finetune-llama2-q-a

April 21, 2024

Name : Yatharth Thakare **PRN** : 12111403 **Roll No** : 51 **PS**: Write Python/R code to implement Transfer Learning.

[1]:
```
%%capture
%pip install accelerate peft bitsandbytes transformers trl datasets==2.16.0
```

[2]:
```
%pip install -U datasets
```

```
Requirement already satisfied: datasets in /opt/conda/lib/python3.10/site-
packages (2.16.0)
Collecting datasets
  Obtaining dependency information for datasets from https://files.pythonhosted.
org/packages/95/fc/661a7f06e8b7d48fcbd3f55423b7ff1ac3ce59526f146fda87a1e1788ee4/
datasets-2.18.0-py3-none-any.whl.metadata
  Downloading datasets-2.18.0-py3-none-any.whl.metadata (20 kB)
Requirement already satisfied: filelock in /opt/conda/lib/python3.10/site-
packages (from datasets) (3.12.2)
Requirement already satisfied: numpy>=1.17 in /opt/conda/lib/python3.10/site-
packages (from datasets) (1.24.3)
Collecting pyarrow>=12.0.0 (from datasets)
  Obtaining dependency information for pyarrow>=12.0.0 from https://files.python
hosted.org/packages/e9/0e/0d30e6fd1e0fc9cc267381520f9386a56b2b51c4066d8f9a0d4a5a
2e0b44/pyarrow-15.0.2-cp310-cp310-manylinux_2_28_x86_64.whl.metadata
  Downloading pyarrow-15.0.2-cp310-cp310-manylinux_2_28_x86_64.whl.metadata (3.0
kB)
Requirement already satisfied: pyarrow-hotfix in /opt/conda/lib/python3.10/site-
packages (from datasets) (0.6)
Requirement already satisfied: dill<0.3.9,>=0.3.0 in
/opt/conda/lib/python3.10/site-packages (from datasets) (0.3.7)
Requirement already satisfied: pandas in /opt/conda/lib/python3.10/site-packages
(from datasets) (2.0.3)
Requirement already satisfied: requests>=2.19.0 in
/opt/conda/lib/python3.10/site-packages (from datasets) (2.31.0)
Requirement already satisfied: tqdm>=4.62.1 in /opt/conda/lib/python3.10/site-
packages (from datasets) (4.66.1)
Requirement already satisfied: xxhash in /opt/conda/lib/python3.10/site-packages
(from datasets) (3.4.1)
Requirement already satisfied: multiprocess in /opt/conda/lib/python3.10/site-
```

```
packages (from datasets) (0.70.15)
Requirement already satisfied: fsspec[http]<=2024.2.0,>=2023.1.0 in
/opt/conda/lib/python3.10/site-packages (from datasets) (2023.10.0)
Requirement already satisfied: aiohttp in /opt/conda/lib/python3.10/site-
packages (from datasets) (3.8.5)
Requirement already satisfied: huggingface-hub>=0.19.4 in
/opt/conda/lib/python3.10/site-packages (from datasets) (0.19.4)
Requirement already satisfied: packaging in /opt/conda/lib/python3.10/site-
packages (from datasets) (21.3)
Requirement already satisfied: pyyaml>=5.1 in /opt/conda/lib/python3.10/site-
packages (from datasets) (6.0.1)
Requirement already satisfied: attrs>=17.3.0 in /opt/conda/lib/python3.10/site-
packages (from aiohttp->datasets) (23.1.0)
Requirement already satisfied: charset-normalizer<4.0,>=2.0 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->datasets) (3.2.0)
Requirement already satisfied: multidict<7.0,>=4.5 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->datasets) (6.0.4)
Requirement already satisfied: async-timeout<5.0,>=4.0.0a3 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->datasets) (4.0.3)
Requirement already satisfied: yarl<2.0,>=1.0 in /opt/conda/lib/python3.10/site-
packages (from aiohttp->datasets) (1.9.2)
Requirement already satisfied: frozenlist>=1.1.1 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->datasets) (1.4.0)
Requirement already satisfied: aiosignal>=1.1.2 in
/opt/conda/lib/python3.10/site-packages (from aiohttp->datasets) (1.3.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in
