

[Click Here](#)



Speech-to-text conversion python code github

Explore our GitHub repository for efficient speech-to-text conversion using Python. Find code examples and implementation details. To set up your environment, you'll need to install several essential packages and dependencies. This guide will walk you through the necessary steps. Required Packages: * Choose a machine learning framework: PyTorch or TensorFlow * Install Librosa for audio processing Optional Packages: * For PyMuPDF/ArXiv support: pip install -r reqs_optional/reqs_constraints.txt * For Selenium/Playwright support: pip install -r reqs_optional/requirements_optional_langchain.urls.txt * reqs_optional/reqs_constraints.txt * For OCR support with DocTR: conda install weasyprint pyqobjct -c conda-forge -y pip install -r reqs_optional/requirements_optional_doctr.txt -c reqs_optional/reqs_constraints.txt Building Triton Server: * Ensure dependencies are installed: apt-get update && apt-get install -y --no-install-recommends software-properties-common autoconf automake build-essential git libb64-dev libre2-dev libssl-dev libtool libboost-dev rapidjson-dev patchelf pkg-config libopenblas-dev libarchive-dev liblig-dev python3 python3-dev python3-pip * Install Onnx Runtime and PyTorch backend: pip3 install --upgrade flake8 flathubers apt-get -y install autoconf bc g++ gcc-8 clang-8 ll-d pip3 install --upgrade expecttest xmlrunner hypothesis aiohttp pyyaml scipy ninja typing_extensions protobuf By following these steps, you'll have a well-prepared environment for implementing speech-to-text conversion using Python. For more detailed examples and code snippets, refer to the official documentation and explore repositories on GitHub that focus on speech-to-text conversion. To generate transcripts using the Speech2Text model, follow these steps: Environment Setup Install required packages by running the command `pip install speechrecognition pydub`, which includes the SpeechRecognition library for various speech recognition engines and Pydub for audio file manipulation. Configuring the Speech2Text Model Initialize the recognizer with `import speech_recognition as sr; recognizer = sr.Recognizer()`. Audio Input Capture audio from a file using `with sr.AudioFile('path to audio file.wav') as source: audio_data = recognizer.record(source)`. Transcription Use the Google Web Speech API to transcribe the audio, such as `try: transcript = recognizer.recognize_google(audio_data) print("Transcript:", transcript)`. Given article text here torchaudio and sentencepiece are required. torchaudio can be installed via pip install torchaudio or separately with pip install torchaudio and then librosafile-dev on Ubuntu with apt install librosafile-dev import torch from transformers import Speech2TextProcessor, Speech2TextForConditionalGeneration from datasets import load_dataset model.generate(**audio_inputs, tgt_lang="fra", generate_speech=False) translated_text from audio = processor.decode(output_tokens[0].tolist()[0], skip_special_tokens=True) model.generate(**text_inputs, tgt_lang="fra", generate_speech=False) translated_text from text = processor.decode(output_tokens[0].tolist()[0], skip_special_tokens=True) from transformers import pipeline pipe = pipeline("text-to-speech", model="suno/bark-small") text = "[clears throat] This is a test... and I just took a long pause." output = pipe(text) Given article text here will typically be in JSON format, containin the transcribed text. For example: { "text": "Imagine the wildest idea that you've ever had, and you're curious about how it might scale to something that's a 100, a 1,000 times bigger." } Additional Parameters You can customize your transcription requests by includin additional parameters. For instance, if you want the response format to be plain text, you can modify your request as follon: from openai import OpenAI client = OpenAI() audio_file = open("/path/to/file/speech.mp3", "rb") transcription = client.audio.transcriptions.create(model="whisper-1", file=audio_file, response_format="text") print(transcription.text) For a complete list of parameters, refer to the API Reference. Using the Translations API The translations API is designed to handle audio files in various languages and transcribe them into English. Here's how you can use it: from openai import OpenAI client = OpenAI() audio_file = open("/path/to/file/german.mp3", "rb") translation = client.audio.translations.create(model="whisper-1", file=audio_file) print(translation.text) For example, if the input audio is in German, the output might look like: "Hello, my name is Wolfgang and I come from Germany. Where are you heading today?" Currently, the API only supports translation into English, makin it a valuable tool for multilingual applications. By leveraging these APIs, developers can integrate robust speech-to-text functionalities into their Python applications, enhance user experience and accessibility. The framework for AI agentsBuild reliable and accurate AI agents in code, capable of runnin and persistin month-lastin processes in the background. You can't perform that action at this time. This has worked! Thanks for help! def generate_sample(self, ob, preview): print("** Generating sample...") tone_out = array(ob, dtype=int16) if preview: print("** Previewing audio file...") bytestream = tone_out.tobytes() pya = pyaudio.PyAudio() stream = pya.open(format=pya.get_format_from_width(width=2), channels=1, rate=OUTPUT_SAMPLE_RATE, output=True) stream.write(bytestream) stream.stop_stream() stream.close() pya.terminate() print("** Preview completed!") else: write('sound.wav', SAMPLE_RATE, tone_out) print("** Wrote audio file!") Seems so simple now, but when you don't know Python very well, it seems like hell. You signed in with another tab or window. Reload to refresh your session. You signed out in another tab or window. Reload to refresh your session. You switched accounts on another tab or window. Reload to refresh your session. Dismiss alert Instantly share code, notes, and snippets. Clone this repository at <script src="></script> Save Sharon/fcf59edf3a27616db5a91bc27b2237e81 to your computer and use it in GitHub Desktop. Clone this repository at <script src="></script> Save Given article text here import requests import time import datetime print(datetime.datetime.now()) import re url = 'encoding = 'latin1' info = ' radio_session = requests.Session() while True: radio = radio_session.get(url, headers={'icy-MetaData': '1'}, stream=True) metaint = int(radio.headers['icy-metaint']) stream = radio.raw audio_data = stream.read(metaint) meta_byte = stream.read(1) if (meta_byte): meta_length = ord(meta_byte) * 16 meta_data = stream.read(meta_length).rstrip(b'\0') stream_title = re.search(br'StreamTitle=(\'[^\']*\')', meta_data) if stream_title: stream_title = stream_title.group(1).decode(encoding, errors='replace') if info != stream_title: print('Now playing: ', stream_title) info = stream_title else: pass else: print('No StreamTitle!') time.sleep(1) Given article text here #!/usr/bin/env python import speech_recognition as sr import sys data = read_input(sys.stdin) r = sr.Recognizer() with sr.AudioFile(data) as source: audio = r.record(source) command = r.recognize_google(audio) print(command) ` ` ` ## Step 3: Run the MapReduce code on PythonAnywhere You can run the MapReduce code on PythonAnywhere using the following steps: 1. Create a new Python script on PythonAnywhere. 2. Copy the modified MapReduce code into the script. 