Homework 3 – Deep Learning

By: Joshua Lumpkin

Dataset:

I used the squad dataset from datasets library. Figured it was the easiest to manage with a train and validation split built in and not iterate through a large dataset in my local machine.

```
DatasetDict({
    train: Dataset({
        features: ['id', 'title', 'context', 'question', 'answers'],
        num_rows: 87599
    })
    validation: Dataset({
        features: ['id', 'title', 'context', 'question', 'answers'],
        num_rows: 10570
    })
})
```

Preprocessing:

I preprocessed training and validation separately, I knew the max length was 384, so I set that as a static variable. Stride was set at 128.

Training:

I used huggingfaces trainer and trainingargs from transformers, this was a great learning experience and cool tool to use. It made training very straightforward. But does create some limitations if you want to customize training beyond its base capabilities.

- Epochs: 3
- Learning rate: 2e-5
- Weight decay: 0.01
- Total time: 48mins
- train loss: 0.963
- Steps: 8319

Step	Training Loss
500	2.255600
1000	1.354300
1500	1.206400
2000	1.152000
2500	1.097000
3000	0.967400
3500	0.870800
4000	0.860100
4500	0.850700
5000	0.828800
5500	0.829100
6000	0.692700
6500	0.655000
7000	0.664800
7500	0.664200
8000	0.663300

Training output metrics

```
TrainOutput(global_step=8319, training_loss=0.9630235519986956, metrics={'train_runtime': 2897.9513, 'train_samples_per_second': 91.854, 'train_steps_per_second': 2.871,
'total_flos': 5.216534983896422e+16, 'train_loss': 0.9630235519986956, 'epoch': 3.0})
```

Metrics:

My mini validation dataset output on raw ‘bert cased model.’

```
{'exact_match': 0.0, 'f1': 3.598412698412699}
```

Metrics on same mini validation dataset, after training the model

```
{'exact_match': 84.0, 'f1': 89.51428571428572}
```