

```
1 !pip install vosk ffmpeg-python
```

```
Collecting vosk
  Downloading vosk-0.3.45-py3-none-manylinux_2_12_x86_64.manylinux2010_x86_64.whl.metadata (1.8 kB)
Collecting ffmpeg-python
  Downloading ffmpeg_python-0.2.0-py3-none-any.whl.metadata (1.7 kB)
Requirement already satisfied: cffi>=1.0 in /usr/local/lib/python3.11/dist-packages (from vosk) (1.17.1)
Requirement already satisfied: requests in /usr/local/lib/python3.11/dist-packages (from vosk) (2.32.3)
Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages (from vosk) (4.67.1)
Collecting srt (from vosk)
  Downloading srt-3.5.3.tar.gz (28 kB)
  Preparing metadata (setup.py) ... done
Requirement already satisfied: websockets in /usr/local/lib/python3.11/dist-packages (from vosk) (14.2)
Requirement already satisfied: future in /usr/local/lib/python3.11/dist-packages (from ffmpeg-python) (1.0.0)
Requirement already satisfied: pycparser in /usr/local/lib/python3.11/dist-packages (from cffi>=1.0->vosk) (2.22)
Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests->vosk) (3.4.1)
Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests->vosk) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests->vosk) (2.3.0)
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests->vosk) (2025.1.31)
Downloading vosk-0.3.45-py3-none-manylinux_2_12_x86_64.manylinux2010_x86_64.whl (7.2 MB)
7.2/7.2 MB 18.9 MB/s eta 0:00:00
Downloading ffmpeg_python-0.2.0-py3-none-any.whl (25 kB)
Building wheels for collected packages: srt
  Building wheel for srt (setup.py) ... done
  Created wheel for srt: filename=srt-3.5.3-py3-none-any.whl size=22428 sha256=a55a408bdbb3307a1e55f31501294e162d38141b07f16b0c2e0e9
  Stored in directory: /root/.cache/pip/wheels/1f/43/f1/23ee9119497fcb57d9f7046fbf34c6d9027c46a1fa7824cf08
Successfully built srt
Installing collected packages: srt, ffmpeg-python, vosk
Successfully installed ffmpeg-python-0.2.0 srt-3.5.3 vosk-0.3.45
```

```
1 !apt install ffmpeg
```

```
Reading package lists... Done
Building dependency tree... Done
Reading state information... Done
ffmpeg is already the newest version (7:4.4.2-0ubuntu2.22.04.1).
0 upgraded, 0 newly installed, 0 to remove and 29 not upgraded.
```

```
1 # Download Hindi model
2 !wget https://alphacephei.com/vosk/models/vosk-model-hi-0.22.zip
3 !unzip vosk-model-hi-0.22.zip
```

```
--2025-03-05 04:14:33-- https://alphacephei.com/vosk/models/vosk-model-hi-0.22.zip
Resolving alphacephei.com (alphacephei.com)... 188.40.21.16, 2a01:4f8:13a:279f::2
Connecting to alphacephei.com (alphacephei.com)|188.40.21.16|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 1561909274 (1.5G) [application/zip]
Saving to: 'vosk-model-hi-0.22.zip'
```

```
vosk-model-hi-0.22. 100%[=====] 1.45G 28.5MB/s in 55s
```

```
2025-03-05 04:15:29 (27.2 MB/s) - 'vosk-model-hi-0.22.zip' saved [1561909274/1561909274]
```

```
Archive: vosk-model-hi-0.22.zip
  creating: vosk-model-hi-0.22/
  creating: vosk-model-hi-0.22/graph/
  inflating: vosk-model-hi-0.22/graph/words.txt
  creating: vosk-model-hi-0.22/graph/phones/
  extracting: vosk-model-hi-0.22/graph/phones/silence.cs1
  inflating: vosk-model-hi-0.22/graph/phones/align_lexicon.txt
  inflating: vosk-model-hi-0.22/graph/phones/word_boundary.txt
  extracting: vosk-model-hi-0.22/graph/phones/optional_silence.cs1
  extracting: vosk-model-hi-0.22/graph/phones/optional_silence.txt
  inflating: vosk-model-hi-0.22/graph/phones/disambig.int
  extracting: vosk-model-hi-0.22/graph/phones/disambig.txt
  inflating: vosk-model-hi-0.22/graph/phones/align_lexicon.int
  inflating: vosk-model-hi-0.22/graph/phones/word_boundary.int
  extracting: vosk-model-hi-0.22/graph/phones/optional_silence.int
  inflating: vosk-model-hi-0.22/graph/phones.txt
  inflating: vosk-model-hi-0.22/graph/HCLG.fst
  extracting: vosk-model-hi-0.22/graph/num_pdfs
  inflating: vosk-model-hi-0.22/graph/disambig_tid.int
  creating: vosk-model-hi-0.22/am/
  inflating: vosk-model-hi-0.22/am/final.mdl
  creating: vosk-model-hi-0.22/rnnlm/
  inflating: vosk-model-hi-0.22/rnnlm/special_symbol_opts.conf
  inflating: vosk-model-hi-0.22/rnnlm/final.raw
  inflating: vosk-model-hi-0.22/rnnlm/special_symbol_opts.txt
  inflating: vosk-model-hi-0.22/rnnlm/feat_embedding.final.mat
  inflating: vosk-model-hi-0.22/rnnlm/word_feats.txt
  inflating: vosk-model-hi-0.22/README
  creating: vosk-model-hi-0.22/rescore/
  inflating: vosk-model-hi-0.22/rescore/G.fst
  inflating: vosk-model-hi-0.22/rescore/G.carpa
  creating: vosk-model-hi-0.22/conf/
```

```

inflating: vosk-model-hi-0.22/conf/model.conf
inflating: vosk-model-hi-0.22/conf/ivector.conf
inflating: vosk-model-hi-0.22/conf/mfcc.conf
creating: vosk-model-hi-0.22/ivector/
inflating: vosk-model-hi-0.22/ivector/final.dubm
inflating: vosk-model-hi-0.22/ivector/global_cmvn.stats
inflating: vosk-model-hi-0.22/ivector/splice_opts
inflating: vosk-model-hi-0.22/ivector/final.ie
inflating: vosk-model-hi-0.22/ivector/final.mat
inflating: vosk-model-hi-0.22/ivector/splice.conf
extracting: vosk-model-hi-0.22/ivector/online_cmvn.conf

