Analysis of the impact of using a CPU vs GPU on training time taken

The following code trains a model on the XOR dataset using the CPU and then using the GPU to train, and then outputs the training time taken.

[5]:
```
import os

from school_project.models.cpu.cat_recognition import CatRecognitionModel as␣
 ↪CPUModel
from school_project.models.gpu.cat_recognition import CatRecognitionModel as␣
 ↪GPUModel

# Change to root directory of project
os.chdir(os.getcwd())

model = CPUModel(hidden_layers_shape=[100, 100],
                 train_dataset_size=209,
                 learning_rate=0.1,
                 use_relu=True)
model.create_model_values()
model.train(epoch_count=3_500)

print(f"CPU Training Time: {model.training_time}s")

model = GPUModel(hidden_layers_shape=[100, 100],
                 train_dataset_size=209,
                 learning_rate=0.1,
                 use_relu=True)
model.create_model_values()
model.train(epoch_count=3_500)

print(f"GPU Training Time: {model.training_time}s")
```

```
CPU Training Time: 160.33s
GPU Training Time: 43.24s
```

As shown above, the GPU is almost four times faster at training the model than the CPU, showing how beneficial it is to utilise the parallel computations of the GPU