HW4

Task 1

1.1

(1) Transformer includes encoders and decoders. The encoder encodes the input texts and allows them to be converted into vectors or other forms for processing. The decoder generates the output in sequence-to-sequence tasks.

The encoder and decoder contain self-attention layers and feedforward neural networks. The self-attention layer is used to help each word in a sentence interact with every other word to capture contextual information efficiently. The feedforward neural network processes the information obtained from the attention layers.

(2) Intrinsic Evaluation: The quality of the generated text is judged by comparing the attributes of the generated text and the target text, such as fluency, internal correlation, correctness, etc.

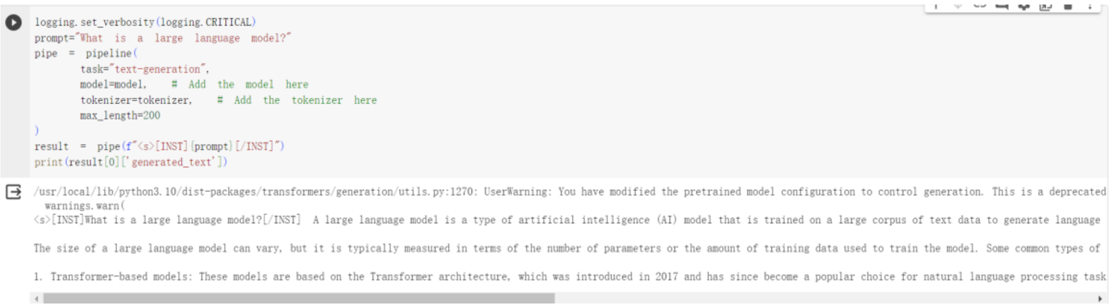
Extrinsic Evaluation: Evaluate the performance of the generated text on downstream subtasks to determine its effectiveness. It is equivalent to judging the quality of the text based on its application in specific scenarios.

Distinction: Intrinsic evaluation focuses solely on a specific language task, whereas extrinsic evaluation measures performance in real-world application contexts.

(3) Token is the smallest unit of the model’s input. Tokens can be individual words or characters.

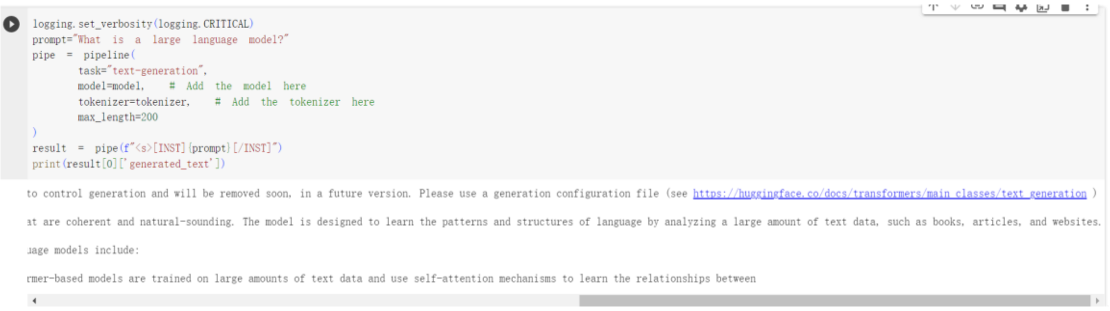
1.2

(1) When max_length = 200



```
logging.set_verbosity(logging.CRITICAL)
prompt="What  is  a  large  language  model?"
pipe  =  pipeline(
        task="text-generation",
        model=model,    #  Add  the  model  here
        tokenizer=tokenizer,    #  Add  the  tokenizer  here
        max_length=200
)
result  =  pipe(f"<s>[INST]{prompt}[/INST]")
print(result[0]['generated_text'])
```

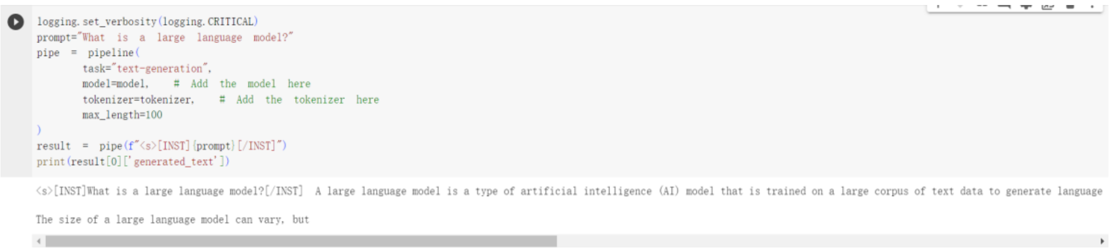
```
/usr/local/lib/python3.10/dist-packages/transformers/generation/utils.py:1270: UserWarning: You have modified the pretrained model configuration to control generation. This is a deprecated
  warnings.warn(
<s>[INST]What is a large language model?[/INST]  A large language model is a type of artificial intelligence (AI) model that is trained on a large corpus of text data to generate language

The size of a large language model can vary, but it is typically measured in terms of the number of parameters or the amount of training data used to train the model. Some common types of

1. Transformer-based models: These models are based on the Transformer architecture, which was introduced in 2017 and has since become a popular choice for natural language processing task
```



```
logging.set_verbosity(logging.CRITICAL)
prompt="What  is  a  large  language  model?"
pipe  =  pipeline(
        task="text-generation",
        model=model,    #  Add  the  model  here
        tokenizer=tokenizer,    #  Add  the  tokenizer  here
        max_length=200
)
result  =  pipe(f"<s>[INST]{prompt}[/INST]")
print(result[0]['generated_text'])
```

```
to control generation and will be removed soon, in a future version. Please use a generation configuration file (see https://huggingface.co/docs/transformers/main_classes/text_generation )

at are coherent and natural-sounding. The model is designed to learn the patterns and structures of language by analyzing a large amount of text data, such as books, articles, and websites.

uage models include:

rmer-based models are trained on large amounts of text data and use self-attention mechanisms to learn the relationships between
```