```

```

1 # Download English model
2 !wget https://alphacephei.com/vosk/models/vosk-model-small-en-us-0.15.zip
3 !unzip vosk-model-small-en-us-0.15.zip

```

```

--2025-03-05 04:16:49-- https://alphacephei.com/vosk/models/vosk-model-small-en-us-0.15.zip
Resolving alphacephei.com (alphacephei.com)... 188.40.21.16, 2a01:4f8:13a:279f::2
Connecting to alphacephei.com (alphacephei.com)|188.40.21.16|:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 41205931 (39M) [application/zip]
Saving to: 'vosk-model-small-en-us-0.15.zip'

```

```
vosk-model-small-en 100%[=====>] 39.30M 16.3MB/s in 2.4s
```

```
2025-03-05 04:16:52 (16.3 MB/s) - 'vosk-model-small-en-us-0.15.zip' saved [41205931/41205931]
```

```

Archive: vosk-model-small-en-us-0.15.zip
creating: vosk-model-small-en-us-0.15/
creating: vosk-model-small-en-us-0.15/am/
inflating: vosk-model-small-en-us-0.15/am/final.mdl
creating: vosk-model-small-en-us-0.15/graph/
inflating: vosk-model-small-en-us-0.15/graph/disambig_tid.int
inflating: vosk-model-small-en-us-0.15/graph/HCLr.fst
inflating: vosk-model-small-en-us-0.15/graph/Gr.fst
creating: vosk-model-small-en-us-0.15/graph/phones/
inflating: vosk-model-small-en-us-0.15/graph/phones/word_boundary.int
creating: vosk-model-small-en-us-0.15/conf/
inflating: vosk-model-small-en-us-0.15/conf/model.conf
inflating: vosk-model-small-en-us-0.15/conf/mfcc.conf
creating: vosk-model-small-en-us-0.15/ivector/
inflating: vosk-model-small-en-us-0.15/ivector/splice.conf
inflating: vosk-model-small-en-us-0.15/ivector/final.dubm
inflating: vosk-model-small-en-us-0.15/ivector/global_cmvn.stats
inflating: vosk-model-small-en-us-0.15/ivector/final.ie
inflating: vosk-model-small-en-us-0.15/ivector/online_cmvn.conf
inflating: vosk-model-small-en-us-0.15/ivector/final.mat
inflating: vosk-model-small-en-us-0.15/README

```

```
1 pip install SpeechRecognition
```

```

Collecting SpeechRecognition
  Downloading SpeechRecognition-3.14.1-py3-none-any.whl.metadata (31 kB)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.11/dist-packages (from SpeechRecognition) (4.12.2)
Downloading SpeechRecognition-3.14.1-py3-none-any.whl (32.9 MB)
32.9/32.9 MB 37.7 MB/s eta 0:00:00
Installing collected packages: SpeechRecognition
Successfully installed SpeechRecognition-3.14.1

```

```
1 !pip install speechrecognition
```

```

Requirement already satisfied: speechrecognition in /usr/local/lib/python3.11/dist-packages (3.14.1)
Requirement already satisfied: typing-extensions in /usr/local/lib/python3.11/dist-packages (from speechrecognition) (4.12.2)

```

```
1 !pip install pydub
```

```

Collecting pydub
  Downloading pydub-0.25.1-py2.py3-none-any.whl.metadata (1.4 kB)
Downloading pydub-0.25.1-py2.py3-none-any.whl (32 kB)
Installing collected packages: pydub
Successfully installed pydub-0.25.1

