

```
1 !pip install transformers
2 !pip install datasets
3 !pip install huggingface_hub
```

```
Requirement already satisfied: requests<2.32.2 in /usr/local/lib/python3.10/dist-packages (from datasets) (2.32.2)
Requirement already satisfied: tqdm>=4.66.3 in /usr/local/lib/python3.10/dist-packages (from datasets) (4.66.5)
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 (from datasets)
  Downloading multiprocess-0.70.16-py310-none-any.whl.metadata (7.2 kB)
Requirement already satisfied: fsspec<=2024.6.1,>=2023.1.0 in /usr/local/lib/python3.10/dist-packages (from fsspec[http]<=2024.6.1,>=2023.1.0->datasets) (2024.6.1)
Requirement already satisfied: aiohttp in /usr/local/lib/python3.10/dist-packages (from datasets) (3.10.5)
Requirement already satisfied: huggingface-hub>=0.22.0 in /usr/local/lib/python3.10/dist-packages (from datasets) (0.24.7)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from datasets) (24.1)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from datasets) (6.0.2)
Requirement already satisfied: aiohappyeyeballs>=2.3.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (2.4.0)
Requirement already satisfied: aiosignal>=1.1.2 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.3.1)
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) (1.4.1)
Requirement already satisfied: multidict<7.0,>=4.5 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (6.1.0)
Requirement already satisfied: yarl<2.0,>=1.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (1.11.1)
Requirement already satisfied: async-timeout<5.0,>=4.0 in /usr/local/lib/python3.10/dist-packages (from aiohttp->datasets) (4.0.3)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.22.0->datasets) (4.12.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests>=2.32.2->datasets) (2024.8.30)
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2.8.2)
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.2)
Requirement already satisfied: tzdata>=2022.1 in /usr/local/lib/python3.10/dist-packages (from pandas->datasets) (2024.1)
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.10/dist-packages (from python-dateutil>=2.8.2->pandas->datasets) (1.16.0)
Downloading datasets-3.0.0-py3-none-any.whl (474 kB)
```

Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface\_hub) (3.10)  
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) (2024.8.30)

```
1 from huggingface_hub import notebook_login
2
3 notebook_login()
```



Copy a token from [your Hugging Face tokens page](#) and paste it below.

Immediately click login after copying your token or it might be stored in plain text in this notebook file.


Token:

☒ Add token as git credential?

Login


**Pro Tip:** If you don't already have one, you can create a dedicated 'notebooks' token with 'write' access, that you can then easily reuse for all notebooks.

```
1 from transformers import pipeline
2
3 # Create a question-answering pipeline
4 qa_pipeline = pipeline('question-answering', model='deepset/roberta-base-squad2')
5
6 # Define a context and a question
7 context = """
8 RoBERTa is a robustly optimized BERT pretraining approach. RoBERTa improves on BERT by training on more data and removing some of the pretraining objectives.
9 """
10 question = "What does RoBERTa do?"
11
12 # Get the answer
13 response = qa_pipeline(question=question, context=context)
14
15 # Print the answer
16 print(response['answer'])
17
```


 `/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 secret in your Google Colab and restart your session.  
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.

<code>warnings.warn(</code>	
<code>config.json: 100%</code>	<code>571/571 [00:00&lt;00:00, 13.3kB/s]</code>
<code>model.safetensors: 100%</code>	<code>496M/496M [00:07&lt;00:00, 28.0MB/s]</code>
<code>tokenizer_config.json: 100%</code>	<code>79.0/79.0 [00:00&lt;00:00, 4.86kB/s]</code>
<code>vocab.json: 100%</code>	<code>899k/899k [00:00&lt;00:00, 14.3MB/s]</code>
<code>merges.txt: 100%</code>	<code>456k/456k [00:00&lt;00:00, 6.83MB/s]</code>
<code>special_tokens_map.json: 100%</code>	<code>772/772 [00:00&lt;00:00, 55.9kB/s]</code>