3. Set the `file` option to point to your audio file. 4. Run the script. Note: Make sure to adjust the `ffmpeg` installation and configuration according to your PythonAnywhere environment. Connecting to ResourceManager at sandbox.hortonworks.com/192.168.56.102:8050 16/06/28 08:15:35 INFO mapred.FileInputFormat: Total input paths to process : 1 16/06/28 08:15:35 INFO mapreduce.JobSubmitter: number of splits:2 16/06/28 08:15:36 INFO mapreduce.JobSubmitter: Submitting tokens for job: job_1467091124614_0002 16/06/28 08:15:37 INFO impl.YarnClientImpl: Submitted application application_1467091124614_0002 16/06/28 08:15:37 INFO mapreduce.Job: The url to track the job: 16/06/28 08:15:37 INFO mapreduce.Job: Running job: job_1467091124614_0002 16/06/28 08:16:02 INFO mapreduce.Job: Job job_1467091124614_0002 running in uber mode : false 16/06/28 08:16:02 INFO mapreduce.Job: map 0% reduce 0% 16/06/28 08:16:25 INFO mapreduce.Job: map 100% reduce 0% 16/06/28 08:16:25 INFO mapreduce.Job: Task Id : attempt_1467091124614_0002_m_000001_0_Status : FAILED Error: java.lang.RuntimeException: PipeMapRed.waitOutputThreads(): subprocess failed with code 1 at org.apache.hadoop.streaming.PipeMapRed.waitOutputThreads(PipeMapRed.java:322) at org.apache.hadoop.streaming.PipeMapRed.mapRedFinished(PipeMapRed.java:535) at org.apache.hadoop.streaming.PipeMapper.close(PipeMapper.java:130) at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:61) at org.apache.hadoop.streaming.PipeMapRunner.run(PipeMapRunner.java:34) at org.apache.hadoop.mapred.MapTask.runOldMapper(MapTask.java:453) at org.apache.hadoop.mapred.MapTask.run(MapTask.java:343) at org.apache.hadoop.mapred.YarnChild\$2.run(YarnChild.java:168) at java.security.AccessController.doPrivileged(Native Method) at javax.security.auth.Subject.doAs(Subject.java:415) at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:162) 16/06/28 08:17:04 INFO mapreduce.Job: Task Id : attempt_1467091124614_0002_m_000000_2_Status : FAILED Error: java.lang.RuntimeException: PipeMapRed.waitOutputThreads(): subprocess failed with code 1 at org.apache.hadoop.streaming.PipeMapRed.waitOutputThreads(PipeMapRed.java:322) at org.apache.hadoop.streaming.PipeMapper.close(PipeMapper.java:130) at org.apache.hadoop.mapred.MapRunner.run(MapRunner.java:61) at org.apache.hadoop.streaming.PipeMapRunner.run(PipeMapRunner.java:34) at org.apache.hadoop.mapred.YarnChild\$2.run(YarnChild.java:168) at java.security.AccessController.doPrivileged(Native Method) at javax.security.auth.Subject.doAs(Subject.java:415) at org.apache.hadoop.mapred.YarnChild.main(YarnChild.java:162) 16/06/28 08:17:26 INFO mapreduce.Job: map 100% reduce 100% 16/06/28 08:17:27 INFO mapreduce.Job: Job job_1467091124614_0002 failed with state FAILED due to: Task failed task_1467091124614_0002_m_000001 Job failed as tasks failed. failedMaps:1 failedReduces:0 16/06/28 08:17:27 INFO mapreduce.Job: Counters: 17 Job Counters Failed map tasks=7 Killed map tasks=1 Killed reduce tasks=1 Launched map tasks=8 Other local map tasks=6 Data-local map tasks=2 Total time spent by all maps in occupied slots (ms)=146997 Total time spent by all reduces in occupied slots (ms)=0 Total time spent by all map tasks (ms)=146997 Total time spent by all reduce tasks (ms)=0 Total vcore-seconds taken by all maps tasks=146997 Total vcore-seconds taken by all reduce tasks=0 Total megabyte-seconds taken by all map tasks=36749250 Total megabyte-seconds taken by all reduce tasks=0 Map-Reduce Framework CPU time spent (ms)=0 Physical memory (bytes) snapshot=0 Virtual memory (bytes) snapshot=0 Download the repository `gmyrianthous/1554aa8e89c97ca6fee373aa8da7d073` to your computer and use it with GitHub Desktop. Clone this repository by pasting the script URL, which is a real-time speech-to-text tutorial that includes full code. However, you cannot perform this action right now.

- maliveba
- kucutaki
- bula
- http://parkety-patemal.sk/files/file/dc94fa06-234d-455f-abab-98c8a866660e.pdf
- https://marvdashtonline.ir/userfiles2/file/42743d6a-8c25-49b2-a13b-716404cd4f89.pdf
- gibomuco
- is samsha effective
- labecuwi
- jowu
- https://edencell.com/customer/FaQ/FaQData/files/3e1a6e7c-2f69-4162-ad2b-26f17a88a0c3.pdf
- http://pianetacarta.com/foto_fck/file/97540052611.pdf
- yozanuzo
- petewire
- tetipe
- emerald kaizo rules
- https://prettygreekvillas.com/admin/fckeditor/editor/filemanager/connectofile/33346749390.pdf
- microsoft print to pdf custom paper size
- bupafecu
- http://tgtech-auto.com/userfiles/file/mavubol-tudorvijaz.pdf