```

```





1 from pydub import AudioSegment
2 import speech_recognition as sr
3
4 def convert_to_wav(input_file, output_file):
5     """Convert audio file to WAV format."""
6     audio = AudioSegment.from_file(input_file)
7     audio.export(output_file, format='wav')
8     print("✅ Conversion to WAV successful!")
9
10 def transcribe_audio(audio_path):
11     """Transcribe audio file to text using Google's Speech Recognition API."""
12     recognizer = sr.Recognizer()

```


```

13 with sr.AudioFile(audio_path) as source:
14     print("🔊 Processing audio...")
15     audio = recognizer.record(source)
16     try:
17         text = recognizer.recognize_google(audio)
18         print("✅ Transcription Successful!\n")
19         return text
20     except sr.UnknownValueError:
21         return "⚠️ Could not understand the audio."
22     except sr.RequestError:
23         return "⚠️ Could not connect to the speech recognition service."
24
25 # Convert to WAV
26 input_file = "/content/WhatsApp Audio 2025-03-05 at 09.43.53_2247dcdcf.dat.unknown"
27 wav_file = "/content/converted_audio.wav"
28 convert_to_wav(input_file, wav_file)
29
30 # Transcribe
31 transcribed_text = transcribe_audio(wav_file)
32 print("🗨️ Transcribed Text:\n", transcribed_text)

```

 Conversion to WAV successful!
 Processing audio...
 Transcribed Text:
 ⚠️ Could not connect to the speech recognition service.

```
1 pip install openai-whisper
```

