0 45 0]
 [456 79 0 0 0 121 0 0 21 0]
 [682 129 0 0 0 255 0 0 23 0]
 [731 86 0 0 0 171 0 0 12 0]
 [940 20 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Two Task Two complete!
-----
(base) cs744@885fb17e8de6:/CS744/AssignmentTwo$
```

Figure 4: Output snapshots on Node 2 for Part Two Task One.

```
(base) cs744@9377d7e37183:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 3
-----
Starting Part Two Task Two!
-----
Running Rank: 3!
GPU not available!
Files already uploaded and verified
Files already downloaded and verified
Directory: Checkpoints already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Logs/PartTwo already exists!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Checkpoints will be stored at Checkpoints/PartTwo/TaskTwo/
Training log will be stored at Logs/PartTwo/TaskTwo/logs_rank3.txt!
Evaluation report will be stored at Logs/PartTwo/TaskTwo/report_rank3.txt!
Prepare model for training!
Parameters: 923114
-----
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.119 seconds
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.956 seconds
Batch Idx: 40, Batch Loss: 2.441, Batch Accuracy: 0.031, Batch Duration: 0.963 seconds
-----
Successfully trained the model for 40 batches!
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.068 seconds
Avg Epoch Duration: 40.068 seconds
-----
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/EPOCH_1.pt
Test Loss: 2.753, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
      0       0.120     0.721     0.212     1000
      1       0.064     0.849     0.056     1000
      2       0.000     0.000     0.000     1000
      3       0.000     0.000     0.000     1000
      4       0.000     0.000     0.000     1000
      5       0.153     0.214     0.178     1000
      6       0.000     0.000     0.000     1000
      7       0.000     0.000     0.000     1000
      8       0.053     0.000     0.014     1000
      9       0.000     0.000     0.000     1000
accuracy                           0.119   10000
macro avg       0.039     0.119     0.046   10000
weighted avg    0.039     0.119     0.046   10000
Confusion Matrix
[[971 26 0 0 0 49 0 0 4 0]
 [1884 49 0 0 0 87 0 0 18 0]
 [748 94 0 0 0 145 0 0 13 0]
 [675 106 0 0 0 195 0 0 24 0]
 [1689 112 0 0 0 184 0 0 45 0]
 [656 97 0 0 0 141 0 0 17 0]
 [1682 129 0 0 0 255 0 0 23 0]
 [731 86 0 0 0 171 0 0 12 0]
 [940 20 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Two Task Two complete!
-----
(base) cs744@9377d7e37183:/CS744/AssignmentTwo$
```

Figure 5: Output snapshots on Node 3 for Part Two Task One.

2.2 Part 2b: Sync gradient with allreduce using Gloo backend

2.2.1 Solution

The training process was completed successfully, producing the expected results. The final average train loss value was 5.784, the average test loss 2.753, while the test accuracy was 11.9%. Since we set the seed for PyTorch and NumPy to be 42 across the tasks, we were able to achieve the same loss and accuracy value for Task One and Task Two in Part Two. The expected difference in the batch duration is expected due to the difference in code. Table 3 denotes the average time it took to run 40 mini batch iterations and the epoch duration across the different nodes with a batch size of 64. From the table, we see that the average batch duration was 0.967 seconds with an average epoch duration of 40.066 seconds.

Table 2: Metrics across the different nodes for Part Two Task Two.

Node	Avg Batch Duration	Epoch Duration
Master Node	0.967	40.065
Node 1	0.967	40.066
Node 2	0.967	40.065
Node 3	0.967	40.068
Avg	0.967	40.066