(2) When max_length = 100

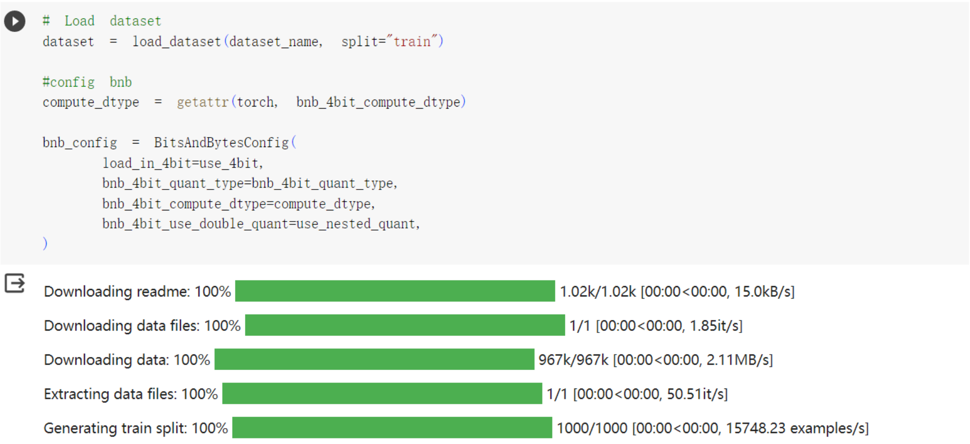


```
logging.set_verbosity(logging.CRITICAL)
prompt="What  is  a  large  language  model?"
pipe  =  pipeline(
        task="text-generation",
        model=model,    #  Add  the  model  here
        tokenizer=tokenizer,    #  Add  the  tokenizer  here
        max_length=100
)
result  =  pipe(f"<s>[INST]{prompt}[/INST]")
print(result[0]['generated_text'])
```

```
<s>[INST]What is a large language model?[/INST]  A large language model is a type of artificial intelligence (AI) model that is trained on a large corpus of text data to generate language

The size of a large language model can vary, but
```