/opt/conda/lib/python3.10/site-packages (from huggingface-hub>=0.19.4->datasets)
(4.11.0)
Requirement already satisfied: pyparsing!=3.0.5,>=2.0.2 in
/opt/conda/lib/python3.10/site-packages (from packaging->datasets) (3.0.9)
Requirement already satisfied: idna<4,>=2.5 in /opt/conda/lib/python3.10/site-
packages (from requests>=2.19.0->datasets) (3.4)
Requirement already satisfied: urllib3<3,>=1.21.1 in
/opt/conda/lib/python3.10/site-packages (from requests>=2.19.0->datasets)
(1.26.15)
Requirement already satisfied: certifi>=2017.4.17 in
/opt/conda/lib/python3.10/site-packages (from requests>=2.19.0->datasets)
(2023.11.17)
Requirement already satisfied: python-dateutil>=2.8.2 in
/opt/conda/lib/python3.10/site-packages (from pandas->datasets) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /opt/conda/lib/python3.10/site-
packages (from pandas->datasets) (2023.3)
Requirement already satisfied: tzdata>=2022.1 in /opt/conda/lib/python3.10/site-
packages (from pandas->datasets) (2023.3)
Requirement already satisfied: six>=1.5 in /opt/conda/lib/python3.10/site-
packages (from python-dateutil>=2.8.2->pandas->datasets) (1.16.0)
Downloading datasets-2.18.0-py3-none-any.whl (510 kB)
                           510.5/510.5 kB
```

```
3.6 MB/s eta 0:00:0000:0100:01m
Downloading pyarrow-15.0.2-cp310-cp310-manylinux_2_28_x86_64.whl (38.3 MB)
                        38.3/38.3 MB
32.0 MB/s eta 0:00:00:00:0100:01
Installing collected packages: pyarrow, datasets
  Attempting uninstall: pyarrow
    Found existing installation: pyarrow 11.0.0
    Uninstalling pyarrow-11.0.0:
      Successfully uninstalled pyarrow-11.0.0
  Attempting uninstall: datasets
    Found existing installation: datasets 2.16.0
    Uninstalling datasets-2.16.0:
      Successfully uninstalled datasets-2.16.0
```

```
ERROR: pip's dependency resolver does not currently take into account all
the packages that are installed. This behaviour is the source of the following
dependency conflicts.
cudf 23.8.0 requires cupy-cuda11x>=12.0.0, which is not installed.
cuml 23.8.0 requires cupy-cuda11x>=12.0.0, which is not installed.
dask-cudf 23.8.0 requires cupy-cuda11x>=12.0.0, which is not installed.
apache-beam 2.46.0 requires dill<0.3.2,>=0.3.1.1, but you have dill 0.3.7 which
is incompatible.
apache-beam 2.46.0 requires pyarrow<10.0.0,>=3.0.0, but you have pyarrow 15.0.2
which is incompatible.
beatrix-jupyterlab 2023.814.150030 requires jupyter-server~=1.16, but you have
jupyter-server 2.12.1 which is incompatible.
beatrix-jupyterlab 2023.814.150030 requires jupyterlab~=3.4, but you have
jupyterlab 4.0.5 which is incompatible.
cudf 23.8.0 requires pandas<1.6.0dev0,>=1.3, but you have pandas 2.0.3 which is
incompatible.
cudf 23.8.0 requires protobuf<5,>=4.21, but you have protobuf 3.20.3 which is
incompatible.
cudf 23.8.0 requires pyarrow==11.*, but you have pyarrow 15.0.2 which is
incompatible.
cuml 23.8.0 requires dask==2023.7.1, but you have dask 2023.12.0 which is
incompatible.
cuml 23.8.0 requires distributed==2023.7.1, but you have distributed 2023.12.0
which is incompatible.
dask-cudf 23.8.0 requires dask==2023.7.1, but you have dask 2023.12.0 which is
incompatible.
dask-cudf 23.8.0 requires distributed==2023.7.1, but you have distributed
2023.12.0 which is incompatible.
dask-cudf 23.8.0 requires pandas<1.6.0dev0,>=1.3, but you have pandas 2.0.3
which is incompatible.
Successfully installed datasets-2.18.0 pyarrow-15.0.2
Note: you may need to restart the kernel to use updated packages.
```