 Collecting openai-whisper
 Downloading openai-whisper-20240930.tar.gz (800 kB)
 800.5/800.5 kB 6.0 MB/s eta 0:00:00
 Installing build dependencies ... done
 Getting requirements to build wheel ... done
 Preparing metadata (pyproject.toml) ... done
 Requirement already satisfied: numba in /usr/local/lib/python3.11/dist-packages (from openai-whisper) (0.61.0)
 Requirement already satisfied: numpy in /usr/local/lib/python3.11/dist-packages (from openai-whisper) (1.26.4)
 Requirement already satisfied: torch in /usr/local/lib/python3.11/dist-packages (from openai-whisper) (2.5.1+cu124)
 Requirement already satisfied: tqdm in /usr/local/lib/python3.11/dist-packages (from openai-whisper) (4.67.1)
 Requirement already satisfied: more-itertools in /usr/local/lib/python3.11/dist-packages (from openai-whisper) (10.6.0)
 Collecting tiktoken (from openai-whisper)
 Downloading tiktoken-0.9.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl.metadata (6.7 kB)
 Requirement already satisfied: triton>=2.0.0 in /usr/local/lib/python3.11/dist-packages (from openai-whisper) (3.1.0)
 Requirement already satisfied: filelock in /usr/local/lib/python3.11/dist-packages (from triton>=2.0.0->openai-whisper) (3.17.0)
 Requirement already satisfied: llvmlite<0.45,>=0.44.0dev0 in /usr/local/lib/python3.11/dist-packages (from numba->openai-whisper)
 Requirement already satisfied: regex>=2022.1.18 in /usr/local/lib/python3.11/dist-packages (from tiktoken->openai-whisper) (2024)
 Requirement already satisfied: requests>=2.26.0 in /usr/local/lib/python3.11/dist-packages (from tiktoken->openai-whisper) (2.32)
 Requirement already satisfied: typing-extensions>=4.8.0 in /usr/local/lib/python3.11/dist-packages (from torch->openai-whisper)
 Requirement already satisfied: networkx in /usr/local/lib/python3.11/dist-packages (from torch->openai-whisper) (3.4.2)
 Requirement already satisfied: Jinja2 in /usr/local/lib/python3.11/dist-packages (from torch->openai-whisper) (3.1.5)
 Requirement already satisfied: fsspec in /usr/local/lib/python3.11/dist-packages (from torch->openai-whisper) (2024.10.0)
 Collecting nvidia-cuda-nvrtc-cu12==12.4.127 (from torch->openai-whisper)
 Downloading nvidia_cuda_nvrtc_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-cuda-runtime-cu12==12.4.127 (from torch->openai-whisper)
 Downloading nvidia_cuda_runtime_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-cuda-cupti-cu12==12.4.127 (from torch->openai-whisper)
 Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
 Collecting nvidia-cudnn-cu12==9.1.0.70 (from torch->openai-whisper)
 Downloading nvidia_cudnn_cu12-9.1.0.70-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
 Collecting nvidia-cublas-cu12==12.4.5.8 (from torch->openai-whisper)
 Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-cufft-cu12==11.2.1.3 (from torch->openai-whisper)
 Downloading nvidia_cufft_cu12-11.2.1.3-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-curand-cu12==10.3.5.147 (from torch->openai-whisper)
 Downloading nvidia_curand_cu12-10.3.5.147-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Collecting nvidia-cusolver-cu12==11.6.1.9 (from torch->openai-whisper)
 Downloading nvidia_cusolver_cu12-11.6.1.9-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
 Collecting nvidia-cuspars-cu12==12.3.1.170 (from torch->openai-whisper)
 Downloading nvidia_cuspars-cu12-12.3.1.170-py3-none-manylinux2014_x86_64.whl.metadata (1.6 kB)
 Requirement already satisfied: nvidia-nccl-cu12==2.21.5 in /usr/local/lib/python3.11/dist-packages (from torch->openai-whisper)
 Requirement already satisfied: nvidia-nvtx-cu12==12.4.127 in /usr/local/lib/python3.11/dist-packages (from torch->openai-whisper)
 Collecting nvidia-nvjitlink-cu12==12.4.127 (from torch->openai-whisper)
 Downloading nvidia_nvjitlink_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl.metadata (1.5 kB)
 Requirement already satisfied: sympy==1.13.1 in /usr/local/lib/python3.11/dist-packages (from torch->openai-whisper) (1.13.1)
 Requirement already satisfied: mpmath<1.4,>=1.1.0 in /usr/local/lib/python3.11/dist-packages (from sympy==1.13.1->torch->openai-whisper)
 Requirement already satisfied: charset-normalizer<4,>=2 in /usr/local/lib/python3.11/dist-packages (from requests>=2.26.0->tiktoken->openai-whisper)
 Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests>=2.26.0->tiktoken->openai-whisper)
 Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests>=2.26.0->tiktoken->openai-whisper)
 Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests>=2.26.0->tiktoken->openai-whisper)
 Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from Jinja2->torch->openai-whisper) (2.1.5)
 Downloading tiktoken-0.9.0-cp311-cp311-manylinux_2_17_x86_64.manylinux2014_x86_64.whl (1.2 MB)
 1.2/1.2 MB 27.5 MB/s eta 0:00:00
 Downloading nvidia_cublas_cu12-12.4.5.8-py3-none-manylinux2014_x86_64.whl (363.4 MB)
 363.4/363.4 MB 4.4 MB/s eta 0:00:00
 Downloading nvidia_cuda_cupti_cu12-12.4.127-py3-none-manylinux2014_x86_64.whl (13.8 MB)
 13.8/13.8 MB 79.1 MB/s eta 0:00:00


```

Requirement already satisfied: idna<4,>=2.5 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->streamlit) (3.10)
Requirement already satisfied: urllib3<3,>=1.21.1 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->streamlit) (2
Requirement already satisfied: certifi>=2017.4.17 in /usr/local/lib/python3.11/dist-packages (from requests<3,>=2.27->streamlit) (20
Requirement already satisfied: smmap<6,>=3.0.1 in /usr/local/lib/python3.11/dist-packages (from gitdb<5,>=4.0.1->gitpython!=3.1.19,<
Requirement already satisfied: MarkupSafe>=2.0 in /usr/local/lib/python3.11/dist-packages (from jinja2->altair<6,>=4.0->streamlit)
Requirement already satisfied: attrs>=22.2.0 in /usr/local/lib/python3.11/dist-packages (from jsonschema>=3.0->altair<6,>=4.0->stre
Requirement already satisfied: jsonschema-specifications>=2023.03.6 in /usr/local/lib/python3.11/dist-packages (from jsonschema>=3.0
Requirement already satisfied: referencing>=0.28.4 in /usr/local/lib/python3.11/dist-packages (from jsonschema>=3.0->altair<6,>=4.0
Requirement already satisfied: rpds-py>=0.7.1 in /usr/local/lib/python3.11/dist-packages (from jsonschema>=3.0->altair<6,>=4.0->stre
Requirement already satisfied: six>=1.5 in /usr/local/lib/python3.11/dist-packages (from python-dateutil>=2.8.2->pandas<3,>=1.4.0->
Downloading streamlit-1.43.0-py2.py3-none-any.whl (9.7 MB)
----- 9.7/9.7 MB 30.1 MB/s eta 0:00:00
Downloading pydeck-0.9.1-py2.py3-none-any.whl (6.9 MB)
----- 6.9/6.9 MB 55.1 MB/s eta 0:00:00
Downloading watchdog-6.0.0-py3-none-manylinux2014_x86_64.whl (79 kB)
----- 79.1/79.1 kB 5.8 MB/s eta 0:00:00
Installing collected packages: watchdog, pydeck, streamlit
Successfully installed pydeck-0.9.1 streamlit-1.43.0 watchdog-6.0.0

