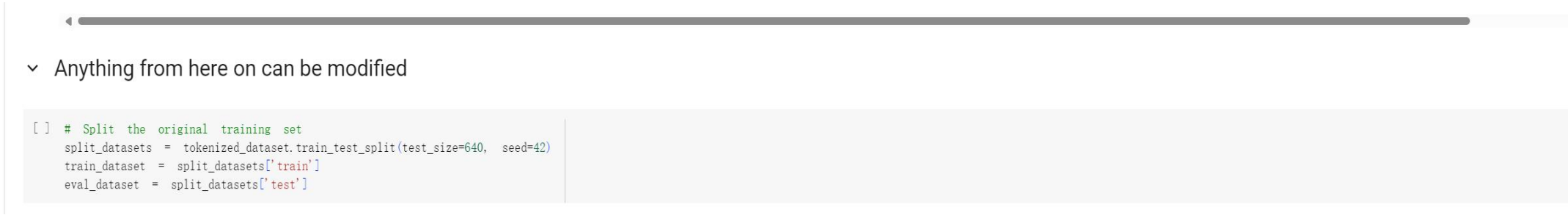
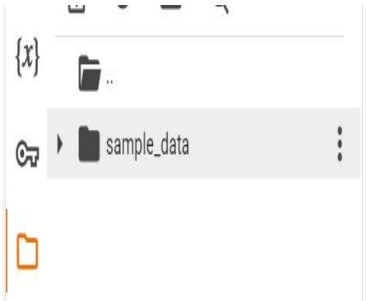
The steps to reproduce the result:

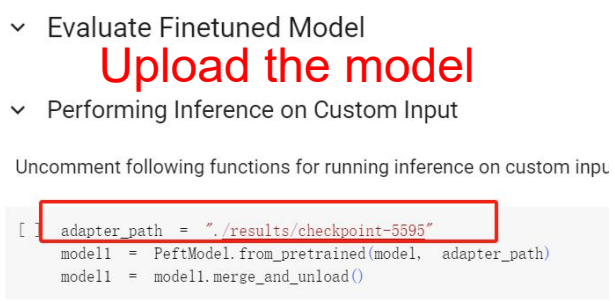
1. Open the notebook file
2. Upload the model in the “best” file folder and the data in the “data” floder
3. First run the code until it reaches the “Setup LoRA Config” block
4. Then run the code in the “Evaluate Finetuned Model” block to get model1
5. Then run the code in the “Run Inference on unlabelled dataset” block to get the csv file

1. Run the code from beginning to this code(this code also need to run)

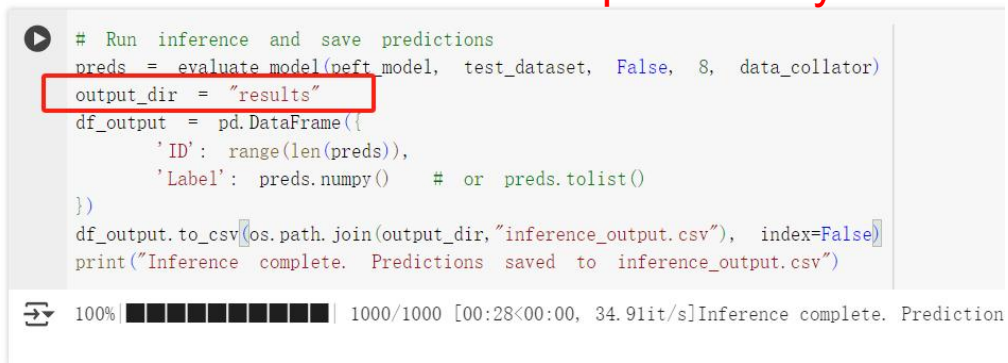


2. Upload the related model and test dataset here





Set the output path, it depends on you



3. Run this part to evaluate the model

Evaluate Finetuned Model

Performing Inference on Custom Input

Uncomment following functions for running inference on custom inputs

```
adapter_path = "./results/checkpoint-5595"
model1 = PeftModel.from_pretrained(model, adapter_path)
model1 = model1.merge_and_unload()
```

```
# def classify(model, tokenizer, text):
#     device = torch.device("cuda" if torch.cuda.is_available() else "cpu")
#     inputs = tokenizer(text, truncation=True, padding=True, return_tensors="pt").to(devic
#     output = model(**inputs)

#     prediction = output.logits.argmax(dim=-1).item()

#     print(f'\n Class: {prediction}, Label: {id2label[prediction]}, Text: {text}')
#     return id2label[prediction]
```

```
# classify( peft_model, tokenizer, "Kederis proclaims innocence Olympic champion Kostas Ked
# classify( peft_model, tokenizer, "Wall St. Bears Claw Back Into the Black (Reuters) Re
```

Run Inference on eval_dataset

4. Run this part to get csv file predicted by the model

Run Inference on unlabelled dataset

```
#Load your unlabelled data
unlabelled_dataset = pd.read_pickle("test_unlabelled.pkl")
test_dataset = unlabelled_dataset.map(preprocess, batched=True, remove_columns=["text"])
unlabelled_dataset
```

```
Map: 100% 8000/8000 [00:05<00:00, 1491.94 examples/s]
Dataset({
    features: ['text'],
    num_rows: 8000
})
```

```
# Run inference and save predictions
preds = evaluate_model(peft_model, test_dataset, False, 8, data_collator)
output_dir = "results"
df_output = pd.DataFrame({
        'ID': range(len(preds)),
        'Label': preds.numpy()    # or preds.tolist()
})
df_output.to_csv(os.path.join(output_dir,"inference_output.csv"), index=False)
print("Inference complete. Predictions saved to inference_output.csv")
```

```
100%|██████████| 1000/1000 [00:28<00:00, 34.91it/s]Inference complete. Predictions saved to inference_output.csv
```