Fine-Tuning LLMs with Hugging Face

Step 1: Installing and importing the libraries

```
!pip install -q accelerate==0.21.0 peft==0.4.0 bitsandbytes==0.40.2 transformers==4.31.0 trl==0.4.7
```

```
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 244.2/244.2 kB 1.9 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 72.9/72.9 kB 8.5 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 92.5/92.5 MB 6.8 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 7.4/7.4 MB 56.9 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 77.4/77.4 kB 11.8 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 7.8/7.8 MB 57.5 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 542.0/542.0 kB 49.4 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 21.3/21.3 MB 37.5 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 116.3/116.3 kB 15.8 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 194.1/194.1 kB 22.6 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 134.8/134.8 kB 16.2 MB/s eta 0:00:00
```

```
!pip install huggingface_hub
```

```
Requirement already satisfied: huggingface_hub in /usr/local/lib/python3.10/dist-packages (0.23.1)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (3.14.0)
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (2023.6.0)
Requirement already satisfied: packaging>=20.9 in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (24.0)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (6.0.1)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (2.31.0)
Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (4.66.4)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (4.11.0)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface_hub) (3.3
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface_hub) (3.7)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface_hub) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface_hub) (2024.2.2)
```

```
import torch
from trl import SFTTrainer
from peft import LoraConfig
from datasets import load_dataset
from transformers import (AutoModelForCausalLM, AutoTokenizer, BitsAndBytesConfig, TrainingArguments, pipeline)
```

Step 2: Loading the model

```
llama_model = AutoModelForCausalLM.from_pretrained(pretrained_model_name_or_path = "aboonaji/llama2finetune-v2",
                                                   quantization_config = BitsAndBytesConfig(load_in_4bit = True, bnb_4bit_compute_dtype = ge
llama_model.config.use_cache = False
llama_model.config.pretraining_tp = 1
```

```
/usr/local/lib/python3.10/dist-packages/huggingface_hub/file_download.py:1132: FutureWarning: `resume_download` is deprecated and will b
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:89: UserWarning:
The secret `HF_TOKEN` does not exist in your Colab secrets.
To authenticate with the Hugging Face Hub, create a token in your settings tab (https://huggingface.co/settings/tokens), set it as secre
You will be able to reuse this secret in all of your notebooks.
Please note that authentication is recommended but still optional to access public models or datasets.
  warnings.warn(
config.json: 100%                                  632/632 [00:00<00:00, 38.3kB/s]
pytorch_model.bin.index.json: 100%                 26.8k/26.8k [00:00<00:00, 1.82MB/s]
Downloading shards: 100%                           2/2 [01:36<00:00, 43.80s/it]
pytorch_model-00001-of-00002.bin: 100%             9.98G/9.98G [01:12<00:00, 195MB/s]
pytorch_model-00002-of-00002.bin: 100%             3.50G/3.50G [00:23<00:00, 198MB/s]
Loading checkpoint shards: 100%                    2/2 [01:05<00:00, 29.83s/it]
generation_config.json: 100%                       174/174 [00:00<00:00, 8.14kB/s]
```

Step 3: Loading the tokenizer

```
llama_tokenizer =  AutoTokenizer.from_pretrained(pretrained_model_name_or_path = "aboonaji/llama2finetune-v2" , trust_remote_code = True)
llama_tokenizer.pad_token = llama_tokenizer.eos_token
llama_tokenizer.padding_side = "right"
```

```
tokenizer_config.json:  100%                    695/695  [00:00<00:00,  50.1kB/s]

tokenizer.model:  100%                          500k/500k  [00:00<00:00,  2.22MB/s]

tokenizer.json:  100%                           1.84M/1.84M  [00:00<00:00,  30.8MB/s]

added_tokens.json:  100%                        21.0/21.0  [00:00<00:00,  1.54kB/s]

special_tokens_map.json:  100%                  435/435  [00:00<00:00,  28.5kB/s]
```

Step 4: Setting the training arguments

```
training_arguments = TrainingArguments(output_dir = "./results", per_device_train_batch_size = 4, max_steps= 100)
```

Step 5: Creating the Supervised Fine-Tuning trainer

```
llama_sft_trainer = SFTTrainer(model = llama_model,
                               args = training_arguments,
                               train_dataset = load_dataset(path = "aboonaji/wiki_medical_terms_llam2_format", split = "train"),
                               tokenizer = llama_tokenizer,
                               peft_config = LoraConfig(task_type = "CAUSAL_LM", r = 64, lora_alpha = 16, lora_dropout = 0.1),
                               dataset_text_field = "text")
```

```
Downloading data:  100%                         54.1M/54.1M  [00:00<00:00,  119MB/s]

Generating train split:  100%                   6861/6861  [00:00<00:00,  18107.88 examples/s]
/usr/local/lib/python3.10/dist-packages/peft/utils/other.py:102: FutureWarning: prepare_model_for_int8_training is deprecated and will b
  warnings.warn(
/usr/local/lib/python3.10/dist-packages/trl/trainer/sft_trainer.py:159: UserWarning: You didn't pass a `max_seq_length` argument to the
  warnings.warn(
Map:  100%                                      6861/6861  [01:01<00:00,  108.21 examples/s]
```

Step 6: Training the model

```
llama_sft_trainer.train()
```

```
[100/100 24:29, Epoch 0/1]
Step  Training Loss
TrainOutput(global_step=100, training_loss=1.6551913452148437, metrics={'train_runtime': 1485.2403, 'train_samples_per_second': 0.269,
'train_steps_per_second': 0.067, 'total_flos': 8228119310991360.0, 'train_loss': 1.6551913452148437, 'epoch': 0.06})
[100/100 24:32, Epoch 0/1]
Step  Training Loss
TrainOutput(global_step=100, training_loss=1.3250384521484375, metrics={'train_runtime': 1488.5068, 'train_samples_per_second': 0.269,
'train_steps_per_second': 0.067, 'total_flos': 8197819930705920.0, 'train_loss': 1.3250384521484375, 'epoch': 0.06})
```

Step 7: Chatting with the model

```
user_prompt = "What is Paracetomol Poisoning"
text_generation_pipeline = pipeline(task ="text-generation", model = llama_model, tokenizer = llama_tokenizer, max_length = 10000)
model_answer = text_generation_pipeline(f"<s>[INST] {user_prompt} [/INST]")
print(model_answer[0]['generated_text'])
```

```
<s>[INST] What is Paracetomol Poisoning [/INST]  Paracetamol poisoning, also known as acetaminophen poisoning, occurs when a person take

Paracetamol poisoning can cause a range of symptoms, including:
```

```
1. Nausea and vomiting
2. Abdominal pain
3. Headache
4. Dizziness and confusion
5. Yellowing of the skin and eyes (jaundice)
6. Liver damage
7. Kidney damage
8. Respiratory depression
9. Coma
10. Death

The severity of paracetamol poisoning depends on the dose taken and the time elapsed since ingestion. In severe cases, poisoning can cau

Treatment of paracetamol poisoning involves supportive care, such as intravenous fluids, oxygen therapy, and management of symptoms. In

Prevention of paracetamol poisoning is important, and this includes:

1. Following the recommended dosage on the label
2. Avoiding overdose by taking only as directed
3. Keeping medications out of reach of children and pets
4. Disposing of unused medications properly
5. Using medications only for their intended purpose
6. Monitoring medication use and seeking medical attention if symptoms occur

In conclusion, paracetamol poisoning is a serious medical condition that can cause liver damage, kidney damage, and even death. It is im
```