```

```

1 # prompt: Create the interface for the above using streamlit
2
3 import streamlit as st
4 from pydub import AudioSegment
5 import speech_recognition as sr
6 import whisper
7
8 def convert_to_wav(input_file, output_file):
9     audio = AudioSegment.from_file(input_file)
10    audio.export(output_file, format='wav')
11    st.write("✅ Conversion to WAV successful!")
12
13 def transcribe_audio(audio_path):
14     recognizer = sr.Recognizer()
15     with sr.AudioFile(audio_path) as source:
16         audio = recognizer.record(source)
17     try:
18         text = recognizer.recognize_google(audio)
19         st.write("✅ Transcription Successful!\n")
20         return text
21     except sr.UnknownValueError:
22         return "⚠️ Could not understand the audio."
23     except sr.RequestError:
24         return "⚠️ Could not connect to the speech recognition service."
25
26 def convert_wav_to_mp3(wav_file, mp3_file=None):
27     audio = AudioSegment.from_wav(wav_file)
28     if not mp3_file:
29         mp3_file = wav_file.rsplit('.', 1)[0] + ".mp3"
30     audio.export(mp3_file, format="mp3")
31     st.write(f"Conversion complete: {mp3_file}")
32
33 def transcribe_audio_whisper(audio_file, model_size="base"):
34     model = whisper.load_model(model_size)
35     result = model.transcribe(audio_file)
36     transcript_file = audio_file.rsplit('.', 1)[0] + "_transcript.txt"
37     with open(transcript_file, "w", encoding="utf-8") as f:
38         f.write(result["text"])
39     st.write(f"Transcription saved to {transcript_file}")
40     return result["text"]
41
42 st.title("Audio Transcription App")
43
44 uploaded_file = st.file_uploader("Choose an audio file", type=["wav", "mp3", "ogg"])
45
46 if uploaded_file is not None:
47     # Save the uploaded file
48     with open("uploaded_audio.wav", "wb") as f:
49         f.write(uploaded_file.getvalue())
50
51     # Perform transcription
52     if st.button("Transcribe"):
53         wav_file = "uploaded_audio.wav"
54         mp3_file = "uploaded_audio.mp3"
55         convert_to_wav(wav_file, mp3_file)
56         convert_wav_to_mp3(wav_file, mp3_file)
57         transcript = transcribe_audio_whisper(mp3_file)
58         st.write("📄 Transcribed Text:\n", transcript)

```

⚠️ 2025-03-05 05:18:14.779 WARNING streamlit.runtime.scriptrunner_utils.script_run_context: Thread 'MainThread': missing ScriptRunContext

2025-03-05 05:18:14.903

Warning: to view this Streamlit app on a browser, run it with the following command:

```
streamlit run /usr/local/lib/python3.11/dist-packages/colab_kernel_launcher.py [ARGUMENTS]
2025-03-05 05:18:14.905 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2025-03-05 05:18:14.906 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2025-03-05 05:18:14.907 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2025-03-05 05:18:14.908 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2025-03-05 05:18:14.909 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
2025-03-05 05:18:14.910 Thread 'MainThread': missing ScriptRunContext! This warning can be ignored when running in bare mode.
```

```
1 !pip install pyngrok
```

Collecting pyngrok
 Downloading pyngrok-7.2.3-py3-none-any.whl.metadata (8.7 kB)
 Requirement already satisfied: PyYAML>=5.1 in /usr/local/lib/python3.11/dist-packages (from pyngrok) (6.0.2)
 Downloading pyngrok-7.2.3-py3-none-any.whl (23 kB)
 Installing collected packages: pyngrok
 Successfully installed pyngrok-7.2.3

```
1 %%writefile app.py
2
3 import streamlit as st
4 from pydub import AudioSegment
5 import speech_recognition as sr
6 import whisper
7
8 def convert_to_wav(input_file, output_file):
9     audio = AudioSegment.from_file(input_file)
10    audio.export(output_file, format='wav')
11    st.write("✅ Conversion to WAV successful!")
12
13 def transcribe_audio(audio_path):
14    recognizer = sr.Recognizer()
15    with sr.AudioFile(audio_path) as source:
16        audio = recognizer.record(source)
17    try:
18        text = recognizer.recognize_google(audio)
19        st.write("✅ Transcription Successful!\n")
20        return text
21    except sr.UnknownValueError:
22        return "⚠️ Could not understand the audio."
23    except sr.RequestError:
24        return "⚠️ Could not connect to the speech recognition service."
25
26 def convert_wav_to_mp3(wav_file, mp3_file=None):
27    audio = AudioSegment.from_wav(wav_file)
28    if not mp3_file:
29        mp3_file = wav_file.rsplit('.', 1)[0] + ".mp3"
30    audio.export(mp3_file, format="mp3")
31    st.write(f"Conversion complete: {mp3_file}")
32
33 def transcribe_audio_whisper(audio_file, model_size="base"):
34    model = whisper.load_model(model_size)
35    result = model.transcribe(audio_file)
36    transcript_file = audio_file.rsplit('.', 1)[0] + "_transcript.txt"
37    with open(transcript_file, "w", encoding="utf-8") as f:
38        f.write(result["text"])
39    st.write(f"Transcription saved to {transcript_file}")
40    return result["text"]
41
42 st.title("Audio Transcription App")
43
44 uploaded_file = st.file_uploader("Choose an audio file", type=["wav", "mp3", "ogg"])
45
46 if uploaded_file is not None:
47     # Save the uploaded file
48     with open("uploaded_audio.wav", "wb") as f:
49         f.write(uploaded_file.getvalue())
50
51     # Perform transcription
52     if st.button("Transcribe"):
53         wav_file = "uploaded_audio.wav"
54         mp3_file = "uploaded_audio.mp3"
55         convert_to_wav(wav_file, mp3_file)
56         convert_wav_to_mp3(wav_file, mp3_file)
57         transcript = transcribe_audio_whisper(mp3_file)
58         st.write("📄 Transcribed Text:\n", transcript)
59
60 import base64
61
62 # Display the transcribed text
63 st.write("📄 Transcribed Text:\n", transcript)
```



```

64
65 # Provide a download button for the transcript
66 with open("uploaded_audio_transcript.txt", "r", encoding="utf-8") as file:
67     transcript_content = file.read()
68
69 # Create a download link for the .txt file
70 st.download_button(
71     label="📄 Download Transcript as .txt",
72     data=transcript_content,
73     file_name="uploaded_audio_transcript.txt",
74     mime="text/plain"
75 )

