```
pip install transformers datasets torch accelerate
```

```
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (6.0.2)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (2024.9.1
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from transformers) (2.32.3)
Requirement already satisfied: safetensors>=0.4.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.4.5)
Requirement already satisfied: tokenizers<0.21,>=0.20 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.2
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-packages (from transformers) (4.66.6)
Requirement already satisfied: pyarrow>=15.0.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (17.0.0)
Collecting dill<0.3.9,>=0.3.0 (from datasets)
  Downloading dill-0.3.8-py3-none-any.whl.metadata (10 kB)
Requirement already satisfied: pandas in /usr/local/lib/python3.10/dist-packages (from datasets) (2.2.2)
Collecting xxhash (from datasets)
  Downloading xxhash-3.5.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (12 kB)
Collecting multiprocess<0.70.17 (from datasets)
  Downloading multiprocess-0.70.16-py310-none-any.whl.metadata (7.2 kB)
Collecting fsspec<=2024.9.0,>=2023.1.0 (from fsspec[http]<=2024.9.0,>=2023.1.0->datasets)
  Downloading fsspec-2024.9.0-py3-none-any.whl.metadata (11 kB)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from datasets) (3.10.10)
Requirement already satisfied: typing-extensions>=4.8.0 in /usr/local/lib/python3.10/dist-packages (from torch) (4.12.2)
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch) (3.4.2)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch) (3.1.4)
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.10/dist-packages (from torch) (1.13.1)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from sympy==1.13.1->torch)
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages (from accelerate) (5.9.5)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.3.
Requirement already satisfied: attrs>=17.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (24.2.0)
Requirement already satisfied: frozenlist>=1.1.1 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.5
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (6
Requirement already satisfied: yarl<2.0,>=1.12.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.1
Requirement already satisfied: async-timeout<5.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->transf
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.1
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->transformers
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->transformers
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->torch) (3.0.2)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.2)
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.2)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas->
Requirement already satisfied: propcache>=0.2.0 in /usr/local/lib/python3.10/dist-packages (from yarl<2.0,>=1.12.0->aioht
Downloading datasets-3.1.0-py3-none-any.whl (480 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 480.6/480.6 kB 10.6 MB/s eta 0:00:00
Downloading dill-0.3.8-py3-none-any.whl (116 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 116.3/116.3 kB 7.8 MB/s eta 0:00:00
Downloading fsspec-2024.9.0-py3-none-any.whl (179 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 179.3/179.3 kB 12.3 MB/s eta 0:00:00
Downloading multiprocess-0.70.16-py310-none-any.whl (134 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 134.8/134.8 kB 8.6 MB/s eta 0:00:00
Downloading xxhash-3.5.0-cp310-cp310-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (194 kB)
   ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 194.1/194.1 kB 10.2 MB/s eta 0:00:00
Installing collected packages: xxhash, fsspec, dill, multiprocess, datasets
  Attempting uninstall: fsspec
    Found existing installation: fsspec 2024.10.0
    Uninstalling fsspec-2024.10.0:
      Successfully uninstalled fsspec-2024.10.0
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour
gcsfs 2024.10.0 requires fsspec==2024.10.0, but you have fsspec 2024.9.0 which is incompatible.
Successfully installed datasets-3.1.0 dill-0.3.8 fsspec-2024.9.0 multiprocess-0.70.16 xxhash-3.5.0
```

```
pip install peft
```

```
Requirement already satisfied: peft in /usr/local/lib/python3.10/dist-packages (0.13.2)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from peft) (1.26.4)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from peft) (24.2)
Requirement already satisfied: psutil in /usr/local/lib/python3.10/dist-packages (from peft) (5.9.5)
Requirement already satisfied: pyyaml in /usr/local/lib/python3.10/dist-packages (from peft) (6.0.2)
Requirement already satisfied: torch>=1.13.0 in /usr/local/lib/python3.10/dist-packages (from peft) (2.5.0+cu121)
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-packages (from peft) (4.46.2)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from peft) (4.66.6)
Requirement already satisfied: accelerate>=0.21.0 in /usr/local/lib/python3.10/dist-packages (from peft) (1.1.1)
Requirement already satisfied: safetensors in /usr/local/lib/python3.10/dist-packages (from peft) (0.4.5)
Requirement already satisfied: huggingface-hub>=0.17.0 in /usr/local/lib/python3.10/dist-packages (from peft) (0.26.2)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.17.0->peft) (
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.17.0-
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.17.0->peft) (
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hu
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch>=1.13.0->peft) (3.4.2)
Requirement already satisfied: jinja2 in /usr/local/lib/python3.10/dist-packages (from torch>=1.13.0->peft) (3.1.4)
Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.10/dist-packages (from torch>=1.13.0->peft) (1.13.1
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from sympy==1.13.1->torch>=1
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from transformers->peft) (202
Requirement already satisfied: tokenizers<0.21,>=0.20 in /usr/local/lib/python3.10/dist-packages (from transformers->peft
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from jinja2->torch>=1.13.0->pe
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->huggin
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-hub>=0
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-h
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface-h
```

```
import pandas as pd
import torch
from transformers import RobertaTokenizer, RobertaForSequenceClassification, Trainer, TrainingArguments
from datasets import Dataset, DatasetDict
from transformers import AdamW
from peft import LoraConfig, get_peft_model
```

```
df = pd.read_excel('/content/final_clinical_mutation_data.xlsx')
```

```
df = df.head(10000)
```