```
(base) cs744@dd5a90e55317:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 0
-----
Starting Part Two Task Two!
-----
Running Rank: 0!
GPU not available!
Files already uploaded and verified
Files already downloaded and verified
Directory: Checkpoints already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Logs/PartTwo already exists!
Directory: Logs/PartTwo/TaskTwo already exists!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Creating checkpoints stored at Checkpoints/PartTwo/TaskTwo/
Training log will be stored att Logs/PartTwo/TaskTwo/logs/rank0.txt!
Evaluation report will be stored at Logs/PartTwo/TaskTwo/report_rank0.txt!
Prepare model for training!
Parameters: 923114
-----
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.121 seconds
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.966 seconds
Batch Idx: 40, Batch Loss: 2.443, Batch Accuracy: 0.031, Batch Duration: 0.963 seconds
-----
Successfully trained the model for 40 batches!
-----
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.065 seconds
-----
Avg Epoch Duration: 40.065 seconds
-----
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/EPOCH_1.pt
Test Loss: 2.763, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
0           0.120    0.721    0.212     1000
1           0.064    0.849    0.056     1000
2           0.000    0.000    0.000     1000
3           0.000    0.000    0.000     1000
4           0.000    0.000    0.000     1000
5           0.153    0.214    0.178     1000
6           0.000    0.000    0.000     1000
7           0.000    0.000    0.000     1000
8           0.053    0.000    0.014     1000
9           0.000    0.000    0.000     1000
accuracy          0.119      10000
macro avg       0.039    0.119    0.046     10000
weighted avg    0.039    0.119    0.046     10000
-----
Confusion Matrix
[[921 26 0 0 0 42 0 0 4 0]
 [854 49 0 0 0 87 0 0 10 0]
 [748 94 0 0 0 145 0 0 13 0]
 [675 106 0 0 0 195 0 0 24 0]
 [601 112 0 0 0 214 0 0 31 0]
 [1443 29 0 0 0 255 0 0 23 0]
 [1602 120 0 0 0 252 0 0 23 0]
 [731 86 0 0 0 171 0 0 12 0]
 [940 20 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Two Task Two complete!
-----
(base) cs744@dd5a90e55317:/CS744/AssignmentTwo$
```

Figure 6: Output snapshots on Master Node for Part Two Task Two.

```
(base) cs744@clifbee@clcs53:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 1
-----
Starting Part Two Task Two!

Running Rank: 1
GPU not available!
Files already downloaded and verified
Files already downloaded and verified
Directory: Checkpoints/PartTwo already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Logs already exists!
Directory: Logs/PartTwo already exists!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Checkpoints will be stored at: Checkpoints/PartTwo/TaskTwo!
Training logs will be stored at: Logs/PartTwo/TaskTwo/logs_rank1.txt!
Evaluation report will be stored at: Logs/PartTwo/TaskTwo/report_rank1.txt!
Prepare model for training!
Parameters: 9231114
-----
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.096 seconds
-----
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.958 seconds
-----
Batch Idx: 48, Batch Loss: 2.461, Batch Accuracy: 0.031, Batch Duration: 0.968 seconds
-----
Successfully trained the model for 48 batches!
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.066 seconds
Avg Epoch Duration: 40.066 seconds
-----
```



```
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/Epoch_1.pt
Test Loss: 2.753, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
      0       0.120     0.721     0.212     1000
      1       0.064     0.049     0.056     1000
      2       0.080     0.080     0.080     1000
      3       0.080     0.080     0.080     1000
      4       0.080     0.080     0.080     1000
      5       0.153     0.214     0.178     1000
      6       0.080     0.080     0.080     1000
      7       0.080     0.080     0.080     1000
      8       0.053     0.088     0.071     1000
      9       0.080     0.080     0.080     1000
accuracy                           0.119   10000
macro avg       0.089     0.119     0.086   10000
weighted avg    0.089     0.119     0.086   10000
-----
```



```
Confusion Matrix
[[92 26 0 0 0 42 0 0 4 0]
 [184 0 0 0 57 0 0 28 0]
 [748 94 0 0 0 145 0 0 13 0]
 [675 106 0 0 0 195 0 0 24 0]
 [689 112 0 0 0 184 0 0 35 0]
 [656 97 0 0 0 141 0 0 21 0]
 [1492 128 0 0 0 265 0 0 23 0]
 [731 86 0 0 0 171 0 0 12 0]
 [940 20 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Two Task Two complete!
-----
```

Figure 7: Output snapshots on Node 1 for Part Two Task Two.