`/usr/local/lib/python3.10/dist-packages/transformers/tokenization_utils_base.py:1601: FutureWarning: `clean_up_tokenization_spaces` was not set. It will be set to `True` by default. This b`  
`warnings.warn(`  
Hardware accelerator e.g. GPU is available in the environment, but no ``device`` argument is passed to the ``Pipeline`` object. Model will be on CPU.  
training on more data and removing some of the pretraining objectives




```
1 from transformers import pipeline
2
3 # Create a summarization pipeline
4 summarization_pipeline = pipeline('summarization', model='facebook/bart-large-cnn')
5
6 # Define text to summarize
7 text = "The transformers library provides thousands of pre-trained models to perform tasks on texts such as classification, information extraction, question answering, summarization, transl
8
9 # Summarize the text
10 response = summarization_pipeline(text)
11
12 # Print the summarization result
13 print(response)
```

 `config.json: 100%`

<code>config.json: 100%</code>	<code>1.58k/1.58k [00:00&lt;00:00, 91.8kB/s]</code>
<code>model.safetensors: 100%</code>	<code>1.63G/1.63G [00:13&lt;00:00, 98.5MB/s]</code>
<code>generation_config.json: 100%</code>	<code>363/363 [00:00&lt;00:00, 11.6kB/s]</code>
<code>vocab.json: 100%</code>	<code>899k/899k [00:00&lt;00:00, 35.7MB/s]</code>
<code>merges.txt: 100%</code>	<code>456k/456k [00:00&lt;00:00, 7.29MB/s]</code>
<code>tokenizer.json: 100%</code>	<code>1.36M/1.36M [00:00&lt;00:00, 27.7MB/s]</code>

`Hardware accelerator e.g. GPU is available in the environment, but no `device` argument is passed to the `Pipeline` object. Model will be on CPU.`  
`Your max_length is set to 142, but your input_length is only 37. Since this is a summarization task, where outputs shorter than the input are typically wanted, you might consider decreasin`  
`[{'summary_text': 'The transformers library provides thousands of pre-trained models to perform tasks on texts. These tasks include classification, information extraction, question answeri`



```
1 from transformers import pipeline
2
3 # Create a summarization pipeline
```

```

4 summarizer = pipeline('summarization', model='facebook/bart-large-cnn')
5
6 # Define a long piece of text
7 text = """
8 Facebook AI has developed BART, a model that combines the strengths of both bidirectional and auto-regressive transformers. BART uses a bidirectional encoder like BERT, which understands co
9 """
10
11 # Get the summary
12 summary = summarizer(text, max_length=100, min_length=30, do_sample=False)
13
14 # Print the summary
15 print(summary[0]['summary_text'])

```

Hardware accelerator e.g. GPU is available in the environment, but no `device` argument is passed to the `Pipeline` object. Model will be on CPU.  
 BART is a model that combines the strengths of both bidirectional and auto-regressive transformers. BART can be applied to various NLP tasks including text generation, summarization, and t

```

1 from transformers import pipeline
2
3 # Load translation model
4 translator = pipeline('translation_en_to_fr')
5
6 # Interactive input
7 text = input("Enter a sentence in English to translate to French: ")
8
9 # Translate text
10 response = translator(text)
11
12 print(f"Translation: {response[0]['translation_text']}")

```

No model was supplied, defaulted to google-t5/t5-base and revision 686f1db (<https://huggingface.co/google-t5/t5-base>).  
 Using a pipeline without specifying a model name and revision in production is not recommended.

config.json: 100% 1.21k/1.21k [00:00<00:00, 89.7kB/s]

model.safetensors: 100% 892M/892M [00:07<00:00, 192MB/s]

generation\_config.json: 100% 147/147 [00:00<00:00, 8.65kB/s]

spiece.model: 100% 792k/792k [00:00<00:00, 12.9MB/s]

tokenizer.json: 100% 1.39M/1.39M [00:00<00:00, 7.26MB/s]

Hardware accelerator e.g. GPU is available in the environment, but no `device` argument is passed to the `Pipeline` object. Model will be on CPU.