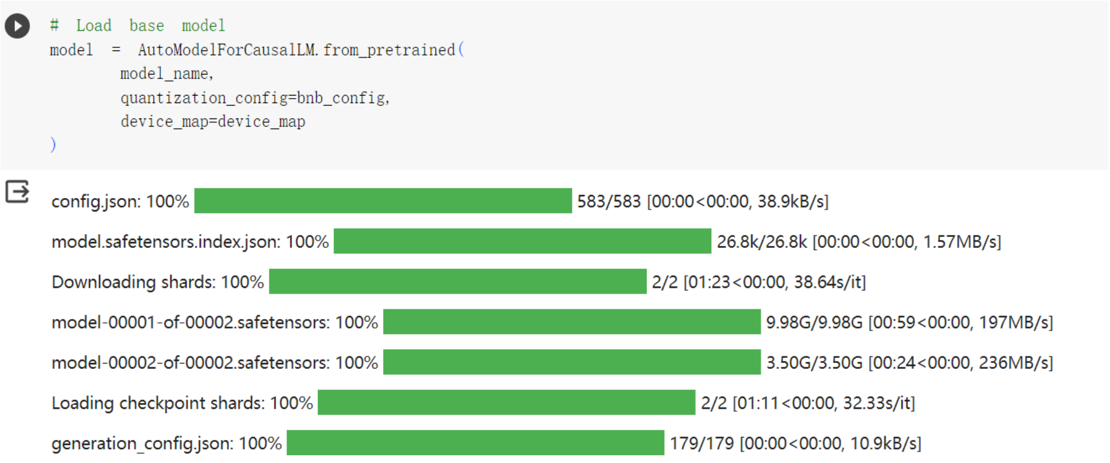
Task2

2.1



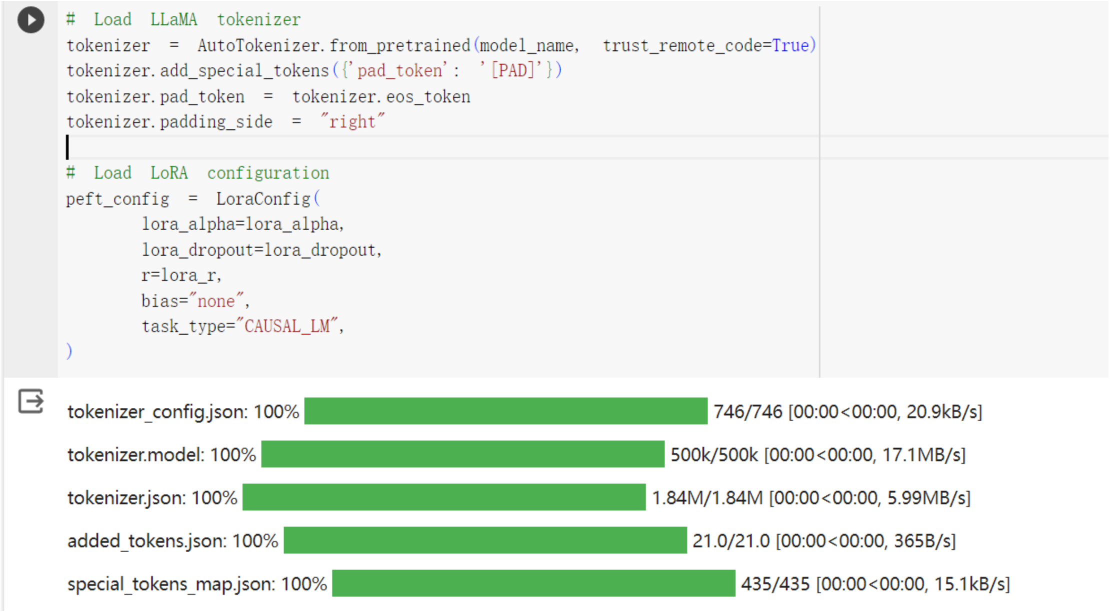
```
#  Load  dataset
dataset  =  load_dataset(dataset_name,  split="train")

#config  bnb
compute_dtype  =  getattr(torch,  bnb_4bit_compute_dtype)

bnb_config  =  BitsAndBytesConfig(
        load_in_4bit=use_4bit,
        bnb_4bit_quant_type=bnb_4bit_quant_type,
        bnb_4bit_compute_dtype=compute_dtype,
        bnb_4bit_use_double_quant=use_nested_quant,
)
```

```
Downloading readme: 100% 1.02k/1.02k [00:00<00:00, 15.0kB/s]
Downloading data files: 100% 1/1 [00:00<00:00, 1.85it/s]
Downloading data: 100% 967k/967k [00:00<00:00, 2.11MB/s]
Extracting data files: 100% 1/1 [00:00<00:00, 50.51it/s]
Generating train split: 100% 1000/1000 [00:00<00:00, 15748.23 examples/s]
```



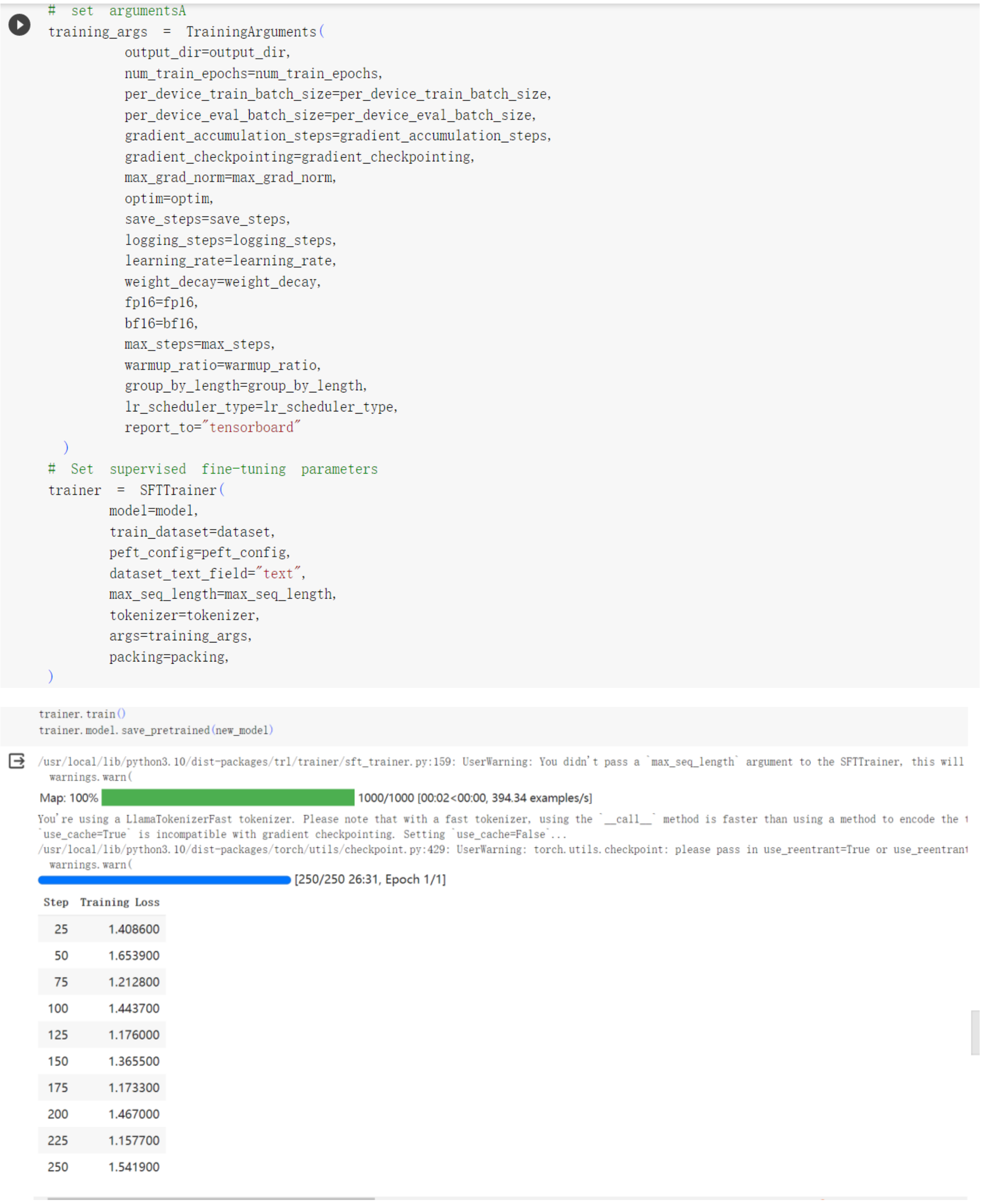
```
# Load base model
model = AutoModelForCausalLM.from_pretrained(
        model_name,
        quantization_config=bnb_config,
        device_map=device_map
)
```

```
config.json: 100%  583/583 [00:00<00:00, 38.9kB/s]
model.safetensors.index.json: 100%  26.8k/26.8k [00:00<00:00, 1.57MB/s]
Downloading shards: 100%  2/2 [01:23<00:00, 38.64s/it]
model-00001-of-00002.safetensors: 100%  9.98G/9.98G [00:59<00:00, 197MB/s]
model-00002-of-00002.safetensors: 100%  3.50G/3.50G [00:24<00:00, 236MB/s]
Loading checkpoint shards: 100%  2/2 [01:11<00:00, 32.33s/it]
generation_config.json: 100%  179/179 [00:00<00:00, 10.9kB/s]
```



```
# Load LLaMA tokenizer
tokenizer = AutoTokenizer.from_pretrained(model_name, trust_remote_code=True)
tokenizer.add_special_tokens({'pad_token': '[PAD]'})
tokenizer.pad_token = tokenizer.eos_token
tokenizer.padding_side = "right"

# Load LoRA configuration
peft_config = LoraConfig(
        lora_alpha=lora_alpha,
        lora_dropout=lora_dropout,
        r=lora_r,
        bias="none",
        task_type="CAUSAL_LM",
)
```

```
tokenizer_config.json: 100%  746/746 [00:00<00:00, 20.9kB/s]
tokenizer.model: 100%  500k/500k [00:00<00:00, 17.1MB/s]
tokenizer.json: 100%  1.84M/1.84M [00:00<00:00, 5.99MB/s]
added_tokens.json: 100%  21.0/21.0 [00:00<00:00, 365B/s]
special_tokens_map.json: 100%  435/435 [00:00<00:00, 15.1kB/s]
```

