

```
from transformers import pipeline
```

```
image_captioner = pipeline("image-to-text", model = 'Salesforce/blip-image-captioning-large', device=0)
```

```

/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)
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(
Using a slow image processor as `use_fast` is unset and a slow processor was saved with this model. `use_fast=True`
Device set to use cpu

```

```

image = r'/content/dog.jpg'
caption = image_captioner(image)
caption[0].get('generated_text')

```

```
'arafed group of puppies sitting on a wooden bench in the woods'
```

```
vqa = pipeline('visual-question-answering')
```

```

No model was supplied, defaulted to dandelin/vilt-b32-finetuned-vqa and revision d0a1f6a (https://huggingface.co/dandelin/vilt-b32-finetuned-vqa)
Using a pipeline without specifying a model name and revision in production is not recommended.
Device set to use cuda:0

```

```
image = r'/content/dog.jpg'
```

```

q = input('Ask a question about the image: ')
result = vqa(image, q)
result

```

```

Ask a question about the image: what is the animal
[{'score': 0.8666151165962219, 'answer': 'dog'},
 {'score': 0.300300657749176, 'answer': 'dogs'},
 {'score': 0.025989240035414696, 'answer': 'lab'},
 {'score': 0.013781215995550156, 'answer': 'yes'},
 {'score': 0.012482884339988232, 'answer': 'dog and cat'}]

```

```

!pip install diffusers
import torch
from diffusers import DiffusionPipeline
generator = DiffusionPipeline.from_pretrained("runwayml/stable-diffusion-v1-5")
generator.to("cpu")

```

```

pprompt = input("Enter the prompt:")
image = generator(prompt).images[0]
image

```



6%

3/50 [02:17<31:30, 40.22s/it]



```
prompt = "a purple coloured cup cake covered in chocolate chips "  
image = generator(prompt).images[0]  
image
```



0%| | 0/50 [00:00<?, ?it/s]



```
from transformers import pipeline  
img_cls=pipeline('image-classification')
```