Enter a sentence in English to translate to French: hi my name is rehan

Translation: Bonjour, mon nom est rehan

```
1 !pip install huggingface_hub
```

Requirement already satisfied: huggingface\_hub in /usr/local/lib/python3.10/dist-packages (0.24.7)  
 Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from huggingface\_hub) (3.16.1)  
 Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface\_hub) (2024.6.1)  
 Requirement already satisfied: packaging>=20.9 in /usr/local/lib/python3.10/dist-packages (from huggingface\_hub) (24.1)  
 Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from huggingface\_hub) (6.0.2)  
 Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from huggingface\_hub) (2.32.3)  
 Requirement already satisfied: tqdm>=4.42.1 in /usr/local/lib/python3.10/dist-packages (from huggingface\_hub) (4.66.5)  
 Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface\_hub) (4.12.2)

```
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface_hub) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->huggingface_hub) (3.10)
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) (2024.8.30)
```

```
1 import os
2
3 # Replace 'your_hugging_face_here' with your actual token # ANOTHER WAY OF LOGGING THROUGH HUGGING FACE.
4 os.environ['HF_TOKEN'] = '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'
```

```
1 !pip install --upgrade transformers huggingface_hub
```

```
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-packages (4.44.2)
Requirement already satisfied: huggingface_hub in /usr/local/lib/python3.10/dist-packages (0.25.1)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from transformers) (3.16.1)
Requirement already satisfied: numpy>=1.17 in /usr/local/lib/python3.10/dist-packages (from transformers) (1.26.4)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from transformers) (24.1)
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) (2024.9.11)
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<0.20,>=0.19 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.19.1)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-packages (from transformers) (4.66.5)
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (2024.6.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface_hub) (4.12.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->transformers) (2024.8.30)
```

```
1 !pip install diffusers transformers
2 !pip install -U ftfy regex tqdm
3 !pip install -U git+https://github.com/openai/CLIP.git
```

```
Requirement already satisfied: diffusers in /usr/local/lib/python3.10/dist-packages (0.30.3)
Requirement already satisfied: transformers in /usr/local/lib/python3.10/dist-packages (4.44.2)
Requirement already satisfied: importlib-metadata in /usr/local/lib/python3.10/dist-packages (from diffusers) (8.5.0)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from diffusers) (3.16.1)
Requirement already satisfied: huggingface-hub>=0.23.2 in /usr/local/lib/python3.10/dist-packages (from diffusers) (0.25.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from diffusers) (1.26.4)
Requirement already satisfied: regex!=2019.12.17 in /usr/local/lib/python3.10/dist-packages (from diffusers) (2024.9.11)
Requirement already satisfied: requests in /usr/local/lib/python3.10/dist-packages (from diffusers) (2.32.3)
Requirement already satisfied: safetensors>=0.3.1 in /usr/local/lib/python3.10/dist-packages (from diffusers) (0.4.5)
Requirement already satisfied: Pillow in /usr/local/lib/python3.10/dist-packages (from diffusers) (10.4.0)
Requirement already satisfied: packaging>=20.0 in /usr/local/lib/python3.10/dist-packages (from transformers) (24.1)
Requirement already satisfied: pyyaml>=5.1 in /usr/local/lib/python3.10/dist-packages (from transformers) (6.0.2)
Requirement already satisfied: tokenizers<0.20,>=0.19 in /usr/local/lib/python3.10/dist-packages (from transformers) (0.19.1)
Requirement already satisfied: tqdm>=4.27 in /usr/local/lib/python3.10/dist-packages (from transformers) (4.66.5)