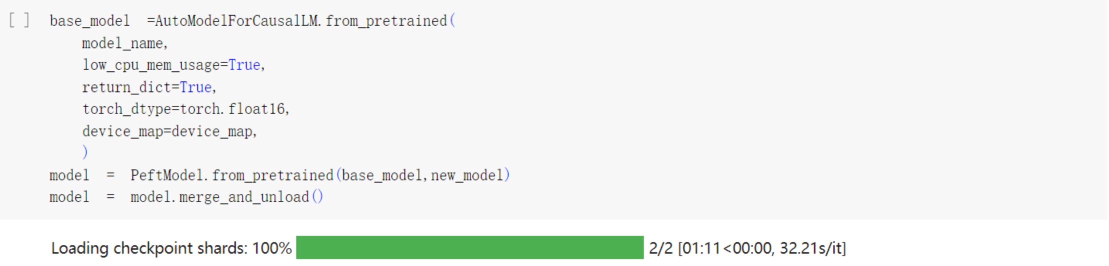


```
# set argumentsA
training_args = TrainingArguments(
            output_dir=output_dir,
            num_train_epochs=num_train_epochs,
            per_device_train_batch_size=per_device_train_batch_size,
            per_device_eval_batch_size=per_device_eval_batch_size,
            gradient_accumulation_steps=gradient_accumulation_steps,
            gradient_checkpointing=gradient_checkpointing,
            max_grad_norm=max_grad_norm,
            optim=optim,
            save_steps=save_steps,
            logging_steps=logging_steps,
            learning_rate=learning_rate,
            weight_decay=weight_decay,
            fp16=fp16,
            bf16=bf16,
            max_steps=max_steps,
            warmup_ratio=warmup_ratio,
            group_by_length=group_by_length,
            lr_scheduler_type=lr_scheduler_type,
            report_to="tensorboard"
  )
# Set supervised fine-tuning parameters
trainer = SFTTrainer(
          model=model,
          train_dataset=dataset,
          peft_config=peft_config,
          dataset_text_field="text",
          max_seq_length=max_seq_length,
          tokenizer=tokenizer,
          args=training_args,
          packing=packing,
)
```

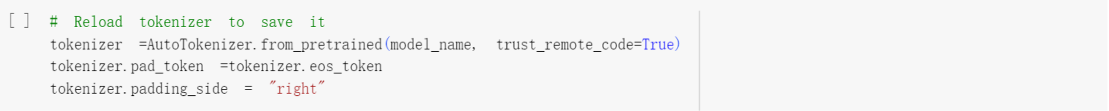
```
trainer.train()
trainer.model.save_pretrained(new_model)
```

```
/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 SFTTrainer, this will
  warnings.warn(
Map: 100% 1000/1000 [00:02<00:00, 394.34 examples/s]
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
`use_cache=True` is incompatible with gradient checkpointing. Setting `use_cache=False`...
/usr/local/lib/python3.10/dist-packages/torch/utils/checkpoint.py:429: UserWarning: torch.utils.checkpoint: please pass in use_reentrant=True or use_reentran
  warnings.warn(
 [250/250 26:31, Epoch 1/1]
Step  Training Loss
  25       1.408600
  50       1.653900
  75       1.212800
 100       1.443700
 125       1.176000
 150       1.365500
 175       1.173300
 200       1.467000
 225       1.157700
 250       1.541900
```

