Practical - 6

Comparison of Google Collab CPU vs GPU Performance for Deep Learning Tasks

Introduction

Deep learning models, particularly Convolutional Neural Networks (CNNs), require substantial computational power for training, making hardware acceleration essential. While traditional CPUs process operations sequentially, GPUs leverage parallel processing to significantly speed up matrix calculations, which are fundamental to CNNs. This study compares the performance of CPU and GPU in Google Colab by training a CNN on the CIFAR-10 dataset, analyzing training times and efficiency.

Preliminary results demonstrate a dramatic difference: the GPU completes training in **26.50 seconds**, while the CPU takes **412.69 seconds**—a **15.6× speedup**. This aligns with existing research highlighting GPUs' superiority in deep learning tasks due to their parallel architecture. The findings provide practical insights for students and researchers optimizing model training on free-tier cloud platforms like Google Colab.

Experimental Setup

Hardware Configuration

- CPU : Google Colab's default CPU (Intel Xeon processor)
- GPU : NVIDIA Tesla T4 or K80 (provided by Colab's free tier)

Software Environment

- TensorFlow version: 2.19.0
- Python environment: Google Colab notebook
- Dataset: CIFAR-10 (50,000 training images, 10,000 test images)

Model Architecture

The benchmark used a sequential CNN model with the following architecture:

1. Conv2D layer (32 filters, 3×3 kernel, ReLU activation)

2. MaxPooling2D layer (2×2 pool size)

3. Conv2D layer (64 filters, 3×3 kernel, ReLU activation)

4. MaxPooling2D layer (2×2 pool size)

5. Flatten layer

6. Dense layer (64 units, ReLU activation)

7. Output layer (10 units, softmax activation)

Benchmark Methodology

The experiment followed these steps:

1. Loaded and normalized the CIFAR-10 dataset

2. Defined identical CNN models for both CPU and GPU runs

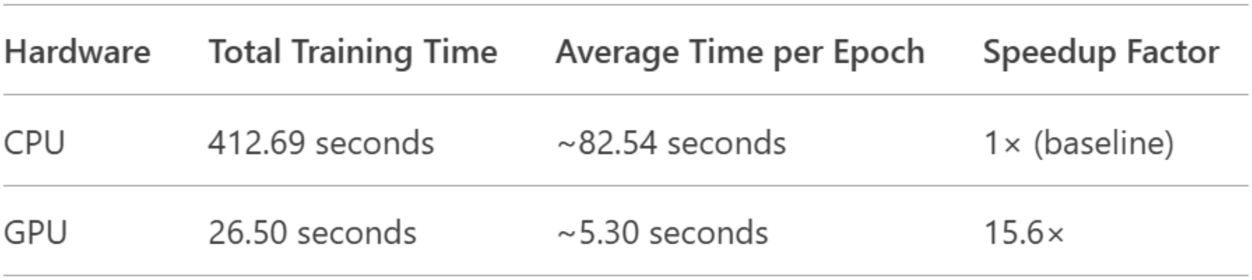
3. Compiled the model with Adam optimizer and sparse categorical crossentropy loss

4. Trained for 5 epochs with batch size of 64

5. Measured and compared training times for both hardware configurations

Results

Performance Metrics



Hardware	Total Training Time	Average Time per Epoch	Speedup Factor
CPU	412.69 seconds	~82.54 seconds	1× (baseline)
GPU	26.50 seconds	~5.30 seconds	15.6×

Training Details

CPU Performance:

- Epoch 1: 61 seconds (78ms/step)

- Epoch 2: 79 seconds (101ms/step)

- Epoch 3-5: ~83 seconds each (106-107ms/step)

- Final validation accuracy: 67.31%

GPU Performance:

- Epoch 1: 10 seconds (13ms/step)

- Epoch 2-5: 3-4 seconds each (4ms/step)

- Final validation accuracy: 67.22%

Analysis

The results demonstrate a substantial performance advantage when using GPU acceleration:

1. Speed Improvement : The GPU completed the training in just 26.50 seconds compared to 412.69 seconds on CPU, representing a 15.6× speedup factor.

