Double-click (or enter) to edit

```
!pip install transformers accelerate torch

```

```
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 883.7/883.7 kB 23.5 MB/s eta 0:00:00
Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl (664.8 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 664.8/664.8 MB 2.9 MB/s eta 0:00:00
Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl (211.5 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 211.5/211.5 MB 5.0 MB/s eta 0:00:00
Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.whl (56.3 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 56.3/56.3 MB 17.8 MB/s eta 0:00:00
Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.whl (127.9 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 127.9/127.9 MB 8.7 MB/s eta 0:00:00
Downloading nvidia_cusparse_cu12-12.3.1.170-py3-none-manylinux2014_x86_64.whl (207.5 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 207.5/207.5 MB 6.8 MB/s eta 0:00:00
Downloading nvidia_nccl_cu12-2.21.5-py3-none-manylinux2014_x86_64.whl (188.7 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 188.7/188.7 MB 5.6 MB/s eta 0:00:00
Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl (21.1 MB)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 21.1/21.1 MB 35.8 MB/s eta 0:00:00
Installing collected packages: nvidia-nvjitlink-cu12, nvidia-nccl-cu12, nvidia-curand-cu12, nvidia-cufft-cu12, nvidia-cuda-runtime-cu1
  Attempting uninstall: nvidia-nvjitlink-cu12
    Found existing installation: nvidia-nvjitlink-cu12 12.5.82
    Uninstalling nvidia-nvjitlink-cu12-12.5.82:
      Successfully uninstalled nvidia-nvjitlink-cu12-12.5.82
  Attempting uninstall: nvidia-nccl-cu12
    Found existing installation: nvidia-nccl-cu12 2.23.4
    Uninstalling nvidia-nccl-cu12-2.23.4:
      Successfully uninstalled nvidia-nccl-cu12-2.23.4
  Attempting uninstall: nvidia-curand-cu12
    Found existing installation: nvidia-curand-cu12 10.3.6.82
    Uninstalling nvidia-curand-cu12-10.3.6.82:
      Successfully uninstalled nvidia-curand-cu12-10.3.6.82
  Attempting uninstall: nvidia-cufft-cu12
    Found existing installation: nvidia-cufft-cu12 11.2.3.61
    Uninstalling nvidia-cufft-cu12-11.2.3.61:
      Successfully uninstalled nvidia-cufft-cu12-11.2.3.61
  Attempting uninstall: nvidia-cuda-runtime-cu12
    Found existing installation: nvidia-cuda-runtime-cu12 12.5.82
    Uninstalling nvidia-cuda-runtime-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-runtime-cu12-12.5.82
  Attempting uninstall: nvidia-cuda-nvrtc-cu12
    Found existing installation: nvidia-cuda-nvrtc-cu12 12.5.82
    Uninstalling nvidia-cuda-nvrtc-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-nvrtc-cu12-12.5.82
  Attempting uninstall: nvidia-cuda-cupti-cu12
    Found existing installation: nvidia-cuda-cupti-cu12 12.5.82
    Uninstalling nvidia-cuda-cupti-cu12-12.5.82:
      Successfully uninstalled nvidia-cuda-cupti-cu12-12.5.82
  Attempting uninstall: nvidia-cublas-cu12
    Found existing installation: nvidia-cublas-cu12 12.5.3.2
    Uninstalling nvidia-cublas-cu12-12.5.3.2:
      Successfully uninstalled nvidia-cublas-cu12-12.5.3.2
  Attempting uninstall: nvidia-cusparse-cu12
    Found existing installation: nvidia-cusparse-cu12 12.5.1.3
    Uninstalling nvidia-cusparse-cu12-12.5.1.3:
      Successfully uninstalled nvidia-cusparse-cu12-12.5.1.3
  Attempting uninstall: nvidia-cudnn-cu12
    Found existing installation: nvidia-cudnn-cu12 9.3.0.75
    Uninstalling nvidia-cudnn-cu12-9.3.0.75:
      Successfully uninstalled nvidia-cudnn-cu12-9.3.0.75
ERROR: Operation cancelled by user
```

```
!pip -q install -U transformers accelerate sentencepiece