[3]:
```
import torch
from datasets import load_dataset, Dataset, DatasetDict
from transformers import (
    AutoModelForCausalLM,
    AutoTokenizer,
    BitsAndBytesConfig,
    HfArgumentParser,
    TrainingArguments,
    pipeline,
    logging,
)
import bitsandbytes as bnb
from sklearn.model_selection import train_test_split
from peft import LoraConfig, PeftConfig, PeftModel, get_peft_model,␣
 ↪prepare_model_for_kbit_training
import pandas as pd
from trl import SFTTrainer
```

```
/opt/conda/lib/python3.10/site-packages/scipy/__init__.py:146: UserWarning: A
NumPy version >=1.16.5 and <1.23.0 is required for this version of SciPy
(detected version 1.24.3
  warnings.warn(f"A NumPy version >={np_minversion} and <{np_maxversion}"
```

[4]:
```
dataset = pd.read_csv('/kaggle/input/comprehensive-medical-q-a-dataset/train.
 ↪csv')
dataset = dataset.drop('qtype', axis=1)
dataset = dataset.rename(columns={'Question': 'question', 'Answer': 'answer'})
```

[5]:
```
df_full_train, df_test = train_test_split(dataset, test_size=0.2,␣
 ↪random_state=56)
df_train, df_val = train_test_split(df_full_train, test_size=0.25,␣
 ↪random_state=56)
```

[6]:
```
df_train = df_train.reset_index(drop=True)
df_val = df_train.reset_index(drop=True)
df_test = df_train.reset_index(drop=True)

train_dataset = Dataset.from_pandas(df_train)
val_dataset = Dataset.from_pandas(df_val)
test_dataset = Dataset.from_pandas(df_test)
```

[7]:
```
health_dataset_dict = DatasetDict({
    'train': train_dataset,
    'validation': val_dataset,
    'test': test_dataset
})
```

[8]:
```
# Define a function to transform the data
def transform_conversation(example):
    conversation_text = example['question']
    conversation_answer = example['answer']

    reformatted_segments = []

    if conversation_answer:
        reformatted_segments.append(f'<s>[INST] {conversation_text} [/INST]␣
 ↪{conversation_answer} </s>')

    else:
        reformatted_segments.append(f'<s>[INST] {conversation_text} [/INST] </
 ↪s>')
    return {'text': ''.join(reformatted_segments)}
```

[9]:
```
transformed_dataset = health_dataset_dict.map(transform_conversation)
transformed_dataset
```

```
Map:   0%|          | 0/9843 [00:00<?, ? examples/s]
Map:   0%|          | 0/9843 [00:00<?, ? examples/s]
Map:   0%|          | 0/9843 [00:00<?, ? examples/s]
```

[9]:
```
DatasetDict({
    train: Dataset({
        features: ['question', 'answer', 'text'],
        num_rows: 9843
    })
    validation: Dataset({
        features: ['question', 'answer', 'text'],
        num_rows: 9843
    })
    test: Dataset({
        features: ['question', 'answer', 'text'],
        num_rows: 9843
    })
})
```

[10]:
```
# Model from Hugging Face hub
base_model = "NousResearch/Llama-2-7b-chat-hf"

# Fine-tuned model
new_model = "llama-2-7b-chat-health"
```

[11]:
```
compute_dtype = getattr(torch, "float16")
```

```
quant_config = BitsAndBytesConfig(
    load_in_4bit=True,
    bnb_4bit_use_double_quant=True,
    bnb_4bit_quant_type="nf4",
    bnb_4bit_compute_dtype=torch.bfloat16
)
```

