

AUTOMATIC SPEECH RECOGNIZATION - (English)

GROUP 2: APARANJITHA KOTHINTI | PARI DHANDAPANI | UDAYKUMAR BKVENKATA

Innovative AI project designed to push the boundaries of technology
and drive transformative change in various industries.



PROBLEM STATEMENT

Developing an Automatic Speech Recognition (ASR) system aimed at accurately transcribing spoken language into written text across diverse domains and accents. The primary focus is on enhancing recognition performance in challenging acoustic environments and effectively managing out-of-vocabulary words.

This problem statement tackles two main challenges in ASR:

1. Speech Recognition Performance in Challenging Acoustic Conditions
2. Dealing with Out-of-Vocabulary (OOV) Words

Meta Information considered for model
Training and testing:

- Data Language: English
- 300 hrs of Transcribed Data
- Format of data: Wav
- Sample Rate : 16k

Kaldi Installation

03

Clone repo from <https://github.com/kaldi-asr/kaldi>

Steps:

```
sudo apt update
sudo apt install -y cmake sox ffmpeg g++ automake autoconf libtool
subversion git zlib1g-dev unzip gfortran python2.7 python3 gawk
```

ffmpeg: resample data to desired Hz

```
cd kaldi/tools
extras/check_dependencies.sh
extras/install_mkl.sh
```

RaspberryPI:
make -j 'nproc'

```
make
extras/install_irstlm.sh
./install_srilm.sh <name> <organisation> <email> <address>
Ex: ./install_srilm.sh a123 b123 acb@gmail.com
812345 (command i executed)
```



1. Installation of kaldi on Raspberry device with Ubuntu OS
2. Defined a new module/setup to fix srilm on Ubuntu
3. Patch ups on few utilities and libraries
4. Conversation of FLAC to WAV, as changing extension will throw error while MFCC feature extraction. Used Audiosegment to convert to bytes and export approach

Tools & Infra:

- Programming : Python, Pip
- ASR: Kaldi, AudioSegment
- Front end: Steamlint, Python
- Back end: fastAPI
- Infrastructure: Azure cloud
- Operating and system: Ubuntu OS in RaspberryPI5

Data Preparation

04

Automatic Speech Recognition (ASR) using Hidden Markov Models (HMMs) and Gaussian Mixture Models (GMMs) is a conventional method extensively employed in the field. Here's an overview of the approach involved in ASR using HMMs and GMMs:

1. Acoustic Modelling:

The acoustic model captures the relationship between speech features and units. In the HMM-GMM approach, GMMs are commonly used. Each phonetic unit is represented by a GMM, modelling the probability distribution of speech features for that unit.

Wave.scp

Text (transcripts)

utt2spk (utterance to speaker mapping)

2. Language Modelling:

It serves as a critical component in automatic speech recognition (ASR) for navigating linguistic constraints and enhancing recognition accuracy. These models encompass the statistical patterns of language, facilitating the identification of the most probable word sequences based on observed speech features. Both N-gram language models and more sophisticated alternatives such as hidden Markov models or neural networks can be employed for this purpose.

Lexican (pronunciation dictionary)

non-silence_phones

optional_silence

silence_phones

Pre Data preparation:

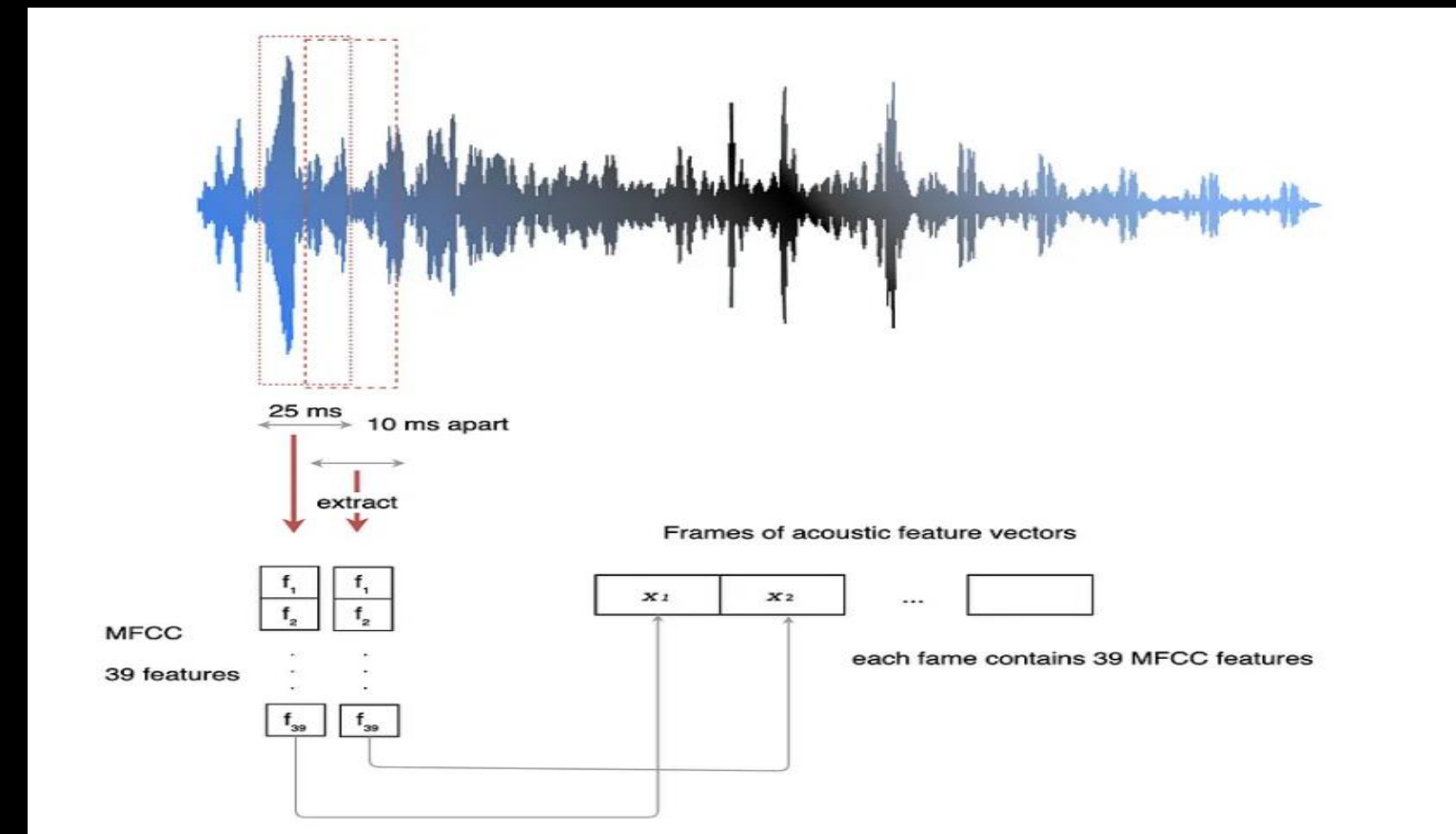
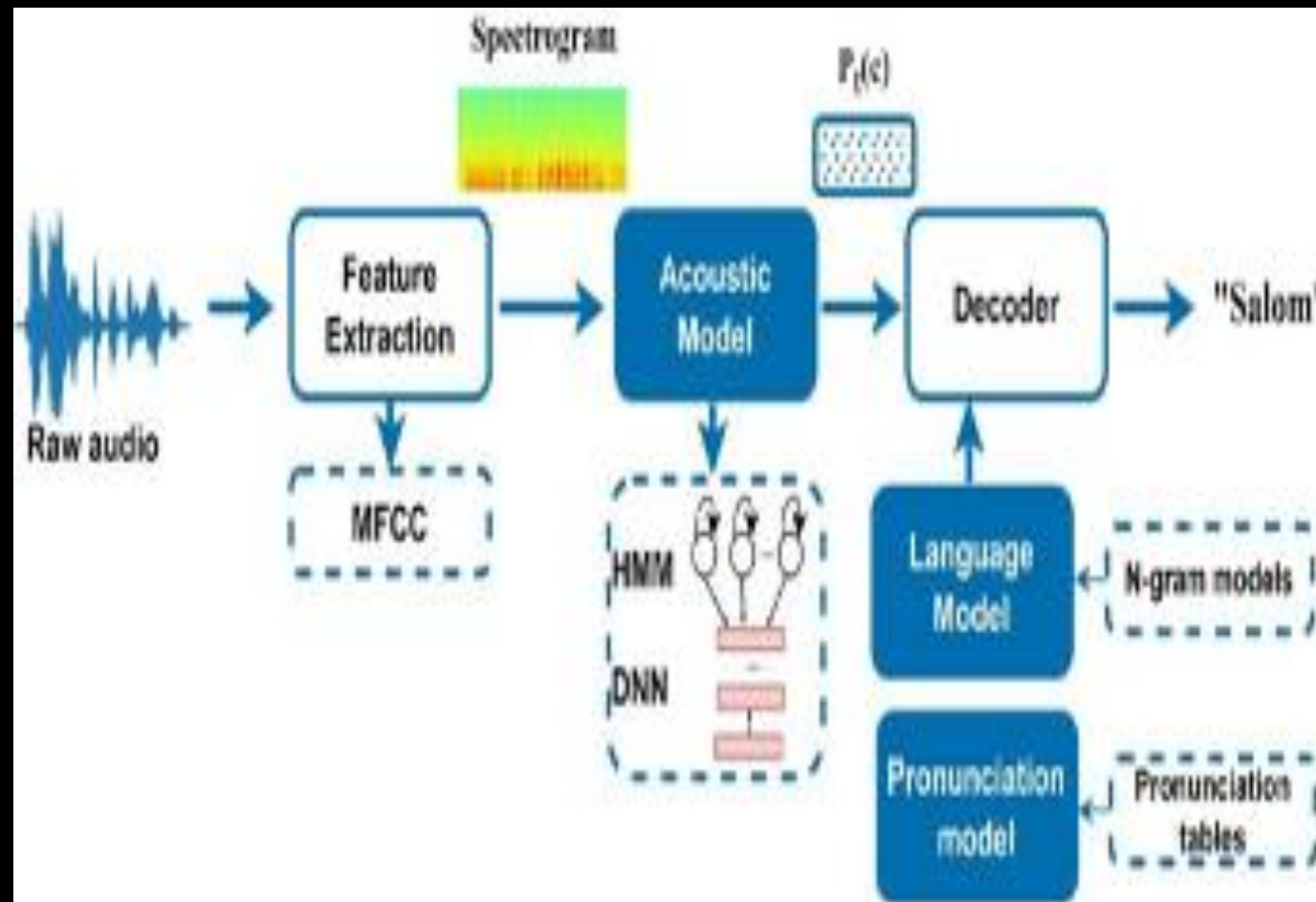
Conversion of FLAC files to WAV using AudioSegment library