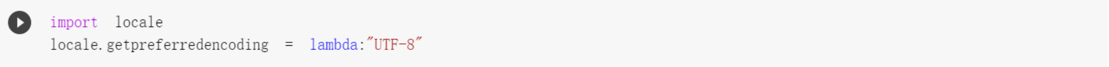
Reload model



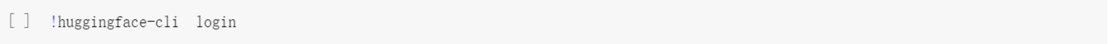
```
[ ] base_model  =AutoModelForCausalLM.from_pretrained(
        model_name,
        low_cpu_mem_usage=True,
        return_dict=True,
        torch_dtype=torch.float16,
        device_map=device_map,
        )
    model  =  PeftModel.from_pretrained(base_model,new_model)
    model  =  model.merge_and_unload()

    Loading checkpoint shards: 100% 2/2 [01:11<00:00, 32.21s/it]
```



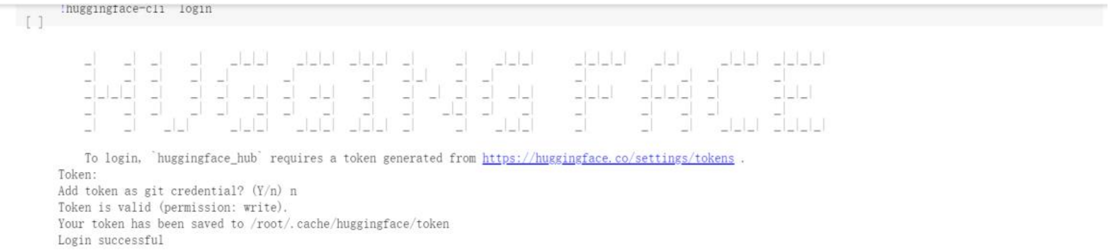
```
[ ] #  Reload  tokenizer  to  save  it
    tokenizer  =AutoTokenizer.from_pretrained(model_name,  trust_remote_code=True)
    tokenizer.pad_token  =tokenizer.eos_token
    tokenizer.padding_side  =  "right"
```



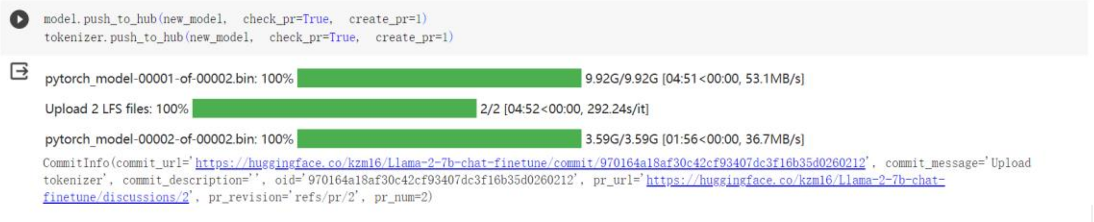
```
import  locale
locale.getpreferredencoding  =  lambda:"UTF-8"
```



```
[ ] !huggingface-cli  login
```

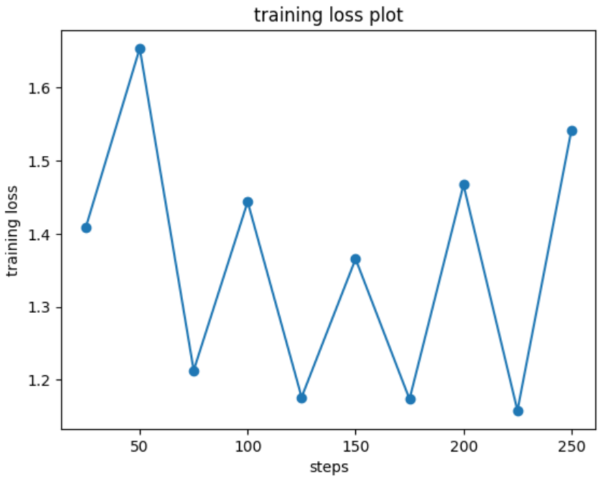


```
    !huggingface-cli  login
[ ]

    To login, `huggingface_hub` requires a token generated from https://huggingface.co/settings/tokens .
Token:
Add token as git credential? (Y/n) n
Token is valid (permission: write).
Your token has been saved to /root/.cache/huggingface/token
Login successful
```