Double-click (or enter) to edit

```
# Preprocess the dataset
df['text'] = df[['Hugo_Symbol', 'Chromosome', 'Consequence', 'Variant_Type', 'Reference_Allele', 'Tumor_Seq_Allele1', 'Tumor_
df = df[['text', 'Variant_Classification']]
```

```
# Encode labels
label_mapping = {label: idx for idx, label in enumerate(df['Variant_Classification'].unique())}
df['label'] = df['Variant_Classification'].map(label_mapping)
```

```
# Split into train and test
train_df = df.sample(frac=0.8, random_state=42)
test_df = df.drop(train_df.index)
```

```
# Convert to Hugging Face Dataset
train_dataset = Dataset.from_pandas(train_df[['text', 'label']])
test_dataset = Dataset.from_pandas(test_df[['text', 'label']])
dataset = DatasetDict({"train": train_dataset, "test": test_dataset})
```

```
# Load tokenizer and model
tokenizer = RobertaTokenizer.from_pretrained('roberta-base')
model = RobertaForSequenceClassification.from_pretrained('roberta-base', num_labels=len(label_mapping))
```

```
tokenizer_config.json: 100%        25.0/25.0 [00:00<00:00, 980B/s]
vocab.json: 100%                   899k/899k [00:00<00:00, 4.57MB/s]
merges.txt: 100%                   456k/456k [00:00<00:00, 2.35MB/s]
tokenizer.json: 100%               1.36M/1.36M [00:00<00:00, 6.69MB/s]
config.json: 100%                  481/481 [00:00<00:00, 19.6kB/s]
model.safetensors: 100%            499M/499M [00:05<00:00, 23.4MB/s]
Some weights of RobertaForSequenceClassification were not initialized from the model checkpoint at roberta-base and are no
You should probably TRAIN this model on a down-stream task to be able to use it for predictions and inference.
```

```
 #Tokenize data
def preprocess_function(examples):
    return tokenizer(examples['text'], truncation=True, padding=True, max_length=128)
```

```
tokenized_datasets = dataset.map(preprocess_function, batched=True)
```

```
Map: 100%                                             8000/8000 [00:00<00:00, 9404.14 examples/s]

Map: 100%                                             2000/2000 [00:00<00:00, 5310.19 examples/s]
```

```
# Define training arguments
training_args = TrainingArguments(
    output_dir='./results',  # Specify the directory for saving model outputs
    evaluation_strategy="epoch",
    save_strategy="epoch",  # Set save strategy to match evaluation strategy
    per_device_train_batch_size=8,
    per_device_eval_batch_size=8,
    num_train_epochs=3,
    weight_decay=0.01,
    save_total_limit=1,
    load_best_model_at_end=True
)
```

```
/usr/local/lib/python3.10/dist-packages/transformers/training_args.py:1568: FutureWarning: `evaluation_strategy` is depre
  warnings.warn(
```

```
# Fully Fine-tuning
def train_model(tokenized_datasets, model, training_args):
    trainer = Trainer(
        model=model,
        args=training_args,
        train_dataset=tokenized_datasets['train'],
        eval_dataset=tokenized_datasets['test'],
        tokenizer=tokenizer,
        compute_metrics=lambda p: {"accuracy": (p.predictions.argmax(-1) == p.label_ids).astype(float).mean()}
    )
    trainer.train()
    return trainer.evaluate()


# LoRa Fine-tuning
def train_model_lora(tokenized_datasets, model, training_args):
    lora_config = LoraConfig(
        r=8,
        lora_alpha=32,
        lora_dropout=0.1,
        target_modules=["query", "key"],
        bias="none",
        task_type="SEQ_CLS"
    )
    lora_model = get_peft_model(model, lora_config)
    trainer = Trainer(
        model=lora_model,
        args=training_args,
        train_dataset=tokenized_datasets['train'],
        eval_dataset=tokenized_datasets['test'],
        tokenizer=tokenizer,
        compute_metrics=lambda p: {"accuracy": (p.predictions.argmax(-1) == p.label_ids).astype(float).mean()}
    )
    trainer.train()
    return trainer.evaluate()


# Layer Freezing Fine-tuning
def train_model_layer_freezing(tokenized_datasets, model, training_args):
    # Freeze the last 6 layers
    for param in model.roberta.embeddings.parameters():
        param.requires_grad = True  # Ensure embeddings are trainable
    for layer in model.roberta.encoder.layer[:-6]:  # Freeze all layers except the last 6
        for param in layer.parameters():
            param.requires_grad = False

    trainer = Trainer(
        model=model,
        args=training_args,
        train_dataset=tokenized_datasets['train'],
        eval_dataset=tokenized_datasets['test'],
        tokenizer=tokenizer,
        compute_metrics=lambda p: {"accuracy": (p.predictions.argmax(-1) == p.label_ids).astype(float).mean()}
    )
    trainer.train()
    return trainer.evaluate()



# Train and evaluate each model
results_fully_finetuned = train_model(tokenized_datasets, model, training_args)
```

```
<ipython-input-14-eb1ec4aac905>:3: FutureWarning: `tokenizer` is deprecated and will be removed in version 5.0.0 for `Tra
  trainer = Trainer(
wandb: WARNING The `run_name` is currently set to the same value as `TrainingArguments.output_dir`. If this was not intend
wandb: Using wandb-core as the SDK backend.  Please refer to https://wandb.me/wandb-core for more information.
wandb: Logging into wandb.ai. (Learn how to deploy a W&B server locally: https://wandb.me/wandb-server)
wandb: You can find your API key in your browser here: https://wandb.ai/authorize
wandb: Paste an API key from your profile and hit enter, or press ctrl+c to quit: ··········
wandb: Appending key for api.wandb.ai to your netrc file: /root/.netrc
```