```

Overwriting app.py

```
1 !fuser 8501/tcp -k
```

8501/tcp: 17607

```
1 !kill ngrok
```

```
1 public_url = ngrok.connect(8502) # Change to 8502 or any other free port
2 print(f"🚀 Streamlit app is live at: {public_url}")

```

Streamlit app is live at: NgrokTunnel: "<https://c8d1-34-29-69-80.ngrok-free.app>" -> "<http://localhost:8502>"

```

1 from pyngrok import ngrok
2 import threading
3 import time
4
5 # Start Streamlit in a separate thread
6 def run_streamlit():
7     !streamlit run app.py --server.port 8502
8
9 thread = threading.Thread(target=run_streamlit)
10 thread.start()
11
12 # Give Streamlit some time to start
13 time.sleep(5)
14
15 # Create a public URL using ngrok
16 public_url = ngrok.connect(8502)
17 print(f"🚀 Streamlit app is live at: {public_url}")
18

```

Collecting usage statistics. To deactivate, set browser.gatherUsageStats to false.

You can now view your Streamlit app in your browser.

Local URL: <http://localhost:8502>

Network URL: <http://172.28.0.12:8502>

External URL: <http://34.29.69.80:8502>

🚀 Streamlit app is live at: NgrokTunnel: "<https://5773-34-29-69-80.ngrok-free.app>" -> "<http://localhost:8502>"

```
1 !ngrok config add-authtoken 2tssscRRTIeUkSZwCRcBUy14amY_2rS11ZXrVrzt4kVzVriC4
```

Authtoken saved to configuration file: /root/.config/ngrok/ngrok.yml

```
1 # A wild code
```

```

1 import whisper
2
3 # Load Whisper model
4 model = whisper.load_model("base")
5
6 # Path to your audio file
7 audio_file = "/content/converted_audio.wav" # Replace with the path to your audio
8
9 # Transcribe audio
10 transcription = model.transcribe(audio_file)
11 transcription_text = transcription['text']
12
13 print("Transcription:\n", transcription_text)

```

/usr/local/lib/python3.11/dist-packages/whisper/__init__.py:150: FutureWarning: You are using `torch.load` with `weights_only=False` which is unsafe. Please update your code to use `weights_only=True` to prevent unexpected behavior from future changes to torch.load. See https://pytorch.org/docs/stable/generated/torch.load.html for more details.

```


/usr/local/lib/python3.11/dist-packages/whisper/transcribe.py:126: UserWarning: FP16 is not supported on CPU; using FP32 instead
  warnings.warn("FP16 is not supported on CPU; using FP32 instead")
Transcription:
  So I will tell a little bit about each of the three books and then I will try to strengthen them. So, Sapien talks about how at or

```

```

1 # Define the path to save the text file
2 text_file_path = "/content/transcription.txt" # Change path if needed
3
4 # Save the transcription to a text file
5 with open(text_file_path, 'w') as file:
6     file.write("Transcription:\n" + transcription_text)
7
8 print(f"Transcription saved successfully at: {text_file_path}")