```

```
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 42.0/42.0 kB 3.3 MB/s eta 0:00:00
     ━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━━ 11.3/11.3 MB 86.3 MB/s eta 0:00:00
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
    WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
    WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
```

```
import torch, platform
print("PyTorch:", torch.__version__)
print("CUDA available:", torch.cuda.is_available())
```

```
if torch.cuda.is_available():
    print("GPU:", torch.cuda.get_device_name(0))
print("Python:", platform.python_version())

```

```
PyTorch: 2.6.0+cu124
CUDA available: True
GPU: Tesla T4
Python: 3.11.13
```

```
!pip -q install -U "transformers>=4.44" "accelerate>=0.34"

```

```
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
WARNING: Ignoring invalid distribution ~vidia-cudnn-cu12 (/usr/local/lib/python3.11/dist-packages)
```

1. Error Loading

```
from transformers import pipeline, AutoTokenizer
import torch

def main():
    try:
        model_id = "TinyLlama/TinyLlama-1.1B-Chat-v1.0"

        print("Loading tokenizer...")
        tok = AutoTokenizer.from_pretrained(model_id, use_fast=True)

        prompt = "write a poem about I am Just a Girl."
        messages = [{"role": "system", "content": prompt}]

        # Build the input text using the model's chat template if present.
        try:
            input_text = tok.apply_chat_template(
                messages,
                tokenize=False,
                add_generation_prompt=True
            )
        except Exception as e:
            print(f" Chat template not available, falling back to raw prompt. Details: {e}")
            input_text = prompt  # fallback

        use_gpu = torch.cuda.is_available()
        dtype = torch.float16 if use_gpu else torch.float32

        print("⚙️ Loading model & pipeline...")
        try:
            generator = pipeline(
                "text-generation",
                model=model_id,
                tokenizer=tok,
                device=0 if use_gpu else -1,  # GPU if available
                torch_dtype=dtype,
            )
        except Exception as e:
            print(f" Error loading pipeline: {e}")
            return

        print(" Generating text...")
        try:
            out = generator(
                input_text,
                max_new_tokens=300,      # keep modest for speed/RAM
                do_sample=True,
                temperature=0.8,
                top_p=0.9,
                repetition_penalty=1.2,
                pad_token_id=tok.eos_token_id if tok.eos_token_id is not None else None,
                eos_token_id=tok.eos_token_id if tok.eos_token_id is not None else None,
                return_full_text=False,
                truncation=True,
            )
            print("\n Output:\n")
            print(out[0]["generated_text"])
```

```
        except Exception as e:
            print(f"❌ Error during generation: {e}")

    except Exception as e:
        print(f"💥 Unexpected error: {e}")

if __name__ == "__main__":
    main()

```

```
Loading tokenizer...
/usr/local/lib/python3.11/dist-packages/huggingface_hub/utils/_auth.py:94: 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(
tokenizer_config.json: 1.29k/? [00:00<00:00, 36.9kB/s]
tokenizer.model: 100% 500k/500k [00:00<00:00, 607kB/s]
tokenizer.json: 1.84M/? [00:00<00:00, 19.5MB/s]
special_tokens_map.json: 100% 551/551 [00:00<00:00, 11.6kB/s]
⚙️ Loading model & pipeline...
config.json: 100% 608/608 [00:00<00:00, 17.7kB/s]
model.safetensors: 100% 2.20G/2.20G [00:32<00:00, 140MB/s]
generation_config.json: 100% 124/124 [00:00<00:00, 7.09kB/s]
Device set to use cuda:0
 Generating text...

 Output:

I Am Just a Girl, with all my heart and soul,
Without a doubt or hesitation,
Facing life's challenges head-on each day;
Growing in ways that make me proud to be known.

From the smallest of things, to the biggest of dreams,
I strive for excellence and always try to excel higher.
The world is full of wonders, but they are nothing compared to me,
For every step I take, every word spoken,
My spirit soars high above and beyond.

As an individual, there is no one like me,
Each part of me unique and lovely too.
There is something within, waiting to unfold,
And though it may take time, I know where I belong.

Through thick and thin, through good times and bad,
I hold on tightly to my inner strength.
It's not easy, but I never give up or lose sight,
Of what makes me who I truly am, unstoppable.

So here I stand, a girl from earth,
A little bit different yet just as brave,
Believing in myself and everything around,
I am Just a Girl, with all her might.

Oh, I Am Just a Girl, living life boldly too!
```