Tracking run with wandb version 0.18.6
Run data is saved locally in `/content/wandb/run-20241113_060614-k1qs06r8`
Syncing run **./results** to Weights & Biases (docs)
View project at https://wandb.ai/jani-miya/huggingface
View run at https://wandb.ai/jani-miya/huggingface/runs/k1qs06r8

[3000/3000 07:42, Epoch 3/3]

Epoch	Training Loss	Validation Loss	Accuracy
1	0.060100	0.037810	0.989000
2	0.027100	0.021287	0.991000
3	0.005600	0.011240	0.997500

[250/250 00:04]

```
results_lora = train_model_lora(tokenized_datasets, model, training_args)
```

```
<ipython-input-15-e578eb7ad04e>:12: FutureWarning: `tokenizer` is deprecated and will be removed in version 5.0.0 for `Tra
  trainer = Trainer(
```

[3000/3000 03:12, Epoch 3/3]

Epoch	Training Loss	Validation Loss	Accuracy
1	0.008800	0.012082	0.996500
2	0.003500	0.010510	0.997500
3	0.002900	0.010169	0.998000

[250/250 00:04]

```
results_layer_freezing = train_model_layer_freezing(tokenized_datasets, model, training_args)
```

```
<ipython-input-20-cddcb469e8db>:10: FutureWarning: `tokenizer` is deprecated and will be removed in version 5.0.0 for `Tra
  trainer = Trainer(
```

[3000/3000 05:20, Epoch 3/3]

Epoch	Training Loss	Validation Loss	Accuracy
1	0.006500	0.012275	0.997000
2	0.001600	0.010314	0.998500
3	0.001700	0.010023	0.998500

[250/250 00:04]

```
# Print out results
print("Fully Fine-tuned Model Results:", results_fully_finetuned)
print("LoRa Fine-tuned Model Results:", results_lora)
print("Layer Freezing Fine-tuned Model Results:", results_layer_freezing)
```

```
Fully Fine-tuned Model Results: {'eval_loss': 0.01124021876603365, 'eval_accuracy': 0.9975, 'eval_runtime': 4.5929, 'eval_
LoRa Fine-tuned Model Results: {'eval_loss': 0.010169061832129955, 'eval_accuracy': 0.998, 'eval_runtime': 5.0043, 'eval_
Layer Freezing Fine-tuned Model Results: {'eval_loss': 0.010023231618106365, 'eval_accuracy': 0.9985, 'eval_runtime': 4.8
```

Start coding or generate with AI.

Start coding or generate with AI.