```


 Transcription saved successfully at: /content/transcription.txt

Hosting it on a localhost

```

1 %%writefile app.py
2
3 import streamlit as st
4 import whisper
5 import os
6
7 # Set page configuration
8 st.set_page_config(page_title="Speech-to-Text App", page_icon="🗣️")
9
10 # Title and description
11 st.title("🗣️ Speech-to-Text Transcription")
12 st.write("Upload an audio file to get the transcription as text!")
13
14 # Upload audio file
15 audio_file = st.file_uploader("Upload Audio", type=["wav", "mp3", "m4a"])
16
17 # Transcription function
18 def transcribe_audio(file_path):
19     # Load Whisper model
20     model = whisper.load_model("base")
21     # Transcribe audio
22     transcription = model.transcribe(file_path)
23     return transcription['text']
24
25 if audio_file is not None:
26     # Save uploaded file temporarily
27     with open("uploaded_audio.wav", "wb") as f:
28         f.write(audio_file.getbuffer())
29
30     st.audio(audio_file, format='audio/wav')
31
32     # Perform transcription
33     with st.spinner("Transcribing... Please wait..."):
34         transcription_text = transcribe_audio("uploaded_audio.wav")
35         st.success("Transcription complete!")
36
37     # Display transcription
38     st.text_area("Transcription:", transcription_text, height=300)
39
40     # Save transcription to a text file
41     text_file_path = "transcription.txt"
42     with open(text_file_path, 'w') as file:
43         file.write("Transcription:\n" + transcription_text)
44
45     # Provide download link for the text file
46     with open(text_file_path, 'rb') as f:
47         st.download_button(
48             label="Download Transcription as Text File",
49             data=f,
50             file_name="transcription.txt",
51             mime="text/plain"
52         )
53
54     # Clean up temporary files
55     os.remove("uploaded_audio.wav")
56     os.remove(text_file_path)
57 else:
58     st.info("Please upload an audio file to proceed.")

```

 Overwriting app.py


```
1 # Install and run ngrok
2 !pip install pyngrok
3 !ngrok authtoken YOUR_NGROK_AUTH_TOKEN # Replace YOUR_NGROK_AUTH_TOKEN with your ngrok token
4
5 # Run Streamlit in the background
6 import threading
7 import subprocess
8 from pyngrok import ngrok
9
10 # Start Streamlit app
11 def run_streamlit():
12     subprocess.run(["streamlit", "run", "app.py"])
13
14 thread = threading.Thread(target=run_streamlit)
15 thread.daemon = True
16 thread.start()
17
18 # Create a public URL using ngrok
19 public_url = ngrok.connect(8501)
20 print("Streamlit app is live at:", public_url)
```

```

Requirement already satisfied: pyngrok in /usr/local/lib/python3.11/dist-packages (7.2.3)
Requirement already satisfied: PyYAML>=5.1 in /usr/local/lib/python3.11/dist-packages (from pyngrok) (6.0.2)
Authtoken saved to configuration file: /root/.config/ngrok/ngrok.yml
WARNING:pyngrok.process.ngrok:t=2025-03-05T06:40:41+0000 lvl=warn msg="failed to start tunnel" pg=/api/tunnels id=edf20c373fc4e6c5

```

```

HTTPError                                Traceback (most recent call last)
/usr/local/lib/python3.11/dist-packages/pyngrok/ngrok.py in api_request(url, method, data, params, timeout, auth)
    556     try:
--> 557         response = urlopen(request, encoded_data, timeout)
    558         response_data = response.read().decode("utf-8")

```

8 frames

HTTPError: HTTP Error 502: Bad Gateway

During handling of the above exception, another exception occurred:

```
PyngrokNgrokHTTPError                                Traceback (most recent call last)
/usr/local/lib/python3.11/dist-packages/pyngrok/ngrok.py in api_request(url, method, data, params, timeout, auth)
    576         logger.debug(f"Response {status_code}: {response_data.strip()}")
    577
--> 578         raise PyngrokNgrokHTTPError(f"ngrok client exception, API returned {status_code}: {response_data}",
    579                                   e.url,
    580                                   status_code, e.reason, e.headers, response_data)
```

```
PyngrokNgrokHTTPError: ngrok client exception, API returned 502: {"error_code":103,"status_code":502,"msg":"failed to start tunnel","details":{"errn":"failed to start tunnel: Your account may not run more than 3 tunnels over a single ngrok agent session.\n\nCgEteVnEgUyQ0lrmeE47u3JSucN\r\n\r\n\r\nERR NGROK 324\r\n\r\n"}}
```

```
1 !kill ngrok
```

```
1 !curl http://localhost:4040/api/tunnels
```

```
➡ curl: (7) Failed to connect to localhost port 4040 after 0 ms: Connection refused
```

```
1 !npm install -g localtunnel
```

```
➔ "": 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100% | 100%  
added 22 packages in 3s  
..  
..3 packages are looking for funding  
.. run `npm fund` for details  
..
```

```
1 # Run Streamlit app and use localtunnel for a public URL
2 import threading
3 import subprocess
4
5 # Start Streamlit in the background
6 def run_streamlit():
7     subprocess.run(["streamlit", "run", "app.py", "--server.port=8501"])
8
9 thread = threading.Thread(target=run_streamlit)
10 thread.daemon = True
11 thread.start()
12
```

```
13 # Use localtunnel to expose the Streamlit app
14 !lt --port 8501 --subdomain $(openssl rand -hex 4)
```

🔗 your url is: <https://b3230a0f.local.lt>
^C

```
1 !pip install cloudflared
```