```
(base) cs744@885fb17e8de6:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 2
-----
Starting Part Two Task Two!
-----
Running Rank: 2
GPU not available!
Files already uploaded and verified
Files already downloaded and verified
Directory: Checkpoints already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Logs/PartTwo already exists!
Directory: Logs/PartTwo already exists!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Checkpoints will be stored at Checkpoints/PartTwo/TaskTwo/
Logs will be stored at Logs/PartTwo/TaskTwo/
Evaluation report will be stored at Logs/PartTwo/TaskTwo/report_rank2.txt!
Prepare model for training!
Parameters: 923114
-----
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.100 seconds
-----
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.967 seconds
-----
Batch Idx: 40, Batch Loss: 2.461, Batch Accuracy: 0.031, Batch Duration: 0.968 seconds
-----
Successfully trained the model for 40 batches!
-----
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.065 seconds
-----
Avg Epoch Duration: 40.065 seconds
-----
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/EPOCH_1.pt
Test Loss: 2.763, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
0           0.120     0.921     0.212      1000
1           0.064     0.849     0.056      1000
2           0.000     0.000     0.000      1000
3           0.000     0.000     0.000      1000
4           0.000     0.000     0.000      1000
5           0.153     0.214     0.178      1000
6           0.000     0.000     0.000      1000
7           0.000     0.000     0.000      1000
8           0.053     0.000     0.014      1000
9           0.000     0.000     0.000      1000
accuracy                           0.119      10000
macro avg       0.039     0.119     0.046      10000
weighted avg    0.039     0.119     0.046      10000
Confusion Matrix
[[921 26 0 0 0 47 0 0 4 0]
 [1884 49 0 0 0 87 0 0 18 0]
 [748 94 0 0 0 145 0 0 13 0]
 [675 106 0 0 0 195 0 0 24 0]
 [1689 112 0 0 0 184 0 0 45 0]
 [456 9 0 0 0 41 0 0 1 0]
 [1682 129 0 0 0 255 0 0 23 0]
 [731 86 0 0 0 171 0 0 12 0]
 [940 20 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Two Task Two complete!
-----
(base) cs744@885fb17e8de6:/CS744/AssignmentTwo$
```

Figure 8: Output snapshots on Node 2 for Part Two Task Two.

```
(base) cs744@9377d7e37183:/CS744/AssignmentTwo$ python3 PartTwo/TaskTwo/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 3
-----
Starting Part Two Task Two!
-----
Running Rank: 3
GPU not available!
Files already uploaded and verified
Files already downloaded and verified
Directory: Checkpoints already exists!
Directory: Checkpoints/PartTwo already exists!
Directory: Logs/PartTwo already exists!
Directory: Logs/PartTwo/TaskTwo already exists!
Created directory: Checkpoints/PartTwo/TaskTwo!
Created directory: Logs/PartTwo/TaskTwo!
Checkpoints will be stored at Checkpoints/PartTwo/TaskTwo/logs/rank3.txt!
Training log will be stored at Logs/PartTwo/TaskTwo/logs/rank3.txt!
Evaluation report will be stored at Logs/PartTwo/TaskTwo/report_rank3.txt!
Prepare model for training!
Parameters: 923114
-----
Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.119 seconds
Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.950 seconds
Batch Idx: 40, Batch Loss: 2.446, Batch Accuracy: 0.031, Batch Duration: 0.943 seconds
Successfully trained the model for 40 batches!
Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.967 seconds, Epoch Duration: 40.068 seconds
Avg Epoch Duration: 40.068 seconds
-----
Metrics for the checkpoint: Checkpoints/PartTwo/TaskTwo/EPOCH_1.pt
Test Loss: 2.753, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
      0       0.120     0.721     0.212     1000
      1       0.064     0.849     0.056     1000
      2       0.000     0.000     0.000     1000
      3       0.000     0.000     0.000     1000
      4       0.000     0.000     0.000     1000
      5       0.153     0.214     0.178     1000
      6       0.000     0.000     0.000     1000
      7       0.000     0.000     0.000     1000
      8       0.053     0.000     0.014     1000
      9       0.000     0.000     0.000     1000
accuracy                           0.119   10000
macro avg       0.039     0.119     0.046   10000
weighted avg    0.039     0.119     0.046   10000
Confusion Matrix
[[971 26 0 0 0 49 0 0 4 0]
 [1884 49 0 0 0 87 0 0 18 0]
 [748 94 0 0 0 145 0 0 13 0]
 [675 106 0 0 0 195 0 0 24 0]
 [1689 112 0 0 0 184 0 0 15 0]
 [656 97 0 0 0 141 0 0 17 0]
 [1682 129 0 0 0 255 0 0 23 0]
 [731 86 0 0 0 171 0 0 12 0]
 [940 20 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Two Task Two complete!
-----
(base) cs744@9377d7e37183:/CS744/AssignmentTwo$
```

Figure 9: Output snapshots on Node 3 for Part Two Task Two.

3 Part Three: Distributed Data Parallel Training using Built in Module

3.1 Solution

The training process was completed successfully, producing the expected results. The final average train loss value was 5.784, the average test loss 2.753, while the test accuracy was 11.9%. Due to the same random seed for PyTorch and NumPy, we were able to achieve the same loss and accuracy value. We just made use of the inbuilt method instead of writing custom methods. Table ?? denotes the average time it took to run 40 mini batch iterations and the epoch duration across the different nodes with a batch size of 64. From the table, we see that the average batch duration was 0.943 seconds with an average epoch duration of 39.57 seconds.

Table 3: Metrics across the different nodes for Part Three.

Node	Avg Batch Duration	Epoch Duration
Master Node	0.943	39.066
Node 1	0.944	39.050
Node 2	0.943	39.060
Node 3	0.943	39.053
Avg	0.943	39.57

