

Some GPU Tips

Batch your data. Training deep learning models on GPUs is more efficient when done in batches. Divide your data into smaller batches that can fit into GPU memory, and use batch processing during training.

```
train_dataloader = DataLoader(train_data, sampler=train_sampler, batch_size=16)
```

Monitor GPU usage and memory. Keep an eye on your GPU usage and memory consumption during training. If your model is consuming too much memory, consider reducing batch size, using mixed precision training, or reducing the model complexity.

```
gpu_usage()
```

ID	GPU	MEM
0	97%	14%
1	0%	0%

Explicitly clear variables you no longer need using the del keyword, e.g., del model, data. Use the gc.collect() function from the gc module to trigger garbage collection and free up memory.

```
gc.collect()
```

```
torch.cuda.empty_cache()
```

ID	GPU	MEM
0	0%	14%
1	0%	0%

Regularly save your model checkpoints. To avoid losing progress due to kernel timeouts or crashes, regularly save your model checkpoints during training. This will allow you to resume training from the last saved checkpoint if needed.

```
# Save checkpoint at the end of each epoch
```

```
checkpoint_dir = 'checkpoints'
```

```
os.makedirs(checkpoint_dir, exist_ok=True)
```

```
checkpoint_path = os.path.join(checkpoint_dir,  
f'checkpoint_epoch_{epoch}.pt')
```

```
torch.save({
```

```
    'epoch': epoch,
```

```
    'model_state_dict': model.state_dict(),
```

```
    'optimizer_state_dict': optimizer.state_dict(),
```

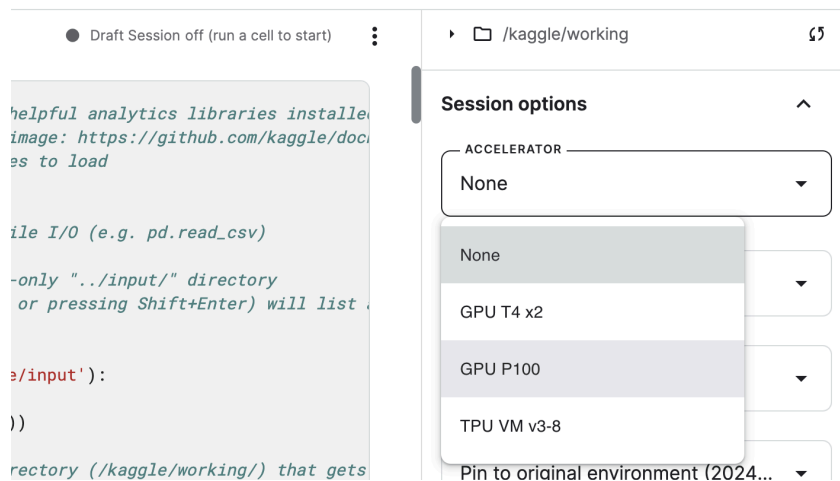
```

        'loss': loss,
    }, checkpoint_path)
    print(f'Checkpoint saved at: {checkpoint_path}')

```

Utilize Kaggle's GPU resources efficiently. Kaggle provides GPU resources for free, but they are limited. Make sure to stop your kernel when not in use to avoid wasting GPU time.

Experiment with different GPU configurations. Kaggle provides different GPU types (e.g., Tesla P100, T4). Try experimenting with different GPU types to see if they impact your model's performance or training speed.



The screenshot displays the Kaggle interface. On the left, a code editor shows a Python script snippet for saving a checkpoint. On the right, the 'Session options' panel is open, showing a dropdown menu for 'ACCELERATOR'. The menu is open, displaying four options: 'None', 'GPU T4 x2', 'GPU P100' (which is highlighted), and 'TPU VM v3-8'. Below the dropdown, there is a button labeled 'Pin to original environment (2024...'.