No model was supplied, defaulted to google/vit-base-patch16-224 and revision 3f49326 (<https://huggingface.co/google>)
Using a pipeline without specifying a model name and revision in production is not recommended.
Fast image processor class <class 'transformers.models.vit.image_processing_vit_fast.ViTImageProcessorFast'> is available
Device set to use cuda:0

```
img="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQP3G99LG0doGaGiXecF4xBWtVB5fucmJIXQA&s"
```

```
r=img_cls(img)
r
```

```
det = pipeline("object-detection")
```

```
r =det("https://plus.unsplash.com/premium_photo-1668704252725-52cc74ef3f39?fm=jpg&q=60&w=3000&ixlib=rb-4.1.0&ixid=M3wx
```

```
r
```

```
!pip install deepface librosa speechRecognition
```

```
Requirement already satisfied: decorator>=4.3.0 in /usr/local/lib/python3.11/dist-packages (from librosa) (4.4.2)
Requirement already satisfied: soundfile>=0.12.1 in /usr/local/lib/python3.11/dist-packages (from librosa) (0.13
Requirement already satisfied: pooch>=1.1 in /usr/local/lib/python3.11/dist-packages (from librosa) (1.8.2)
Requirement already satisfied: soxr>=0.3.2 in /usr/local/lib/python3.11/dist-packages (from librosa) (0.5.0.post
Requirement already satisfied: typing_extensions>=4.1.1 in /usr/local/lib/python3.11/dist-packages (from librosa
Requirement already satisfied: lazy_loader>=0.1 in /usr/local/lib/python3.11/dist-packages (from librosa) (0.4)
Requirement already satisfied: msgpack>=1.0 in /usr/local/lib/python3.11/dist-packages (from librosa) (1.1.0)
Requirement already satisfied: termcolor in /usr/local/lib/python3.11/dist-packages (from fire>=0.4.0->deepface)
Requirement already satisfied: blinker>=1.9.0 in /usr/local/lib/python3.11/dist-packages (from Flask>=1.1.2->dee
Requirement already satisfied: click>=8.1.3 in /usr/local/lib/python3.11/dist-packages (from Flask>=1.1.2->deepf
Requirement already satisfied: itsdangerous>=2.2.0 in /usr/local/lib/python3.11/dist-packages (from Flask>=1.1.2
Requirement already satisfied: Jinja2>=3.1.2 in /usr/local/lib/python3.11/dist-packages (from Flask>=1.1.2->deep
Requirement already satisfied: MarkupSafe>=2.1.1 in /usr/local/lib/python3.11/dist-packages (from Flask>=1.1.2->
Requirement already satisfied: Werkzeug>=3.1.0 in /usr/local/lib/python3.11/dist-packages (from Flask>=1.1.2->de
Requirement already satisfied: BeautifulSoup4 in /usr/local/lib/python3.11/dist-packages (from gdown>=3.10.1->de
Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from gdown>=3.10.1->deepface)
Requirement already satisfied: packaging in /usr/local/lib/python3.11/dist-packages (from gunicorn>=20.1.0->deep
Requirement already satisfied: absl-py in /usr/local/lib/python3.11/dist-packages (from keras>=2.2.0->deepface)
Requirement already satisfied: rich in /usr/local/lib/python3.11/dist-packages (from keras>=2.2.0->deepface) (13
Requirement already satisfied: namex in /usr/local/lib/python3.11/dist-packages (from keras>=2.2.0->deepface) (0
Requirement already satisfied: h5py in /usr/local/lib/python3.11/dist-packages (from keras>=2.2.0->deepface) (3.
Requirement already satisfied: optree in /usr/local/lib/python3.11/dist-packages (from keras>=2.2.0->deepface) (
Requirement already satisfied: ml-dtypes in /usr/local/lib/python3.11/dist-packages (from keras>=2.2.0->deepface)
Requirement already satisfied: lz4>=4.3.3 in /usr/local/lib/python3.11/dist-packages (from mtcnn>=0.1.0->deepfac
Requirement already satisfied: llvmlite<0.44,>=0.43.0dev0 in /usr/local/lib/python3.11/dist-packages (from numba
Requirement already satisfied: python-dateutil>=2.8.2 in /usr/local/lib/python3.11/dist-packages (from pandas>=0
Requirement already satisfied: pytz>=2020.1 in /usr/local/lib/python3.11/dist-packages (from pandas>=0.23.4->dee
Requirement already satisfied: tzdata>=2022.7 in /usr/local/lib/python3.11/dist-packages (from pandas>=0.23.4->d
Requirement already satisfied: platformdirs>=2.5.0 in /usr/local/lib/python3.11/dist-packages (from pooch>=1.1->
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from request
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests>=2.27.1->d
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests>=2.2
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests>=2.2
Requirement already satisfied: threadpoolctl>=3.1.0 in /usr/local/lib/python3.11/dist-packages (from scikit-lear
Requirement already satisfied: cffi>=1.0 in /usr/local/lib/python3.11/dist-packages (from soundfile>=0.12.1->lib
Requirement already satisfied: astunparse>=1.6.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow=1.
Requirement already satisfied: flatbuffers>=24.3.25 in /usr/local/lib/python3.11/dist-packages (from tensorflow)
Requirement already satisfied: gast!=0.5.0,!0.5.1,!0.5.2,>=0.2.1 in /usr/local/lib/python3.11/dist-packages (f
Requirement already satisfied: google-pasta>=0.1.1 in /usr/local/lib/python3.11/dist-packages (from tensorflow)
Requirement already satisfied: libclang>=13.0.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow>=1.9
Requirement already satisfied: opt-einsum>=2.3.2 in /usr/local/lib/python3.11/dist-packages (from tensorflow>=1.
Requirement already satisfied: protobuf!=4.21.0,!4.21.1,!4.21.2,!4.21.3,!4.21.4,!4.21.5,<6.0.0dev,>=3.20.3
Requirement already satisfied: setuptools in /usr/local/lib/python3.11/dist-packages (from tensorflow>=1.9.0->de
Requirement already satisfied: six>=1.12.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow>=1.9.0->d
Requirement already satisfied: wrapt>=1.11.0 in /usr/local/lib/python3.11/dist-packages (from tensorflow>=1.9.0-
Requirement already satisfied: grpcio<2.0,>=1.24.3 in /usr/local/lib/python3.11/dist-packages (from tensorflow)
Requirement already satisfied: tensorboard<2.19,>=2.18 in /usr/local/lib/python3.11/dist-packages (from tensorfl
Requirement already satisfied: tensorflow-io-gcs-filesystem>=0.23.1 in /usr/local/lib/python3.11/dist-packages (
Requirement already satisfied: wheel<1.0,>=0.23.0 in /usr/local/lib/python3.11/dist-packages (from astunparse>=1
Requirement already satisfied: pycparser in /usr/local/lib/python3.11/dist-packages (from cffi>=1.0->soundfile)
Requirement already satisfied: markdown>=2.6.8 in /usr/local/lib/python3.11/dist-packages (from tensorboard<2.19
Requirement already satisfied: tensorboard-data-server<0.8.0,>=0.7.0 in /usr/local/lib/python3.11/dist-packages
Requirement already satisfied: soupsieve>1.2 in /usr/local/lib/python3.11/dist-packages (from beautifulsoup4->gd
Requirement already satisfied: PySocks!=1.5.7,>=1.5.6 in /usr/local/lib/python3.11/dist-packages (from requests[
Requirement already satisfied: markdown-it-py>=2.2.0 in /usr/local/lib/python3.11/dist-packages (from rich->kera
Requirement already satisfied: pygments<3.0.0,>=2.13.0 in /usr/local/lib/python3.11/dist-packages (from rich->ke
Requirement already satisfied: mdurl~=0.1 in /usr/local/lib/python3.11/dist-packages (from markdown-it-py>=2.2.0
```

```

from deepface import DeepFace
import librosa
import speech_recognition as sr
import numpy as np

emotions = ['happy', 'sad', 'neutral', 'angry', 'fear', 'surprise']

r = DeepFace.analyze(img_path = '/content/jimin smile.jpg', actions = ['emotion'])
r[0]