```
(base) cs744@dd5a98e55317:/CS744/AssignmentTwo$ python3 PartThree/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 0
-----
Starting Part Three!
-----
Running Rank: 0!
GPU not available!
File already uploaded and verified
File already uploaded and verified
Directory: Checkpoints already exists!
Created directory: Checkpoints/PartThree!
Directory: Logs already exists!
Created directory: Logs!
Logs directory: Logs/PartThree!
Checkpoints will be stored at: Checkpoints/PartThree!
Training logs will be stored at: Logs/PartThree/logs_rank0.txt!
Evaluation results will be stored at: Logs/PartThree/report_rank0.txt!
Prepared model for training!
parameters: 923114
-----
Rank: 0 Batch Idx: 0, Batch Loss: 2.659, Batch Accuracy: 0.878, Batch Duration: 1.064 seconds
Rank: 0 Batch Idx: 20, Batch Loss: 5.683, Batch Accuracy: 0.847, Batch Duration: 0.939 seconds
Rank: 0 Batch Idx: 40, Batch Loss: 2.461, Batch Accuracy: 0.851, Batch Duration: 0.902 seconds
Rank: 0 Successfully trained the model for 40 batches!
-----
Rank: 0 Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.898, Avg Batch Duration: 0.943 seconds, Epoch Duration: 39.066 seconds
Avg Epoch Duration: 39.066 seconds
-----

Metrics for the checkpoint: Checkpoints/PartThree/Epoch_1.pt
Test Loss: 5.763, Test Accuracy: 0.819
Classification Report
precision    recall   f1-score   support
      0       0.120     0.921     0.212     1000
      1       0.864     0.849     0.856     1000
      2       0.000     0.000     0.000     1000
      3       0.000     0.000     0.000     1000
      4       0.000     0.000     0.000     1000
      5       0.153     0.214     0.178     1000
      6       0.000     0.000     0.000     1000
      7       0.000     0.000     0.000     1000
      8       0.053     0.000     0.014     1000
      9       0.000     0.000     0.000     1000
accuracy          0.117     10000
macro avg       0.839     0.119     0.846     10000
weighted avg    0.839     0.119     0.846     10000

Confusion Matrix
[[921 26 0 0 0 49 0 0 4 0]
 [1854 49 0 0 0 87 0 0 10 0]
 [1704 0 0 0 0 0 145 0 0 0]
 [1675 186 0 0 0 195 0 0 24 0]
 [689 112 0 0 0 184 0 0 15 0]
 [663 97 0 0 0 214 0 0 26 0]
 [1692 128 0 0 0 255 0 0 23 0]
 [731 0 0 0 0 0 217 0 0 0]
 [1940 20 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Three completed!
-----
(base) cs744@dd5a98e55317:/CS744/AssignmentTwo$
```

Figure 10: Output snapshots on Master Node for Part Three.