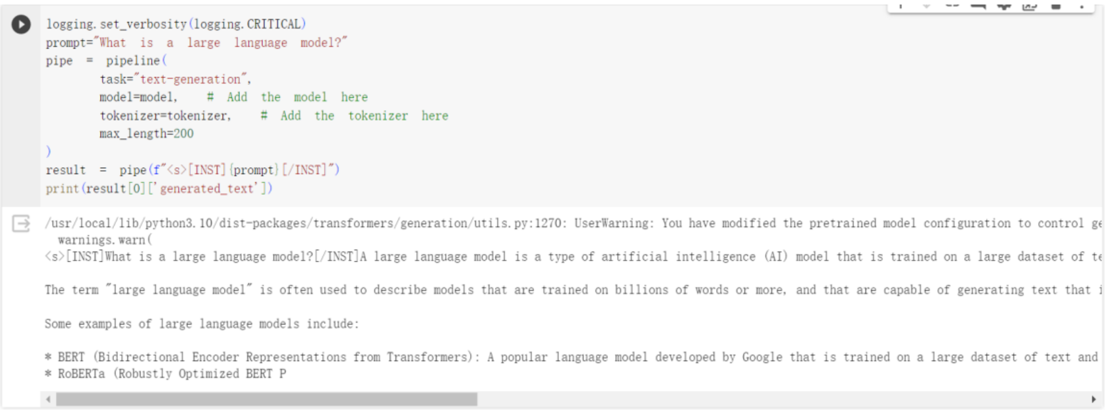


```
model.push_to_hub(new_model,  check_pr=True,  create_pr=1)
tokenizer.push_to_hub(new_model,  check_pr=True,  create_pr=1)

pytorch_model-00001-of-00002.bin: 100% 9.92G/9.92G [04:51<00:00, 53.1MB/s]
Upload 2 LFS files: 100% 2/2 [04:52<00:00, 292.24s/it]
pytorch_model-00002-of-00002.bin: 100% 3.59G/3.59G [01:56<00:00, 36.7MB/s]
CommitInfo(commit_url='https://huggingface.co/kzm16/Llama-2-7b-chat-finetune/commit/970164a18af30c42cf93407dc3f16b35d0260212', commit_message='Upload
tokenizer', commit_description='', oid='970164a18af30c42cf93407dc3f16b35d0260212', pr_url='https://huggingface.co/kzm16/Llama-2-7b-chat-
finetune/discussions/2', pr_revision='refs/pr/2', pr_num=2)
```

2.2 The training loss plot

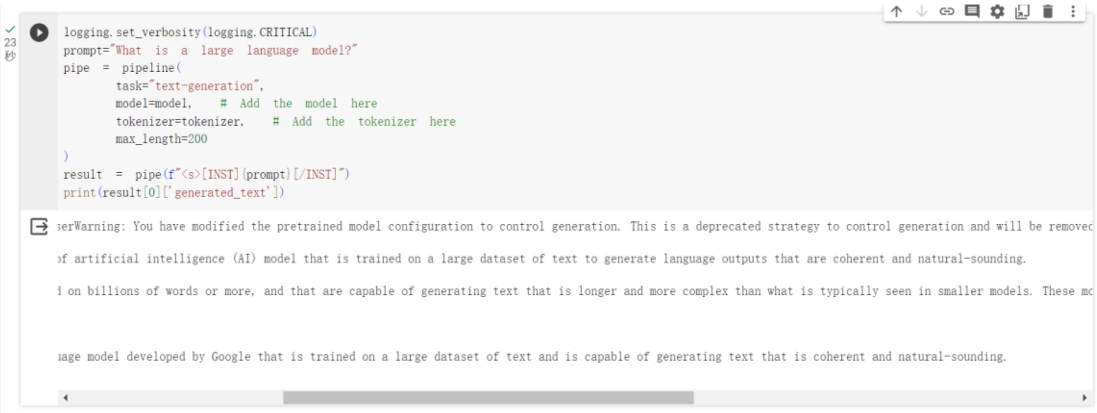


2.3 Generate text basing on fine-tuned model



```
logging.set_verbosity(logging.CRITICAL)
prompt="What  is  a  large  language  model?"
pipe  =  pipeline(
        task="text-generation",
        model=model,    #  Add  the  model  here
        tokenizer=tokenizer,    #  Add  the  tokenizer  here
        max_length=200
)
result  =  pipe(f"<s>[INST]{prompt}[/INST]")
print(result[0]['generated_text'])
```

```
/usr/local/lib/python3.10/dist-packages/transformers/generation/utils.py:1270: UserWarning: You have modified the pretrained model configuration to control ge
  warnings.warn(
<s>[INST]What is a large language model?[/INST]A large language model is a type of artificial intelligence (AI) model that is trained on a large dataset of te

The term "large language model" is often used to describe models that are trained on billions of words or more, and that are capable of generating text that i

Some examples of large language models include:

* BERT (Bidirectional Encoder Representations from Transformers): A popular language model developed by Google that is trained on a large dataset of text and
* RoBERTa (Robustly Optimized BERT P
```