```

```

{
  'emotion': {
    'angry': np.float32(0.012846767),
    'disgust': np.float32(7.793615e-09),
    'fear': np.float32(9.379978),
    'happy': np.float32(0.00012824452),
    'sad': np.float32(89.80199),
    'surprise': np.float32(1.0246321e-06),
    'neutral': np.float32(0.80506384)},
  'dominant_emotion': 'sad',
  'region': {
    'x': 555,
    'y': 344,
    'w': 517,
    'h': 517,
    'left_eye': (900, 539),
    'right_eye': (714, 552)},
  'face_confidence': np.float64(0.95)}

```

```

from transformers import pipeline

```

```

image_captioner = pipeline("image-to-text", model = 'Salesforce/blip-image-captioning-large', device=0)

```

```

/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)
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(
Using a slow image processor as `use_fast` is unset and a slow processor was saved with this model. `use_fast=True`
Device set to use cpu

```

```

image = r'/content/taj.jpeg'
caption = image_captioner(image)
caption[0].get('generated_text')

```

```

'araffe view of a white building with a fountain in the middle'

```

```

!pip install gTTS

```

```

Requirement already satisfied: gTTS in /usr/local/lib/python3.11/dist-packages (2.5.4)
Requirement already satisfied: requests<3,>=2.27 in /usr/local/lib/python3.11/dist-packages (from gTTS) (2.32.3)
Requirement already satisfied: click<8.2,>=7.1 in /usr/local/lib/python3.11/dist-packages (from gTTS) (8.1.8)
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) (3.10.1)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->gTTS) (2.2.3)
Requirement already satisfied: certifi<2025.0.0,>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->gTTS) (2024.12.14)

```

```

from gtts import gTTS
from IPython.display import Audio

```

```

a = gTTS(input("Enter any text:"))

```

```

Enter any text:araffe view of a white building with a fountain in the middle

```

```

a.save("taj.mp3")

```

```

Audio("taj.mp3")