```

(base) cs744@clif9be@aic53:/CS744/AssignmentTwo$ python3 PartThree/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 1
-----
Starting Part Three!
-----
Running Rank: 1
GPU not available!
Files already downloaded and verified
Files already downloaded and verified
Directory: Checkpoints/PartThree!
Created directory: Checkpoints/PartThree!
Directory: Logs already exists!
Logs directory will be created!
Checkpoints will be stored at: Checkpoints/PartThree!
Training logs will be stored at: Logs/PartThree/logs_rank1.txt!
Evaluation report will be stored at: Logs/PartThree/report_rank1.txt!
Prepared model for training!
Parameters: 920114
-----
Rank: 1 Batch Idx: 0, Batch Loss: 2.599, Batch Accuracy: 0.078, Batch Duration: 1.060 seconds
Rank: 1 Batch Idx: 28, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.933 seconds
Rank: 1 Batch Idx: 49, Batch Loss: 2.461, Batch Accuracy: 0.031, Batch Duration: 0.905 seconds
Rank: 1 Successfully trained the model for 48 batches!
-----
Rank: 1 Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.944 seconds, Epoch Duration: 39.058 seconds
Avg Epoch Duration: 39.050 seconds
-----
```



```

Metrics for the checkpoint: Checkpoints/PartThree/Epoch_1.pt
Test Loss: 2.765, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
      0       0.120     0.021     0.212     1000
      1       0.044     0.049     0.056     1000
      2       0.000     0.000     0.000     1000
      3       0.000     0.000     0.000     1000
      4       0.000     0.000     0.000     1000
      5       0.153     0.216     0.178     1000
      6       0.000     0.000     0.000     1000
      7       0.000     0.000     0.000     1000
      8       0.053     0.000     0.014     1000
      9       0.000     0.000     0.000     1000
accuracy          0.119     10000
macro avg       0.039     0.019     0.046     10000
weighted avg    0.039     0.019     0.046     10000
-----
```



```

Confusion Matrix
[[921 26 0 0 0 49 0 0 4 0]
 [854 49 0 0 0 87 0 0 10 0]
 [748 94 0 0 0 107 0 0 51 0]
 [749 0 0 0 105 0 0 24 0]
 [689 112 0 0 0 184 0 0 15 0]
 [663 97 0 0 0 214 0 0 26 0]
 [682 128 0 0 0 255 0 0 23 0]
 [721 50 0 0 0 211 0 0 32 0]
 [1948 29 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
```



```

Part Three completed!
```

Figure 11: Output snapshots on Node 1 for Part Three.

```
(base) cs744@085fb1e8de6:/CS744/AssignmentTwo$ python3 PartThree/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 2
-----
Starting Part Three!
-----
Running Rank: 2!
GPU not available!
Files already downloaded and verified
Files already downloaded and verified
Directory: Checkpoints/PartThree!
Created directory: Checkpoints/PartThree!
Directory: Logs already exists!
Created directory: Logs/PartThree!
Logs directory already exists!
Checkpoints will be stored at: Checkpoints/PartThree!
Training logs will be stored at: Logs/PartThree/logs_rank2.txt!
Evaluation report will be stored at: Logs/PartThree/report_rank2.txt!
Prepared model for training!
Parameters: 220114
-----
Rank: 2 Batch Idx: 0, Batch Loss: 2.599, Batch Accuracy: 0.078, Batch Duration: 1.066 seconds
Rank: 2 Batch Idx: 28, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.933 seconds
-----
Rank: 2 Batch Idx: 46, Batch Loss: 2.461, Batch Accuracy: 0.031, Batch Duration: 0.902 seconds
-----
Rank: 2 Successfully trained the model for 46 batches!
Rank: 2 Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.943 seconds, Epoch Duration: 39.060 seconds
Avg Epoch Duration: 39.060 seconds
-----
```