Requirement already satisfied: fsspec>=2023.5.0 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.23.2->diffusers) (2024.6.1)
Requirement already satisfied: typing-extensions>=3.7.4.3 in /usr/local/lib/python3.10/dist-packages (from huggingface-hub>=0.23.2->diffusers) (4.12.2)
Requirement already satisfied: zipp>=3.20 in /usr/local/lib/python3.10/dist-packages (from importlib-metadata->diffusers) (3.20.2)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.10/dist-packages (from requests->diffusers) (3.3.2)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.10/dist-packages (from requests->diffusers) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.10/dist-packages (from requests->diffusers) (2.0.7)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.10/dist-packages (from requests->diffusers) (2024.8.30)
Collecting ftfy
  Using cached ftfy-6.2.3-py3-none-any.whl.metadata (7.8 kB)
```

```

Requirement already satisfied: regex in /usr/local/lib/python3.10/dist-packages (2024.9.11)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (4.66.5)
Requirement already satisfied: wcwidth<0.3.0,>=0.2.12 in /usr/local/lib/python3.10/dist-packages (from ftfy) (0.2.13)
Using cached ftfy-6.2.3-py3-none-any.whl (43 kB)
Installing collected packages: ftfy
Successfully installed ftfy-6.2.3
Collecting git+https://github.com/openai/CLIP.git
  Cloning https://github.com/openai/CLIP.git to /tmp/pip-req-build-6p1pg4gj
  Running command git clone --filter=blob:none --quiet https://github.com/openai/CLIP.git /tmp/pip-req-build-6p1pg4gj
  Resolved https://github.com/openai/CLIP.git to commit dcba3cb2e2827b402d2701e7e1c7d9fed8a20ef1
  Preparing metadata (setup.py) ... done
Requirement already satisfied: ftfy in /usr/local/lib/python3.10/dist-packages (from clip==1.0) (6.2.3)
Requirement already satisfied: packaging in /usr/local/lib/python3.10/dist-packages (from clip==1.0) (24.1)
Requirement already satisfied: regex in /usr/local/lib/python3.10/dist-packages (from clip==1.0) (2024.9.11)
Requirement already satisfied: tqdm in /usr/local/lib/python3.10/dist-packages (from clip==1.0) (4.66.5)
Requirement already satisfied: torch in /usr/local/lib/python3.10/dist-packages (from clip==1.0) (2.4.1+cu121)
Requirement already satisfied: torchvision in /usr/local/lib/python3.10/dist-packages (from clip==1.0) (0.19.1+cu121)
Requirement already satisfied: wcwidth<0.3.0,>=0.2.12 in /usr/local/lib/python3.10/dist-packages (from ftfy->clip==1.0) (0.2.13)
Requirement already satisfied: filelock in /usr/local/lib/python3.10/dist-packages (from torch->clip==1.0) (3.16.1)
Requirement already satisfied: typing-extensions>=4.8.0 in /usr/local/lib/python3.10/dist-packages (from torch->clip==1.0) (4.12.2)
Requirement already satisfied: sympy in /usr/local/lib/python3.10/dist-packages (from torch->clip==1.0) (1.13.2)
Requirement already satisfied: networkx in /usr/local/lib/python3.10/dist-packages (from torch->clip==1.0) (3.3)
Requirement already satisfied: Jinja2 in /usr/local/lib/python3.10/dist-packages (from torch->clip==1.0) (3.1.4)
Requirement already satisfied: fsspec in /usr/local/lib/python3.10/dist-packages (from torchvision->clip==1.0) (2024.6.1)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from torchvision->clip==1.0) (1.26.4)
Requirement already satisfied: pillow!=8.3.*,>=5.3.0 in /usr/local/lib/python3.10/dist-packages (from torchvision->clip==1.0) (10.4.0)
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.10/dist-packages (from Jinja2->torch->clip==1.0) (2.1.5)
Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.10/dist-packages (from sympy->torch->clip==1.0) (1.3.0)
Building wheels for collected packages: clip
  Building wheel for clip (setup.py) ... done
  Created wheel for clip: filename=clip-1.0-py3-none-any.whl size=1369490 sha256=bf7915fa229e2d318b7e56842a8f8936ec547ad85d96d07d5fab299c01959487
  Stored in directory: /tmp/pip-ephem-wheel-cache-ijtolno3/wheels/da/2b/4c/d6691fa9597aac8bb85d2ac13b112deb897d5b50f5ad9a37e4
Successfully built clip
Installing collected packages: clip
Successfully installed clip-1.0