```




0:00 / 0:03

```
vqa = pipeline('visual-question-answering')
```



No model was supplied, defaulted to dandelin/vilt-b32-finetuned-vqa and revision d0a1f6a (<https://huggingface.co/dandelin/vilt-b32-finetuned-vqa>)
Using a pipeline without specifying a model name and revision in production is not recommended.
Device set to use cpu

```
image = r'/content/taj.jpeg'
```

```
q = input('Ask a question about the image: ')
result = vqa(image, q)
result
```



```
Ask a question about the image: what is the picture
[{'score': 0.13345396518707275, 'answer': 'landscape'},
 {'score': 0.12381446361541748, 'answer': 'nature'},
 {'score': 0.06416231393814087, 'answer': 'outside'},
 {'score': 0.06239980086684227, 'answer': 'church'},
 {'score': 0.04174917936325073, 'answer': 'statue'}]
```

```
!pip install googletrans==4.0.0-rc1
```



```
Requirement already satisfied: googletrans==4.0.0-rc1 in /usr/local/lib/python3.11/dist-packages (4.0.0rc1)
Requirement already satisfied: httpx==0.13.3 in /usr/local/lib/python3.11/dist-packages (from googletrans==4.0.0-rc1)
Requirement already satisfied: certifi in /usr/local/lib/python3.11/dist-packages (from httpx==0.13.3->googletrans)
Requirement already satisfied: hstspreload in /usr/local/lib/python3.11/dist-packages (from httpx==0.13.3->googletrans)
Requirement already satisfied: sniffio in /usr/local/lib/python3.11/dist-packages (from httpx==0.13.3->googletrans)
Requirement already satisfied: chardet==3.* in /usr/local/lib/python3.11/dist-packages (from httpx==0.13.3->googletrans)
Requirement already satisfied: idna==2.* in /usr/local/lib/python3.11/dist-packages (from httpx==0.13.3->googletrans)
Requirement already satisfied: rfc3986<2,>=1.3 in /usr/local/lib/python3.11/dist-packages (from httpx==0.13.3->googletrans)
Requirement already satisfied: httpcore==0.9.* in /usr/local/lib/python3.11/dist-packages (from httpx==0.13.3->googletrans)
Requirement already satisfied: h11<0.10,>=0.8 in /usr/local/lib/python3.11/dist-packages (from httpcore==0.9.*->httpx==0.13.3->googletrans)
Requirement already satisfied: h2==3.* in /usr/local/lib/python3.11/dist-packages (from httpcore==0.9.*->httpx==0.13.3->googletrans)
Requirement already satisfied: hyperframe<6,>=5.2.0 in /usr/local/lib/python3.11/dist-packages (from h2==3.*->httpcore==0.9.*->httpx==0.13.3->googletrans)
Requirement already satisfied: hpack<4,>=3.0 in /usr/local/lib/python3.11/dist-packages (from h2==3.*->httpcore==0.9.*->httpx==0.13.3->googletrans)
```

```
!pip install SpeechRecognition pydub
```



```
Collecting SpeechRecognition
  Downloading speechrecognition-3.14.3-py3-none-any.whl.metadata (30 kB)
Collecting pydub
  Downloading pydub-0.25.1-py2.py3-none-any.whl.metadata (1.4 kB)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.11/dist-packages (from SpeechRecognition)
Downloading speechrecognition-3.14.3-py3-none-any.whl (32.9 MB)
32.9/32.9 MB 29.8 MB/s eta 0:00:00
Downloading pydub-0.25.1-py2.py3-none-any.whl (32 kB)
Installing collected packages: pydub, SpeechRecognition
Successfully installed SpeechRecognition-3.14.3 pydub-0.25.1
```

```
import speech_recognition as sr
reco = sr.Recognizer()
with sr.AudioFile("/content/tajwav.wav") as source:
    audio_data = reco.record(source)
    print("Translation: ",reco.recognize_google(audio_data))
```



```
Translation: view of a white building with a fountain in the middle
```

```
english_text = reco.recognize_google(audio_data, language='en-IN')
print("Translation:", english_text)
```



```
Translation: view of a white building with a fountain in the middle
```

```
from googletrans import Translator
```

```
translator = Translator()
translated = translator.translate(" view of a white building with a fountain in the middle", src='en', dest='te')
print("Telugu Translation:", translated.text)
```