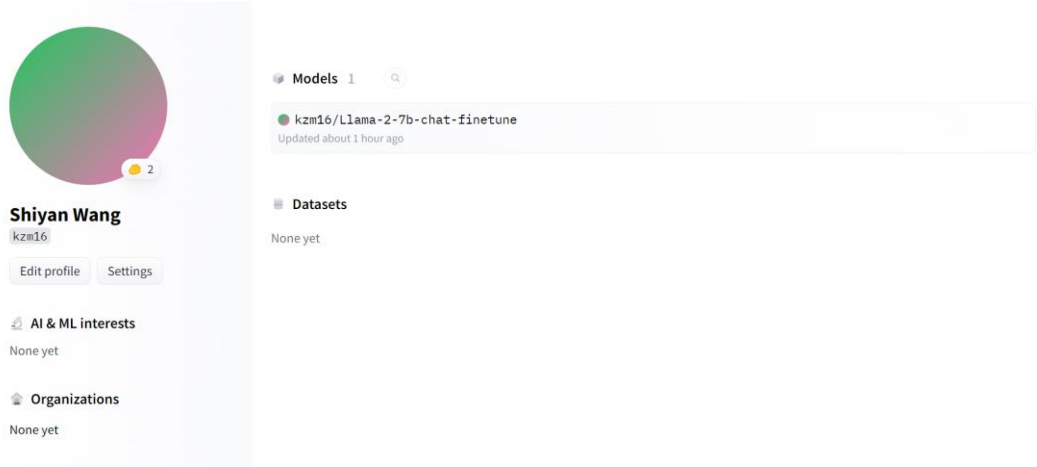


23 秒

```
logging.set_verbosity(logging.CRITICAL)
prompt="What  is  a  large  language  model?"
pipe  =  pipeline(
        task="text-generation",
        model=model,    #  Add  the  model  here
        tokenizer=tokenizer,    #  Add  the  tokenizer  here
        max_length=200
)
result  =  pipe(f"<s>[INST]{prompt}[/INST]")
print(result[0]['generated_text'])
```

```
serWarning: You have modified the pretrained model configuration to control generation. This is a deprecated strategy to control generation and will be removec

of artificial intelligence (AI) model that is trained on a large dataset of text to generate language outputs that are coherent and natural-sounding.

i on billions of words or more, and that are capable of generating text that is longer and more complex than what is typically seen in smaller models. These mc



iage model developed by Google that is trained on a large dataset of text and is capable of generating text that is coherent and natural-sounding.
```

2.4 Hugging face page



Shiyan Wang
kzm16

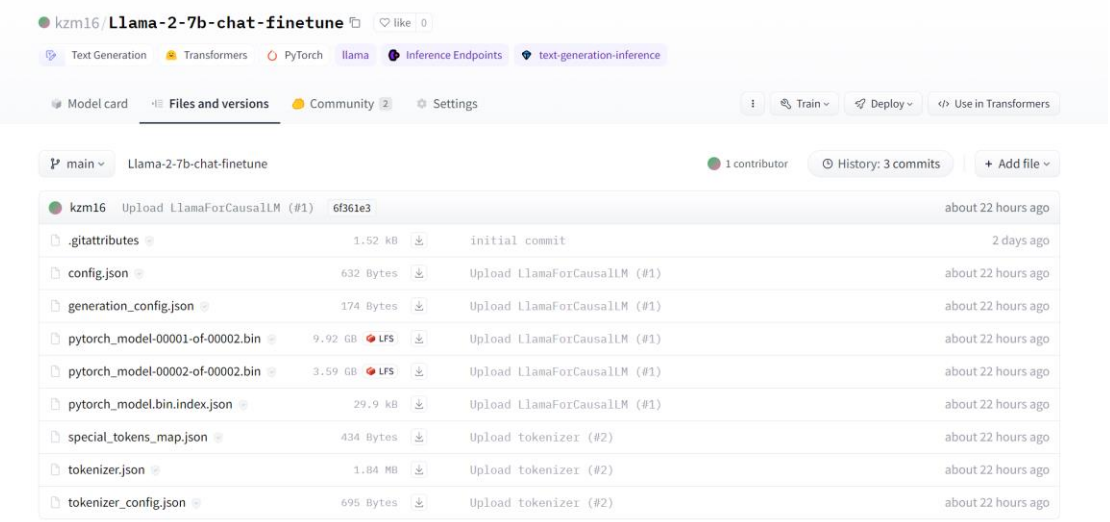
AI & ML interests
None yet

Organizations
None yet

Models 1

kzm16/Llama-2-7b-chat-finetune
Updated about 1 hour ago

Datasets
None yet

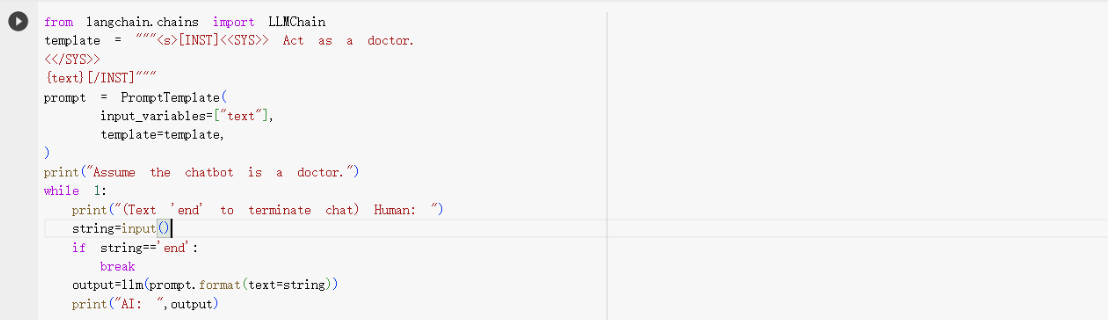


kzm16 Upload LlamaForCausalLM (#1) 6f361e3 about 22 hours ago

.gitattributes	1.52 kB	initial commit	2 days ago
config.json	632 Bytes	Upload LlamaForCausalLM (#1)	about 22 hours ago
generation_config.json	174 Bytes	Upload LlamaForCausalLM (#1)	about 22 hours ago
pytorch_model-00001-of-00002.bin	9.92 GB LFS	Upload LlamaForCausalLM (#1)	about 22 hours ago
pytorch_model-00002-of-00002.bin	3.59 GB LFS	Upload LlamaForCausalLM (#1)	about 22 hours ago
pytorch_model.bin.index.json	29.9 kB	Upload LlamaForCausalLM (#1)	about 22 hours ago
special_tokens_map.json	434 Bytes	Upload tokenizer (#2)	about 22 hours ago
tokenizer.json	1.84 MB	Upload tokenizer (#2)	about 22 hours ago
tokenizer_config.json	695 Bytes	Upload tokenizer (#2)	about 22 hours ago