[12]:
```
model = AutoModelForCausalLM.from_pretrained(
    base_model,
    device_map="auto",
    trust_remote_code=True,
    quantization_config=quant_config
)
model.config.use_cache = False
model.config.pretraining_tp = 1
```

```
config.json:   0%|          | 0.00/583 [00:00<?, ?B/s]
model.safetensors.index.json:   0%|          | 0.00/26.8k [00:00<?, ?B/s]
Downloading shards:   0%|          | 0/2 [00:00<?, ?it/s]
model-00001-of-00002.safetensors:   0%|          | 0.00/9.98G [00:00<?, ?B/s]
model-00002-of-00002.safetensors:   0%|          | 0.00/3.50G [00:00<?, ?B/s]
Loading checkpoint shards:   0%|          | 0/2 [00:00<?, ?it/s]
generation_config.json:   0%|          | 0.00/179 [00:00<?, ?B/s]
/opt/conda/lib/python3.10/site-
packages/transformers/generation/configuration_utils.py:389: UserWarning:
`do_sample` is set to `False`. However, `temperature` is set to `0.9` -- this
flag is only used in sample-based generation modes. You should set
`do_sample=True` or unset `temperature`. This was detected when initializing the
generation config instance, which means the corresponding file may hold
incorrect parameterization and should be fixed.
  warnings.warn(
/opt/conda/lib/python3.10/site-
packages/transformers/generation/configuration_utils.py:394: UserWarning:
`do_sample` is set to `False`. However, `top_p` is set to `0.6` -- this flag is
only used in sample-based generation modes. You should set `do_sample=True` or
unset `top_p`. This was detected when initializing the generation config
instance, which means the corresponding file may hold incorrect parameterization
and should be fixed.
  warnings.warn(
```

[13]:
```
tokenizer = AutoTokenizer.from_pretrained(base_model, trust_remote_code=True)
tokenizer.pad_token = tokenizer.eos_token
```

```
tokenizer_config.json:   0%|          | 0.00/746 [00:00<?, ?B/s]
```

```
tokenizer.model:   0%|          | 0.00/500k [00:00<?, ?B/s]
tokenizer.json:   0%|          | 0.00/1.84M [00:00<?, ?B/s]
added_tokens.json:   0%|          | 0.00/21.0 [00:00<?, ?B/s]
special_tokens_map.json:   0%|          | 0.00/435 [00:00<?, ?B/s]
```

[14]:
```
model = prepare_model_for_kbit_training(model)
```

[15]:
```
peft_params = LoraConfig(
    lora_alpha=16,
    lora_dropout=0.05,
    r=2,
    bias="none",
    task_type="CAUSAL_LM",
)
```

[16]:
```
training_params = TrainingArguments(
    output_dir="/kaggle/working/results",
    num_train_epochs=1,
    per_device_train_batch_size=8,
    gradient_accumulation_steps=4,
    optim="paged_adamw_32bit",
    save_steps=25,
    logging_steps=25,
    learning_rate=2e-4,
    weight_decay=0.001,
    fp16=False,
    max_steps=-1,
    warmup_ratio=0.03,
    group_by_length=True,
    lr_scheduler_type="constant",
    report_to="tensorboard"
)
```

[17]:
```
trainer = SFTTrainer(
    model=model,
    train_dataset=transformed_dataset['train'],
    peft_config=peft_params,
    dataset_text_field="text",
    max_seq_length=None,
    tokenizer=tokenizer,
    args=training_params,
    packing=False,
)
```

```
/opt/conda/lib/python3.10/site-packages/trl/trainer/sft_trainer.py:246:
UserWarning: You didn't pass a `max_seq_length` argument to the SFTTrainer, this
```

```
will default to 1024
  warnings.warn(
```

```
Map:   0%|          | 0/9843 [00:00<?, ? examples/s]
```

```
/opt/conda/lib/python3.10/site-packages/trl/trainer/sft_trainer.py:318:
UserWarning: You passed a tokenizer with `padding_side` not equal to `right` to
the SFTTrainer. This might lead to some unexpected behaviour due to overflow
issues when training a model in half-precision. You might consider adding
`tokenizer.padding_side = 'right'` to your code.
  warnings.warn(
```

[18]:
```
# Free GPU memory
torch.cuda.empty_cache()
```

[19]:
```
print('Training is starting.....')
```

```
Training is starting…
```

[ ]:
```
# Train model
trainer.train()

# Save trained model
trainer.model.save_pretrained(new_model)
trainer.tokenizer.save_pretrained(new_model)
```

```
You're using a LlamaTokenizerFast tokenizer. Please note that with a fast
tokenizer, using the `__call__` method is faster than using a method to encode
the text followed by a call to the `pad` method to get a padded encoding.
```

```
<IPython.core.display.HTML object>
```

[ ]:
```
prompt = "What is bacteria?"
pipe = pipeline(task="text-generation", model=model, tokenizer=tokenizer,␣
 ↪max_length=200)
result = pipe(f"<s>[INST] {prompt} [/INST]")
print(result[0]['generated_text'])
```

[ ]:
```

```