```

```

1 import torch
2 from diffusers import StableDiffusionPipeline
3 from PIL import Image
4
5 # Load the Stable Diffusion model
6 pipe = StableDiffusionPipeline.from_pretrained("CompVis/stable-diffusion-v1-4", torch_dtype=torch.float16)
7 pipe = pipe.to("cuda")
8
9 # Generate an image
10 prompt = "A futuristic cityscape"
11 image = pipe(prompt).images[0]
12
13 # Save the image
14 image.save("generated_image.png") # Save with the correct file name
15
16 # Display the image
17 image.show()
18
19 # Display the image in Google Colab
20 from IPython.display import Image as IPyImage # to display on colab.
21 IPyImage("generated_image.png") # Reference the correct file name

```



```
➦ /usr/local/lib/python3.10/dist-packages/huggingface_hub/utils/_token.py:90: 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 secret in your Google Colab and restart your session.
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(
model_index.json: 100%                               541/541 [00:00<00:00, 27.6kB/s]
Fetching 16 files: 100%                             16/16 [00:31<00:00, 1.97s/it]
model.safetensors: 100%                             492M/492M [00:10<00:00, 76.5MB/s]
(...)ature_extractor/preprocessor_config.json: 100%   342/342 [00:00<00:00, 3.03kB/s]
tokenizer/merges.txt: 100%                           525k/525k [00:00<00:00, 3.22MB/s]
scheduler/scheduler_config.json: 100%                313/313 [00:00<00:00, 2.93kB/s]
text_encoder/config.json: 100%                      592/592 [00:00<00:00, 6.91kB/s]
model.safetensors: 100%                             1.22G/1.22G [00:22<00:00, 115MB/s]
safety_checker/config.json: 100%                    4.56k/4.56k [00:00<00:00, 60.3kB/s]
(...)kpoints/scheduler_config-checkpoint.json: 100%  209/209 [00:00<00:00, 1.44kB/s]
tokenizer/vocab.json: 100%                          1.06M/1.06M [00:00<00:00, 4.27MB/s]
tokenizer/tokenizer_config.json: 100%                806/806 [00:00<00:00, 8.11kB/s]
diffusion_pytorch_model.safetensors: 100%           3.44G/3.44G [00:31<00:00, 243MB/s]
tokenizer/special_tokens_map.json: 100%              472/472 [00:00<00:00, 4.86kB/s]
UNET/config.json: 100%                              743/743 [00:00<00:00, 9.09kB/s]
vae/config.json: 100%                               551/551 [00:00<00:00, 10.9kB/s]
diffusion_pytorch_model.safetensors: 100%           335M/335M [00:07<00:00, 56.1MB/s]
Loading pipeline components...: 100%                 7/7 [00:19<00:00, 3.71s/it]
/usr/local/lib/python3.10/dist-packages/transformers/tokenization_utils_base.py:1601: FutureWarning: `clean_up_tokenization_spaces` was not set. It will be set to `True` by default. This
  warnings.warn(
100%                                                  50/50 [00:07<00:00, 7.66it/s]
```