🔗 Collecting cloudflared
 Downloading cloudflared-1.0.0.2.tar.gz (2.6 kB)
 Preparing metadata (setup.py) ... done
 Collecting setuptools_scm (from cloudflared)
 Downloading setuptools_scm-8.2.0-py3-none-any.whl.metadata (6.8 kB)
 Requirement already satisfied: packaging>=20 in /usr/local/lib/python3.11/dist-packages (from setuptools_scm->cloudflared) (24.2)
 Requirement already satisfied: setuptools>=61 in /usr/local/lib/python3.11/dist-packages (from setuptools_scm->cloudflared) (75.1.0)
 Downloading setuptools_scm-8.2.0-py3-none-any.whl (43 kB)
 43.9/43.9 kB 1.1 MB/s eta 0:00:00
 Building wheels for collected packages: cloudflared
 Building wheel for cloudflared (setup.py) ... done
 Created wheel for cloudflared: filename=cloudflared-1.0.0.2-py3-none-any.whl size=2984 sha256=ff03d65633f072a985d3d87a5ab13b7ee8d7
 Stored in directory: /root/.cache/pip/wheels/3c/9f/f1/ef5e36c9386d737ac05ab8714d611c430d79abe55d862ca2b7
 Successfully built cloudflared
 Installing collected packages: setuptools_scm, cloudflared
 Successfully installed cloudflared-1.0.0.2 setuptools_scm-8.2.0

```
1 !wget https://github.com/cloudflare/cloudflared/releases/latest/download/cloudflared-linux-amd64.deb
2 !dpkg -i cloudflared-linux-amd64.deb
```

🔗 --2025-03-05 06:52:44-- <https://github.com/cloudflare/cloudflared/releases/latest/download/cloudflared-linux-amd64.deb>
 Resolving github.com (github.com)... 140.82.113.4
 Connecting to github.com (github.com)|140.82.113.4|:443... connected.
 HTTP request sent, awaiting response... 302 Found
 Location: <https://github.com/cloudflare/cloudflared/releases/download/2025.2.1/cloudflared-linux-amd64.deb> [following]
 --2025-03-05 06:52:45-- <https://github.com/cloudflare/cloudflared/releases/download/2025.2.1/cloudflared-linux-amd64.deb>
 Reusing existing connection to github.com:443.
 HTTP request sent, awaiting response... 302 Found
 Location: <https://objects.githubusercontent.com/github-production-release-asset-2e65be/106867604/4f284b8f-4452-49e8-a0c0-0e45c3b6e6f5>
 --2025-03-05 06:52:45-- <https://objects.githubusercontent.com/github-production-release-asset-2e65be/106867604/4f284b8f-4452-49e8-a0c0-0e45c3b6e6f5>
 Resolving objects.githubusercontent.com (objects.githubusercontent.com)... 185.199.110.133, 185.199.111.133, 185.199.108.133, ...
 Connecting to objects.githubusercontent.com (objects.githubusercontent.com)|185.199.110.133|:443... connected.
 HTTP request sent, awaiting response... 200 OK
 Length: 18559760 (18M) [application/octet-stream]
 Saving to: 'cloudflared-linux-amd64.deb'

```
cloudflared-linux-a 100%[=====>] 17.70M 62.6MB/s in 0.3s
```

```
2025-03-05 06:52:45 (62.6 MB/s) - 'cloudflared-linux-amd64.deb' saved [18559760/18559760]
```

```
Selecting previously unselected package cloudflared.
(Reading database ... 124947 files and directories currently installed.)
Preparing to unpack cloudflared-linux-amd64.deb ...
Unpacking cloudflared (2025.2.1) ...
Setting up cloudflared (2025.2.1) ...
Processing triggers for man-db (2.10.2-1) ...
```

```
1 import threading
2 import subprocess
3
4 # Start Streamlit app in the background
5 def run_streamlit():
6     subprocess.run(["streamlit", "run", "app.py", "--server.port=8501"])
7
8 thread = threading.Thread(target=run_streamlit)
9 thread.daemon = True
10 thread.start()
11
12 # Expose the app using cloudflared
13 !cloudflared tunnel --url http://localhost:8501
```

🔗 Exception in thread Thread-4 (run_streamlit):
 Traceback (most recent call last):
 File "/usr/lib/python3.11/threading.py", line 1045, in _bootstrap_inner
 self.run()
 File "/usr/lib/python3.11/threading.py", line 982, in run
 self._target(*self._args, **self._kwargs)
 File "<ipython-input-1-cd828985feed>", line 6, in run_streamlit
 File "/usr/lib/python3.11/subprocess.py", line 548, in run
 with Popen(*popenargs, **kwargs) as process:
 ^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
 File "/usr/lib/python3.11/subprocess.py", line 1026, in __init__
 self._execute_child(args, executable, preexec_fn, close_fds,
 File "/usr/lib/python3.11/subprocess.py", line 1955, in _execute_child
 raise child_exception_type(errno_num, err_msg, err_filename)

```
FileNotFoundError: [Errno 2] No such file or directory: 'streamlit'  
/bin/bash: line 1: cloudflared: command not found
```

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