🔍 Telugu Translation: మధ్యలో ఫౌంటెన్ ఉన్న తెల్ల భవనం యొక్క దృశ్యం

```
from gtts import gTTS
from IPython.display import Audio
telugu_text = "మధ్యలో ఫౌంటెన్ ఉన్న తెల్ల భవనం యొక్క దృశ్యం"
```

```
speech = gTTS(text=telugu_text, lang='te')
speech.save("tajmahal.mp3")
```

```
Audio("tajmahal.mp3")
```



0:01 / 0:03



```
from transformers import pipeline
img_cls=pipeline('image-classification')
```



No model was supplied, defaulted to google/vit-base-patch16-224 and revision 3f49326 (<https://huggingface.co/google/vit-base-patch16-224>)
Using a pipeline without specifying a model name and revision in production is not recommended.

config.json: 100% 69.7k/69.7k [00:00<00:00, 2.94MB/s]

Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed. Falling back to regular HTTP download
WARNING:huggingface_hub.file_download:Xet Storage is enabled for this repo, but the 'hf_xet' package is not installed

model.safetensors: 100% 346M/346M [00:04<00:00, 85.3MB/s]

preprocessor_config.json: 100% 160/160 [00:00<00:00, 2.76kB/s]

Fast image processor class <class 'transformers.models.vit.image_processing_vit_fast.ViTImageProcessorFast'> is available
Device set to use cpu

```
img="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQP3G99LG0doGaGiXecF4xBWTVB5fucmJIxQA&s"
```

```
r=img_cls(img)
r
```



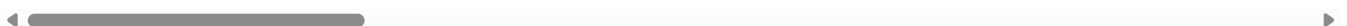
```
[{'label': 'Labrador retriever', 'score': 0.885223388671875},
 {'label': 'golden retriever', 'score': 0.07846732437610626},
 {'label': 'kuvasz', 'score': 0.012887442484498024},
 {'label': 'curly-coated retriever', 'score': 0.0015661275247111917},
 {'label': 'Sussex spaniel', 'score': 0.001219155383296311}]
```

```
det = pipeline("object-detection")
```



No model was supplied, defaulted to facebook/detr-resnet-50 and revision 1d5f47b (<https://huggingface.co/facebook/detr-resnet-50>)
Using a pipeline without specifying a model name and revision in production is not recommended.

Some weights of the model checkpoint at facebook/detr-resnet-50 were not used when initializing DetrForObjectDetection
- This IS expected if you are initializing DetrForObjectDetection from the checkpoint of a model trained on another task
- This IS NOT expected if you are initializing DetrForObjectDetection from the checkpoint of a model that you expect to be initialized with the weights of the model
Device set to use cpu



```
r =det("https://plus.unsplash.com/premium_photo-1668704252725-52cc74ef3f39?fm=jpg&q=60&w=3000&iixlib=rb-4.1.0&iixid=M3wx")
```

```
r
```



```
[{'score': 0.9982910752296448,
 'label': 'vase',
 'box': {'xmin': 842, 'ymin': 2155, 'xmax': 2132, 'ymax': 3917}},
 {'score': 0.9963465332984924,
```

```
!pip install deepface librosa speechRecognition
```

```
from deepface import DeepFace
import librosa
import speech_recognition as sr
import numpy as np
```

```
emotions = ['happy','sad','neutral','angry','fear','surprice']
```

```
r = DeepFace.analyze(img_path = '/content/jimin smile.jpg', actions = ['emotion'])
r[0]
```

```
{
  'emotion': {
    'angry': np.float32(12.333034),
    'disgust': np.float32(1.3174389e-07),
    'fear': np.float32(74.50147),
    'happy': np.float32(3.1286171e-09),
    'sad': np.float32(6.378527),
    'surprise': np.float32(0.0009804316),
    'neutral': np.float32(6.785988)},
  'dominant_emotion': 'fear',
  'region': {
    'x': 627,
    'y': 496,
    'w': 147,
    'h': 147,
    'left_eye': None,
    'right_eye': None},
  'face_confidence': np.float64(0.95)}

```

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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

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