


```
!pip install transformers timm torch torchvision torchaudio pytorch-lightning
!pip install googletrans==4.0.0-rc1
!pip install gtts
!pip install Pillow
!pip install git+https://github.com/salesforce/BLIP.git
!pip install opencv-python
```

```
Uninstalling hpack-4.1.0:
  Successfully uninstalled hpack-4.1.0
Attempting uninstall: h11
  Found existing installation: h11 0.16.0
  Uninstalling h11-0.16.0:
    Successfully uninstalled h11-0.16.0
Attempting uninstall: chardet
  Found existing installation: chardet 5.2.0
  Uninstalling chardet-5.2.0:
    Successfully uninstalled chardet-5.2.0
Attempting uninstall: idna
  Found existing installation: idna 3.10
  Uninstalling idna-3.10:
    Successfully uninstalled idna-3.10
Attempting uninstall: h2
  Found existing installation: h2 4.2.0
  Uninstalling h2-4.2.0:
    Successfully uninstalled h2-4.2.0
Attempting uninstall: httpcore
  Found existing installation: httpcore 1.0.9
  Uninstalling httpcore-1.0.9:
    Successfully uninstalled httpcore-1.0.9
Attempting uninstall: httpx
  Found existing installation: httpx 0.28.1
  Uninstalling httpx-0.28.1:
    Successfully uninstalled httpx-0.28.1
ERROR: pip's dependency resolver does not currently take into account all the packages that are installed. This behaviour is the source of the current failure.
google-genai 1.15.0 requires httpx<1.0.0, >=0.28.1, but you have httpx 0.13.3 which is incompatible.
openai 1.78.1 requires httpx<1.0.0, >=0.23.0, but you have httpx 0.13.3 which is incompatible.
Successfully installed chardet-3.0.4 googletrans-4.0.0rc1 h11-0.9.0 h2-3.2.0 hpack-3.0.0 hstspreload-2025.1.1 httpcore-0.9.1 httpx-0.13.3
Collecting gtts
  Downloading gTTS-2.5.4-py3-none-any.whl.metadata (4.1 kB)
Requirement already satisfied: requests<3, >=2.27 in /usr/local/lib/python3.11/dist-packages (from gtts) (2.32.3)
Collecting click<8.2, >=7.1 (from gtts)
  Downloading click-8.1.8-py3-none-any.whl.metadata (2.3 kB)
Requirement already satisfied: charset-normalizer<4, >=2 in /usr/local/lib/python3.11/dist-packages (from requests<3, >=2.27->gtts) (3.4.0)
Requirement already satisfied: idna<4, >=2.5 in /usr/local/lib/python3.11/dist-packages (from requests<3, >=2.27->gtts) (2.10)
Requirement already satisfied: urllib3<3, >=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests<3, >=2.27->gtts) (2.4.0)
Requirement already satisfied: certifi<2025.4.26, >=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests<3, >=2.27->gtts) (2025.4.26)
Downloading gTTS-2.5.4-py3-none-any.whl (29 kB)
Downloading click-8.1.8-py3-none-any.whl (98 kB)
98.2/98.2 kB 4.2 MB/s eta 0:00:00
Installing collected packages: click, gtts
Attempting uninstall: click
  Found existing installation: click 8.2.0
  Uninstalling click-8.2.0:
    Successfully uninstalled click-8.2.0
Successfully installed click-8.1.8 gtts-2.5.4
Requirement already satisfied: Pillow in /usr/local/lib/python3.11/dist-packages (11.2.1)
Collecting git+https://github.com/salesforce/BLIP.git
  Cloning https://github.com/salesforce/BLIP.git to /tmp/pip-req-build-b_q136p4
  Running command git clone --filter=blob:none --quiet https://github.com/salesforce/BLIP.git /tmp/pip-req-build-b_q136p4
  Resolved https://github.com/salesforce/BLIP.git to commit 3a29b7410476bf5f2ba0955827390eb6ea1f4f9d
ERROR: git+https://github.com/salesforce/BLIP.git does not appear to be a Python project: neither 'setup.py' nor 'pyproject.toml' found
Requirement already satisfied: opencv-python in /usr/local/lib/python3.11/dist-packages (4.11.0.86)
Requirement already satisfied: numpy<2, >=1.21.2 in /usr/local/lib/python3.11/dist-packages (from opencv-python) (2.0.2)
```

```
from google.colab import files
uploaded = files.upload() # Upload a monument or painting image
```

 Choose Files No file chosen Upload widget is only available when the cell has been executed in the current browser session. Please rerun this cell to enable

```
from PIL import Image
from transformers import BlipProcessor, BlipForConditionalGeneration
import torch
```

```
image_path = list(uploaded.keys())[0]
image = Image.open(image_path).convert("RGB")
```

```
processor = BlipProcessor.from_pretrained("Salesforce/blip-image-captioning-base")
model = BlipForConditionalGeneration.from_pretrained("Salesforce/blip-image-captioning-base").eval()

inputs = processor(image, return_tensors="pt")
caption_ids = model.generate(**inputs)
caption = processor.decode(caption_ids[0], skip_special_tokens=True)
print("🖼️ Caption:", caption)
```