```
Metrics for the checkpoint: Checkpoints/PartThree/Epoch_1.pt
Test Loss: 2.753, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
      0       0.128     0.721     0.212     1000
      1       0.424     0.049     0.065     1000
      2       0.000     0.000     0.000     1000
      3       0.000     0.000     0.000     1000
      4       0.000     0.000     0.000     1000
      5       0.000     0.252     0.075     1000
      6       0.000     0.000     0.000     1000
      7       0.000     0.000     0.000     1000
      8       0.053     0.008     0.014     1000
      9       0.000     0.000     0.000     1000
accuracy                           0.119    10000
macro avg       0.039     0.119     0.046    10000
weighted avg    0.039     0.119     0.046    10000

Confusion Matrix
[[921 26 0 0 0 49 0 0 4 0]
 [854 49 0 0 0 87 0 0 10 0]
 [748 94 0 0 0 145 0 0 13 0]
 [670 106 0 0 0 109 0 0 8 0]
 [1469 117 0 0 0 124 0 0 25 0]
 [1643 97 0 0 0 214 0 0 26 0]
 [1692 128 0 0 0 255 0 0 23 0]
 [1731 86 0 0 0 171 0 0 12 0]
 [1498 29 0 0 0 32 0 0 8 0]
 [1848 51 0 0 0 66 0 0 15 0]]
-----
Part Three completed!
```

Figure 12: Output snapshots on Node 2 for Part Three.