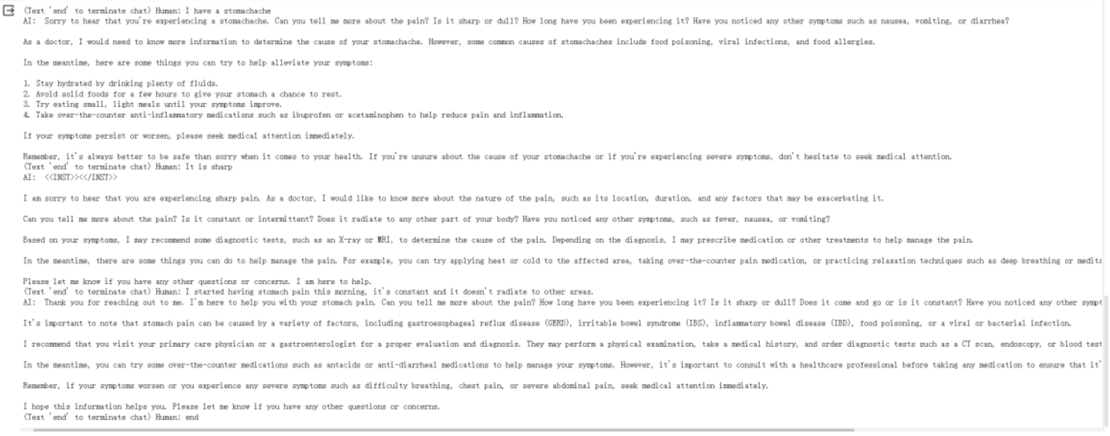
Task 3

3.1 The code for prompt is:



```
from langchain.chains import LLMChain
template = """<s>[INST]<<SYS>> Act as a doctor.
<</SYS>>
{text} [/INST]"""
prompt = PromptTemplate(
        input_variables=["text"],
        template=template,
)
print("Assume the chatbot is a doctor.")
while 1:
    print("(Text 'end' to terminate chat) Human: ")
    string=input()
    if string=='end':
        break
    output=llm(prompt.format(text=string))
    print("AI: ",output)
```

3.2 Test the chatbot



```
(Text 'end' to terminate chat) Human: I have a stomachache
AI:  Sorry to hear that you're experiencing a stomachache. Can you tell me more about the pain? Is it sharp or dull? How long have you been experiencing it? Have you noticed any other symptoms such as nausea, vomiting, or diarrhea?

As a doctor, I would need to know more information to determine the cause of your stomachache. However, some common causes of stomachaches include food poisoning, viral infections, and food allergies.

In the meantime, here are some things you can try to help alleviate your symptoms:

1. Stay hydrated by drinking plenty of fluids.
2. Avoid solid foods for a few hours to give your stomach a chance to rest.
3. Try eating small, light meals until your symptoms improve.
4. Take over-the-counter anti-inflammatory medications such as ibuprofen or acetaminophen to help reduce pain and inflammation.

If your symptoms persist or worsen, please seek medical attention immediately.

Remember, it's always better to be safe than sorry when it comes to your health. If you're unsure about the cause of your stomachache or if you're experiencing severe symptoms, don't hesitate to seek medical attention.
(Text 'end' to terminate chat) Human: It is sharp
AI:  <<INST>><</INST>>

I am sorry to hear that you are experiencing sharp pain. As a doctor, I would like to know more about the nature of the pain, such as its location, duration, and any factors that may be exacerbating it.

Can you tell me more about the pain? Is it constant or intermittent? Does it radiate to any other part of your body? Have you noticed any other symptoms, such as fever, nausea, or vomiting?

Based on your symptoms, I may recommend some diagnostic tests, such as an X-ray or MRI, to determine the cause of the pain. Depending on the diagnosis, I may prescribe medication or other treatments to help manage the pain.

In the meantime, there are some things you can do to help manage the pain. For example, you can try applying heat or cold to the affected area, taking over-the-counter pain medication, or practicing relaxation techniques such as deep breathing or medit

Please let me know if you have any other questions or concerns. I am here to help.
(Text 'end' to terminate chat) Human: I started having stomach pain this morning, it's constant and it doesn't radiate to other areas.
AI:  Thank you for reaching out to me. I'm here to help you with your stomach pain. Can you tell me more about the pain? How long have you been experiencing it? Is it sharp or dull? Does it come and go or is it constant? Have you noticed any other sympt

It's important to note that stomach pain can be caused by a variety of factors, including gastroesophageal reflux disease (GERD), irritable bowel syndrome (IBS), inflammatory bowel disease (IBD), food poisoning, or a viral or bacterial infection.

I recommend that you visit your primary care physician or a gastroenterologist for a proper evaluation and diagnosis. They may perform a physical examination, take a medical history, and order diagnostic tests such as a CT scan, endoscopy, or blood test

In the meantime, you can try some over-the-counter medications such as antacids or anti-diarrheal medications to help manage your symptoms. However, it's important to consult with a healthcare professional before taking any medication to ensure that it'

Remember, if your symptoms worsen or you experience any severe symptoms such as difficulty breathing, chest pain, or severe abdominal pain, seek medical attention immediately.

I hope this information helps you. Please let me know if you have any other questions or concerns.
(Text 'end' to terminate chat) Human: end
```