⚠️ Using a slow image processor as `use_fast` is unset and a slow processor was saved with this model. `use_fast=True` will be the default
 /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 secret
 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(
preprocessor_config.json: 100% 287/287 [00:00<00:00, 24.5kB/s]
tokenizer_config.json: 100% 506/506 [00:00<00:00, 24.1kB/s]
vocab.txt: 100% 232k/232k [00:00<00:00, 5.11MB/s]
tokenizer.json: 100% 711k/711k [00:00<00:00, 15.0MB/s]
special_tokens_map.json: 100% 125/125 [00:00<00:00, 7.68kB/s]
config.json: 100% 4.56k/4.56k [00:00<00:00, 324kB/s]
pytorch_model.bin: 100% 990M/990M [00:05<00:00, 203MB/s]
model.safetensors: 100% 990M/990M [00:06<00:00, 186MB/s]
```

🖼️ Caption: a sculpture of a woman playing the violin in the museum of the louvre, paris, france - stock

```
from gtts import gTTS
import IPython.display as display

tts = gTTS(text=caption, lang='en')
tts.save("caption.mp3")
display.display(display.Audio("caption.mp3", autoplay=True))
```

⏮️ 0:00 / 0:06

⏭️

```
from googletrans import Translator

translator = Translator()
translated = translator.translate(caption, dest="fr") # Change 'fr' to any language code
print("🌐 Translated (French):", translated.text)

tts_trans = gTTS(text=translated.text, lang='fr')
tts_trans.save("caption_fr.mp3")
display.display(display.Audio("caption_fr.mp3", autoplay=True))
```

⏮️ 🌐 Translated (French): Une sculpture d'une femme jouant du violon au musée du Louvre, Paris, France - Stock

0:00 / 0:06

⏭️

```
import torchvision.models as models
import torchvision.transforms as T

resnet = models.resnet50(pretrained=True)
resnet.eval()

transform = T.Compose([
    T.Resize(256),
    T.CenterCrop(224),
    T.ToTensor(),
    T.Normalize(mean=[0.485, 0.456, 0.406],
                 std=[0.229, 0.224, 0.225]),
])
```

```
img_tensor = transform(image).unsqueeze(0)
with torch.no_grad():
    output = resnet(img_tensor)
pred_idx = output.argmax().item()

!wget -q https://raw.githubusercontent.com/pytorch/hub/master/imagenet_classes.txt -O imagenet_classes.txt
with open("imagenet_classes.txt") as f:
    labels = f.readlines()

label = labels[pred_idx].strip()
print("🔍 Classified As:", label)
```

```
🔄 /usr/local/lib/python3.11/dist-packages/torchvision/models/_utils.py:208: UserWarning: The parameter 'pretrained' is deprecated since 0.
warnings.warn(
/usr/local/lib/python3.11/dist-packages/torchvision/models/_utils.py:223: UserWarning: Arguments other than a weight enum or `None` for
warnings.warn(msg)
Downloading: "https://download.pytorch.org/models/resnet50-0676ba61.pth" to /root/.cache/torch/hub/checkpoints/resnet50-0676ba61.pth
100%|██████████| 97.8M/97.8M [00:00<00:00, 155MB/s]
🔍 Classified As: fountain
```

```
from transformers import ViltProcessor, ViltForQuestionAnswering

vqa_processor = ViltProcessor.from_pretrained("dandelin/vilt-b32-finetuned-vqa")
vqa_model = ViltForQuestionAnswering.from_pretrained("dandelin/vilt-b32-finetuned-vqa")

question = "What is in the image?"
inputs = vqa_processor(image, question, return_tensors="pt")
out = vqa_model(**inputs)
answer = vqa_model.config.id2label[out.logits.argmax(-1).item()]

print("🔍 Answer:", answer)

tts_ans = gTTS(text=answer, lang='en')
tts_ans.save("vqa.mp3")
display.display(display.Audio("vqa.mp3", autoplay=True))
```

```
🔄 preprocessor_config.json: 100% 251/251 [00:00<00:00, 13.4kB/s]
tokenizer_config.json: 100% 320/320 [00:00<00:00, 15.8kB/s]
vocab.txt: 100% 232k/232k [00:00<00:00, 3.46MB/s]
tokenizer.json: 100% 466k/466k [00:00<00:00, 9.62MB/s]
special_tokens_map.json: 100% 112/112 [00:00<00:00, 5.71kB/s]
config.json: 100% 136k/136k [00:00<00:00, 7.29MB/s]
pytorch_model.bin: 100% 470M/470M [00:03<00:00, 104MB/s]
model.safetensors: 100% 470M/470M [00:04<00:00, 210MB/s]
🔍 Answer: statue
0:00 / 0:01
```

```
import cv2
import matplotlib.pyplot as plt

img_cv = cv2.imread(image_path)
gray = cv2.cvtColor(img_cv, cv2.COLOR_BGR2GRAY)

face_cascade = cv2.CascadeClassifier(cv2.data.harcascades + "haarcascade_frontalface_default.xml")
faces = face_cascade.detectMultiScale(gray, scaleFactor=1.1, minNeighbors=5)

if len(faces) == 0:
    print("😞 No face detected.")
else:
    print(f"😄 {len(faces)} face(s) detected.")
    for (x, y, w, h) in faces:
        cv2.rectangle(img_cv, (x, y), (x+w, y+h), (0, 255, 0), 2)

plt.imshow(cv2.cvtColor(img_cv, cv2.COLOR_BGR2RGB))
plt.axis('off')
plt.show()
```

🔄 😊 2 face(s) detected.



Start coding or [generate](#) with AI.