```

(base) cs744@9377d7c3715$ ./CS744/AssignmentTwo$ python3 PartThree/main.py --master-ip 172.18.0.2 --num-nodes 4 --rank 3
-----
Starting Part Three!
-----
Rank: 3 GPU not available!
Files already downloaded and verified
Files already downloaded and verified
Directory: Checkpoints already exists!
Created directory: Checkpoints/PartThree!
Directory: Logs already exists!
Created directory: Logs/PartThree!
Checkpoints will be stored at: Checkpoints/PartThree!
Training logs will be stored at: Logs/PartThree/logs_rank3.txt!
Evaluation report will be stored at: Logs/PartThree/report_rank3.txt!
Prepared model for training!
Parameters: 923114
-----
Rank: 3 Batch Idx: 0, Batch Loss: 2.559, Batch Accuracy: 0.078, Batch Duration: 1.069 seconds
Rank: 3 Batch Idx: 28, Batch Loss: 5.683, Batch Accuracy: 0.047, Batch Duration: 0.942 seconds
Rank: 3 Batch Idx: 49, Batch Loss: 2.461, Batch Accuracy: 0.031, Batch Duration: 0.984 seconds
Rank: 3 Successfully trained the model for 49 batches!
Rank: 3 Epoch: 1, Train Loss: 5.784, Train Accuracy: 0.098, Avg Batch Duration: 0.943 seconds, Epoch Duration: 39.053 seconds
Avg Epoch Duration: 39.053 seconds
-----
Metrics for the checkpoint: Checkpoints/PartThree/Epoch_1.pt
Test Loss: 2.703, Test Accuracy: 0.119
Classification Report
precision    recall   f1-score   support
      0       0.128     0.021     0.012     1000
      1       0.044     0.049     0.056     1000
      2       0.000     0.000     0.000     1000
      3       0.000     0.000     0.000     1000
      4       0.000     0.000     0.000     1000
      5       0.153     0.216     0.178     1000
      6       0.000     0.000     0.000     1000
      7       0.000     0.000     0.000     1000
      8       0.053     0.000     0.014     1000
      9       0.000     0.000     0.000     1000
accuracy          0.119     10000
macro avg       0.039     0.119     0.046     10000
weighted avg    0.039     0.119     0.046     10000
-----
Confusion Matrix
[[921 26 0 0 0 49 0 0 4 0]
 [854 49 0 0 0 87 0 0 10 0]
 [749 26 0 0 0 105 0 0 31 0]
 [675 0 0 0 0 195 0 0 24 0]
 [689 112 0 0 0 184 0 0 15 0]
 [663 97 0 0 0 214 0 0 26 0]
 [682 128 0 0 0 255 0 0 23 0]
 [721 50 0 0 0 211 0 0 32 0]
 [1948 29 0 0 0 32 0 0 8 0]
 [868 51 0 0 0 66 0 0 15 0]]
-----
Part Three completed!
-----
(base) cs744@9377d7c3715$ ./CS744/AssignmentTwo$ 

```

Figure 13: Output snapshots on Node 3 for Part Three.

4 Difference or lack of difference among different setups

The training time per iteration decreased as we progressed from Part 1 to DDP, with Part 1 taking the longest average time 1.477 seconds and DDP the shortest 0.943 seconds. This suggests that distributed training can significantly improve speed by parallelizing across nodes, compared to single-node training in Part 1.

Also, higher-level distributed APIs like AllReduce and DDP outperformed lower-level gather/scatter implementations. AllReduce uses a more efficient ring-based approach than scatter/gather. DDP further optimizes communication by bucketing small gradients before synchronizing, reducing total AllReduce calls. Leveraging multiple nodes and optimized distributed training APIs like

DDP can substantially reduce iteration time versus single-node training. The results highlight the performance benefits of distributed training, as well as efficient communication primitives like AllReduce.

In our analysis, we examined and compared the average loss across various setup methods. Overall, we observed that there is generally minimal variation in the average loss among all setups, indicating that different distributed configurations do not substantially affect training loss. This observation is logical because all setups use the same training data, maintain identical model structures, and adhere to consistent parameters such as total epochs, batch sizes, and training parameters. These factors contribute to the stability of average loss across setups.

However, there are still slight discrepancies in both the average loss and the loss recorded at intervals of every 20 iterations among different setups. This could be attributed to two potential factors: Firstly, the VGG models utilized in our study contain Batch Normalization (BN) layers, and the parameters within these layers are not synchronized across all nodes during distributed training. Consequently, this lack of synchronization introduces uncertainty and randomness into the training process, leading to marginal differences in loss across setups. Secondly, due to the inherent limitations of floating-point computation precision, minor errors may occur during training, further contributing to variability in loss computation results.

Similarly, while there is generally little disparity in test accuracy among different setups, slight variations still exist. These discrepancies likely stem from the same factors influencing average loss variation.

5 Comment on the scalability of distributed machine learning based on your results

We compare the average iteration time between the setups with different numbers of nodes to evaluate the scalability of the method. We can see that part 3 is $1.56 \times$ faster than part 1. Thus, we can conclude that the speed does not improve linearly, even with an increase in the number of nodes.

Distributing training across multiple machines comes with a cost - it takes time to synchronize the gradient updates between nodes. How much overhead depends on the network connecting the machines. But when our data or model is too big for one machine, distributed data parallelism is our only option.

In our current setup with 4 Docker containers linked locally, the nodes communicate locally rather than external links. So the main overhead is just opening sockets and sending data back and forth. As the paper mentioned, we could

reduce this cost by doing a few local gradient updates before averaging across nodes.

Since our nodes are already well-connected on the same machine, the communication overhead isn't too bad. But as we scale up, optimizing the synchronization step will become more and more important. Tricks like local averaging before cross-node updates can help minimize the impact on training time. But distributing the work is the only way forward when limited by single-machine resources.

6 Contribution

- Badri
 - 1. Acted as the team lead and formulated the majority of the code.
 - 2. Wrote the code for the different helper functions, and also formulated the idea to use flags to train multiple variants of the model.
 - 3. Worked on training the VGG-16 model and generated results.
 - 4. Maintained version control with Git.
 - 5. Generated the report using L^AT_EX.
- Shreyas
 - 1. Formulated the code for Distributed Training.
 - 2. Implemented gather-scatter mechanism for gradient aggregation.
 - 3. Wrote the content for the report.
- Rohan
 - 1. Applied ring reduce to synchronize gradients across different nodes.
 - 2. Worked on code documentation.
 - 3. Formulated the code for the Distributed DataParallel module.
- Nevin
 - 1. Formulated the code for the Distributed DataParallel module.
 - 2. Worked on the content for the report.
 - 3. Documented the code

The entire team had an in-depth discussion about the results of our run.