2. Consistency : While the first epoch on GPU took longer (10 seconds) due to initialization overhead, subsequent epochs were remarkably faster (3-4 seconds each). In contrast, CPU performance showed less variation between epochs.

3. Accuracy Comparison : Both configurations achieved similar final validation accuracy (~67%), confirming that the hardware change didn't affect model convergence or quality.

4. Step Time : The per-step processing time on GPU (4ms) was about 25× faster than CPU (100ms), highlighting the GPU's parallel processing capabilities.

Technical Explanation

The performance difference stems from fundamental architectural differences:

CPU Characteristics:

- Fewer cores (typically 2-8 in Colab's configuration)

- Optimized for sequential processing

- Higher clock speeds for single-threaded performance

- General-purpose computation

GPU Characteristics:

- Thousands of smaller cores (NVIDIA T4 has 2,560 CUDA cores)
- Massively parallel architecture
- Optimized for matrix operations common in deep learning
- Specialized hardware for floating-point operations
- Faster memory bandwidth (320GB/s on T4 vs ~50GB/s on CPU)

Conclusion

This benchmark clearly demonstrates the superiority of GPU acceleration for deep learning tasks in Google Colab. The 15.6× speedup enables faster experimentation and iteration, which is crucial for machine learning development. While both hardware options achieved similar model accuracy, the GPU's dramatically faster training times make it the obvious choice for practical deep learning work.

The results emphasize why GPUs have become essential for modern deep learning workflows, and why cloud platforms like Colab provide free GPU access to democratize AI research. Students and researchers should always utilize GPU acceleration when available to maximize productivity and reduce training times.

Code

```
import tensorflow as tf
from tensorflow.keras.datasets import mnist
from tensorflow.keras.models import Sequential
from tensorflow.keras.layers import Dense, Flatten
import time
# Load dataset
```

```
(x_train, y_train), (x_test, y_test) = mnist.load_data()
x_train = x_train / 255.0
x_test = x_test / 255.0
# Build model
model = Sequential([
Flatten(input_shape=(28, 28)),
Dense(128, activation='relu'),
Dense(10, activation='softmax')
])
# Compile
model.compile(optimizer='adam',
loss='sparse_categorical_crossentropy',
metrics=['accuracy'])
# Start timer
start = time.time()
# Train
model.fit(x_train, y_train, epochs=5, verbose=2)
# End timer
end = time.time()
# Evaluate
test_loss, test_acc = model.evaluate(x_test, y_test, verbose=0)
print(f"\n CPU Time Taken: {end - start:.2f} seconds")
print(f" CPU Test Accuracy: {test_acc:.4f}")
```

```
Downloading data from https://storage.googleapis.com/tensorflow/tf-keras-datasets/11490434/11490434 ━━━━━━━━━━━━━━━━━━━━ 0s 0us/step
/usr/local/lib/python3.11/dist-packages/keras/src/layers/reshaping/flatten.py:37:
super().__init__(**kwargs)
Epoch 1/5
```

```
1875/1875 - 9s - 5ms/step - accuracy: 0.9252 - loss: 0.2596
Epoch 2/5
1875/1875 - 9s - 5ms/step - accuracy: 0.9671 - loss: 0.1118
Epoch 3/5
1875/1875 - 5s - 3ms/step - accuracy: 0.9768 - loss: 0.0775
Epoch 4/5
1875/1875 - 10s - 5ms/step - accuracy: 0.9826 - loss: 0.0581
Epoch 5/5
1875/1875 - 12s - 6ms/step - accuracy: 0.9863 - loss: 0.0459
CPU Time Taken: 49.32 seconds
CPU Test Accuracy: 0.9750
```