2. Text Generation

```
from transformers import pipeline, AutoTokenizer
import torch

model_id = "TinyLlama/TinyLlama-1.1B-Chat-v1.0"

# Load tokenizer
tok = AutoTokenizer.from_pretrained(model_id)

# Messages in chat format
messages = [
    {"role": "system", "content": "You are a creative storyteller."},
    {"role": "user", "content": "Write a short story about a girl who dreams of touching the stars."}
]

```

```
# Convert messages to model input using the chat template
input_text = tok.apply_chat_template(messages, tokenize=False, add_generation_prompt=True)

# Load pipeline
generator = pipeline(
    "text-generation",
    model=model_id,
    tokenizer=tok,
    device=0 if torch.cuda.is_available() else -1,
    torch_dtype=torch.float16 if torch.cuda.is_available() else torch.float32
)

# Generate output
out = generator(
    input_text,
    max_new_tokens=300,
    do_sample=True,
    temperature=0.8,
    top_p=0.9,
    repetition_penalty=1.1,
    return_full_text=False
)

print("📝 Generated Story:\n")
print(out[0]["generated_text"])

```

```
Device set to use cuda:0
📝 Generated Story:

Sarah had always been fascinated by the night sky, ever since she was a little girl. She would spend hours stargazing, imagining herself

One evening, as she walked home from school, Sarah saw a shooting star streak across the sky. It was the most breathtaking sight she had

She ran home as fast as she could, opening the window to let in the cool breeze. As she stepped outside, she felt a sense of exhilaratic

Sarah closed her eyes and took a deep breath, then stepped into the street. A warm breeze blew against her face, and the sound of rustli

As she drew nearer to the stars, she felt a sudden jolt of electricity, like a shockwave coursing through her body. Her heart raced as s

But just as she reached out, something happened. A bright light shone
```

Streaming Response

```
from transformers import AutoTokenizer, AutoModelForCausalLM, TextIteratorStreamer
import torch
from threading import Thread

model_id = "TinyLlama/TinyLlama-1.1B-Chat-v1.0"

# Load model & tokenizer
tokenizer = AutoTokenizer.from_pretrained(model_id)
model = AutoModelForCausalLM.from_pretrained(
    model_id,
    torch_dtype=torch.float16 if torch.cuda.is_available() else torch.float32,
    device_map="auto"
)

# Chat messages
messages = [
    {"role": "system", "content": "You are a creative storyteller."},
    {"role": "user", "content": "Write a short story about a girl who dreams of touching the stars."}
]

# Build chat input
input_text = tokenizer.apply_chat_template(messages, tokenize=False, add_generation_prompt=True)
inputs = tokenizer(input_text, return_tensors="pt").to(model.device)

# Setup streaming
streamer = TextIteratorStreamer(tokenizer, skip_prompt=True, skip_special_tokens=True)

# Launch generation in a background thread
generation_kwargs = dict(
    **inputs,
    max_new_tokens=300,
    do_sample=True,
```

```
    temperature=0.8,
    top_p=0.9,
    repetition_penalty=1.1,
    streamer=streamer
)
thread = Thread(target=model.generate, kwargs=generation_kwargs)
thread.start()

# Print tokens as they stream in
print("📝 Streaming story:\n")
for new_text in streamer:
    print(new_text, end="", flush=True)

thread.join()

```

```
📝 Streaming story:

Lena had always been fascinated by the cosmos, and one summer evening, she found herself staring at the sky with a sense of wonder. She

As she walked, she felt a sudden surge of energy in her muscles, as if her body was awakening from a deep sleep. Lena opened her eyes an

She stood there for a while, lost in thought. Suddenly, she heard a faint sound coming from the direction of the horizon. Curious, she t

Lena ran after it, her heart pounding in her chest as she caught sight of it moving further away. She could feel a wave of excitement wa

Without thinking, she pulled out her phone and began scrolling through her social media feed. She found a few articles about space explo
```