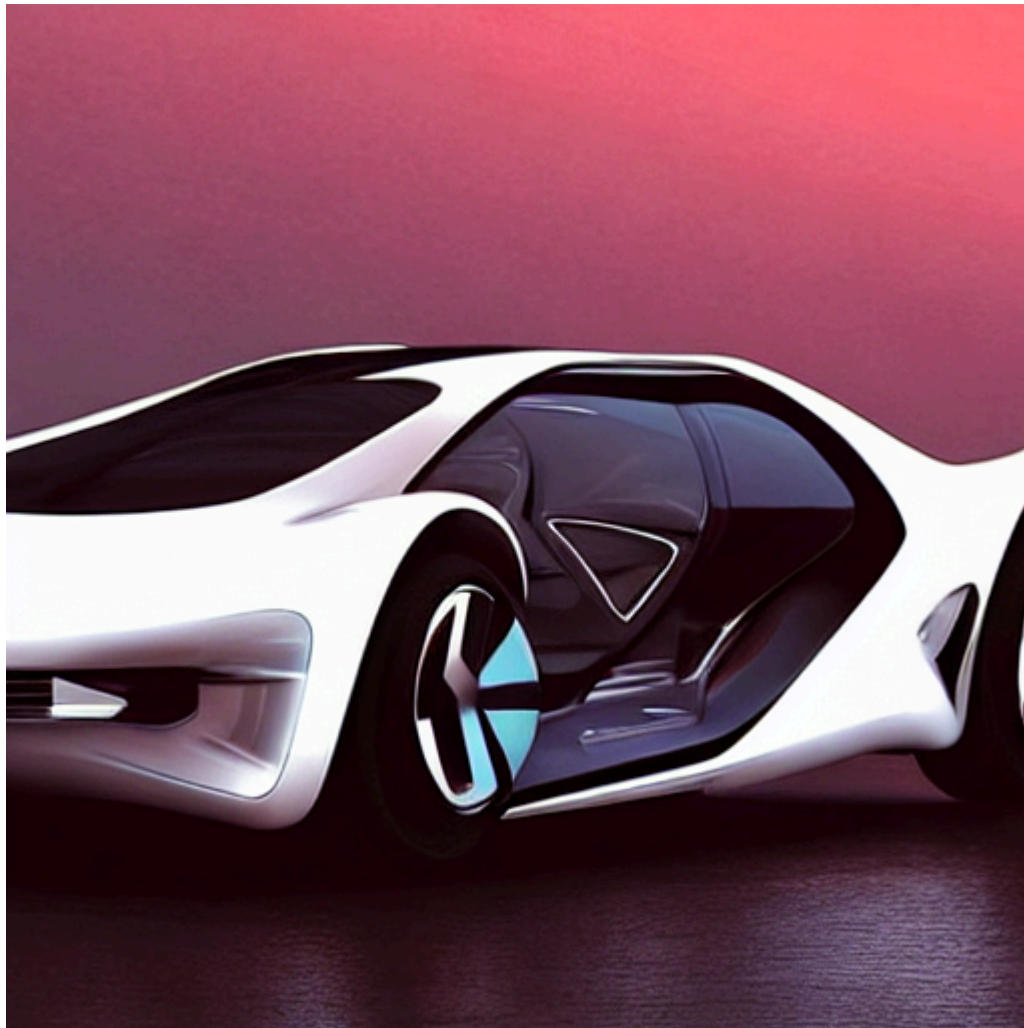


```
1 import torch
2 from diffusers import StableDiffusionPipeline
3 from PIL import Image
4
5 # Load the Stable Diffusion model
6 pipe = StableDiffusionPipeline.from_pretrained("CompVis/stable-diffusion-v1-4", torch_dtype=torch.float16)
7 pipe = pipe.to("cuda")
8
9 # Generate an image
10 prompt = "A futuristic car"
11 image = pipe(prompt).images[0]
12
13 # Save the image
14 image.save("generated_image.png") # Save with the correct file name
15
16 # Display the image
17 image.show()
18
19 # Display the image in Google Colab
20 from IPython.display import Image as IPyImage # to display on colab.
21 IPyImage("generated_image.png") # Reference the correct file name
```



↔ Loading pipeline components...: 100% 7/7 [00:23<00:00, 3.91s/it]

/usr/local/lib/python3.10/dist-packages/transformers/tokenization\_utils\_base.py:1601: FutureWarning: `clean\_up\_tokenization\_spaces` was not set. It will be set to `True` by default. This b  
warnings.warn(  
100% 50/50 [00:06<00:00, 7.57it/s]



```
1 import torch
2 from diffusers import StableDiffusionPipeline
3 from PIL import Image
4
5 # Load the Stable Diffusion model
6 pipe = StableDiffusionPipeline.from_pretrained("CompVis/stable-diffusion-v1-4", torch_dtype=torch.float16)
7 pipe = pipe.to("cuda")
8
9 # Generate an image
10 prompt = "taj mahal"
11 image = pipe(prompt).images[0]
12
13 # Save the image
14 image.save("generated_image.png") # Save with the correct file name
15
16 # Display the image
17 image.show()
18
19 # Display the image in Google Colab
```

```
20 from IPython.display import Image as IPyImage # to display on colab.  
21 IPyImage("generated_image.png") # Reference the correct file name
```

```
1 import torch  
2 from diffusers import StableDiffusionPipeline  
3 from PIL import Image  
4  
5 # Load the Stable Diffusion model  
6 pipe = StableDiffusionPipeline.from_pretrained("CompVis/stable-diffusion-v1-4", torch_dtype=torch.float16)  
7 pipe = pipe.to("cuda")  
8  
9 # Generate an image  
10 prompt = "red for"  
11 image = pipe(prompt).images[0]  
12  
13 # Save the image  
14 image.save("generated_image.png") # Save with the correct file name  
15  
16 # Display the image
```