Modelling

05

Automatic Speech Recognition (ASR) using Hidden Markov Models (HMMs) and Gaussian Mixture Models (GMMs) is a conventional method extensively employed in the field. Here's an overview of the approach involved in ASR using HMMs and GMMs:

- Feature Extraction – MFCC from the audio



- Used GMM-HMM Model (2000 HMM states)
- For LM, used SRILM with n-gram as 3

SRILM package with fix for Raspberrypi5:

https://drive.google.com/file/d/1j4pcfoXWMATdM7uF4nCJs-m7DBkjQb8p/view?usp=drive_link

Word Error Rate

06

$$\text{WER} = \left(\frac{\text{Substitutions} + \text{Deletions} + \text{Insertions}}{\text{Total number of words on reference}} \right)$$

Example Original Audio:

The **quick** brown fox jumps over the **lazy** dog



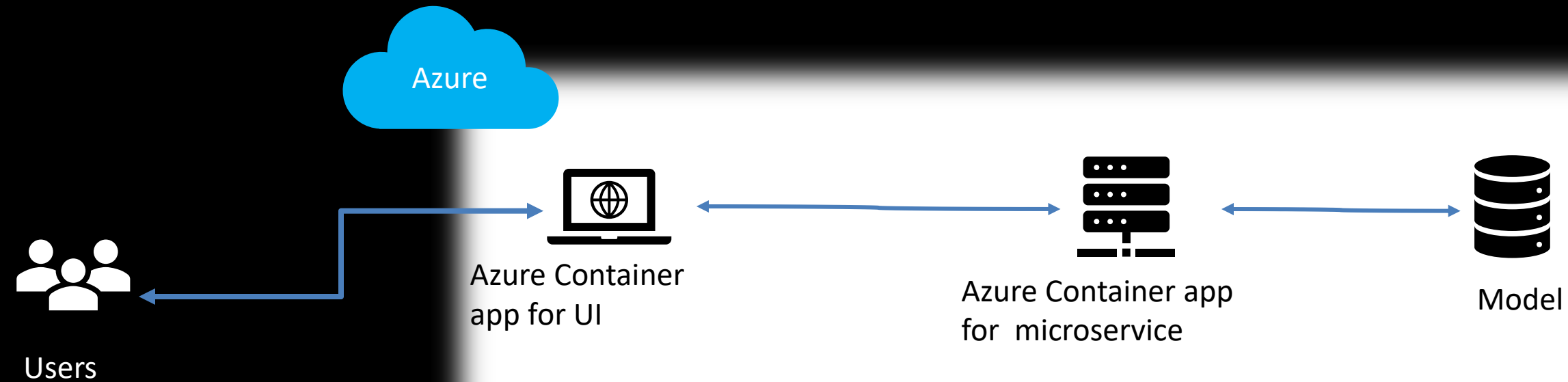
Example Transcribed Audio:

The brown fox jumps **again** over the **crazy** dog

WER of the ASR model with test data = 29 %

ASR System Architecture

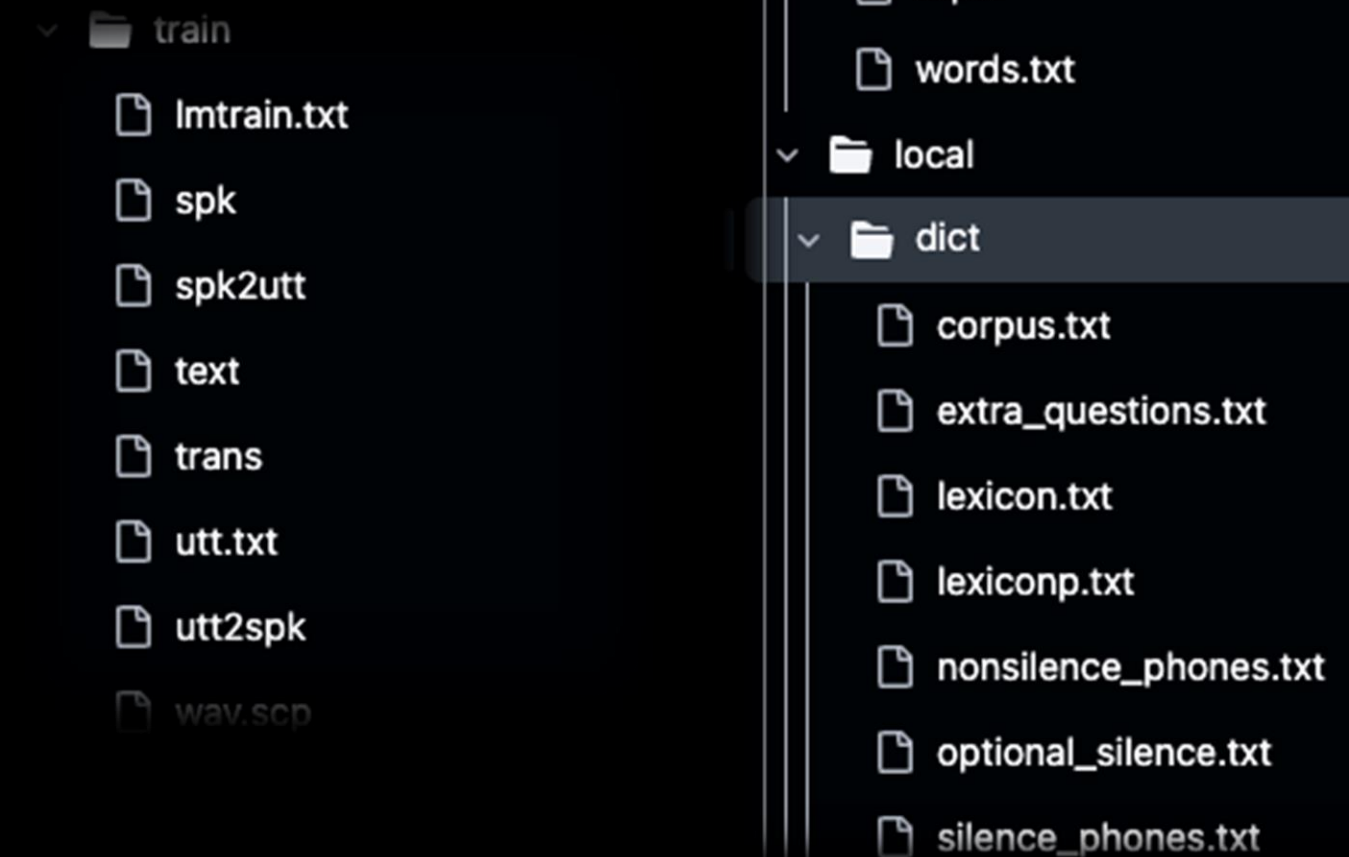
07



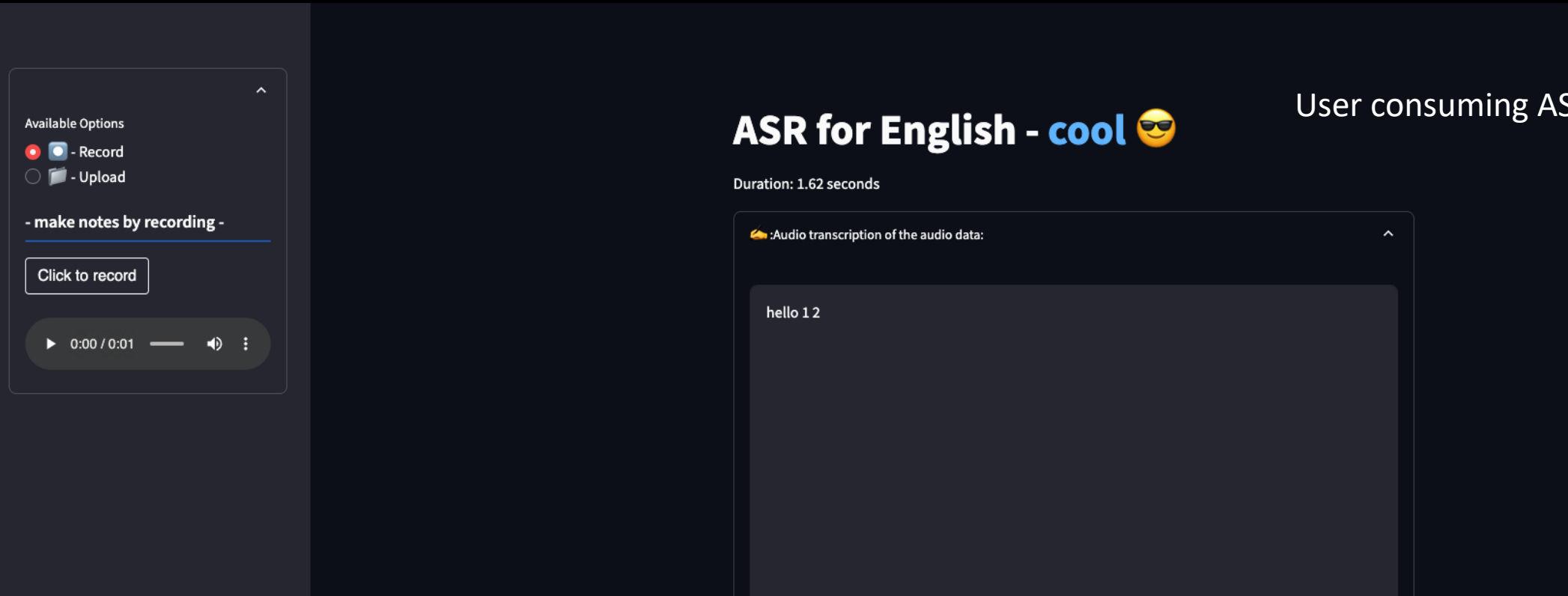
ASR DEMO

Git Hub Code Repo:

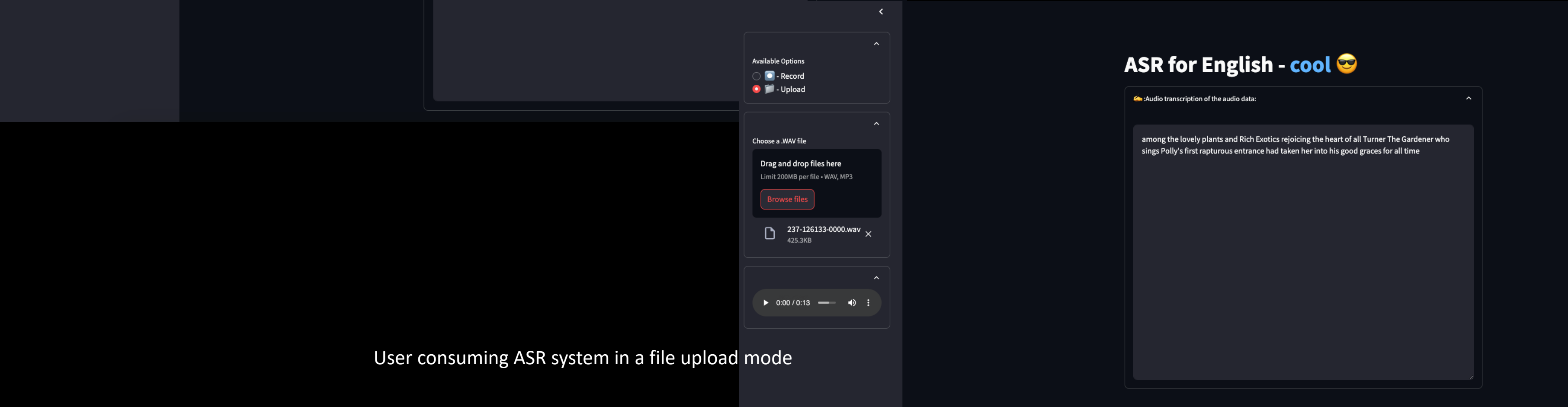
- ASR Model : [uday160386/ai_ml_asr_text \(github.com\)](https://github.com/uday160386/ai_ml_asr_text)
- ASR UI : [uday160386/asr_capstone_en_ui \(github.com\)](https://github.com/uday160386/asr_capstone_en_ui)
- ASR Microservice: [uday160386/asr_capstone_en_ms \(github.com\)](https://github.com/uday160386/asr_capstone_en_ms)



ASR System Screenshots



User consuming ASR system in a recording mode



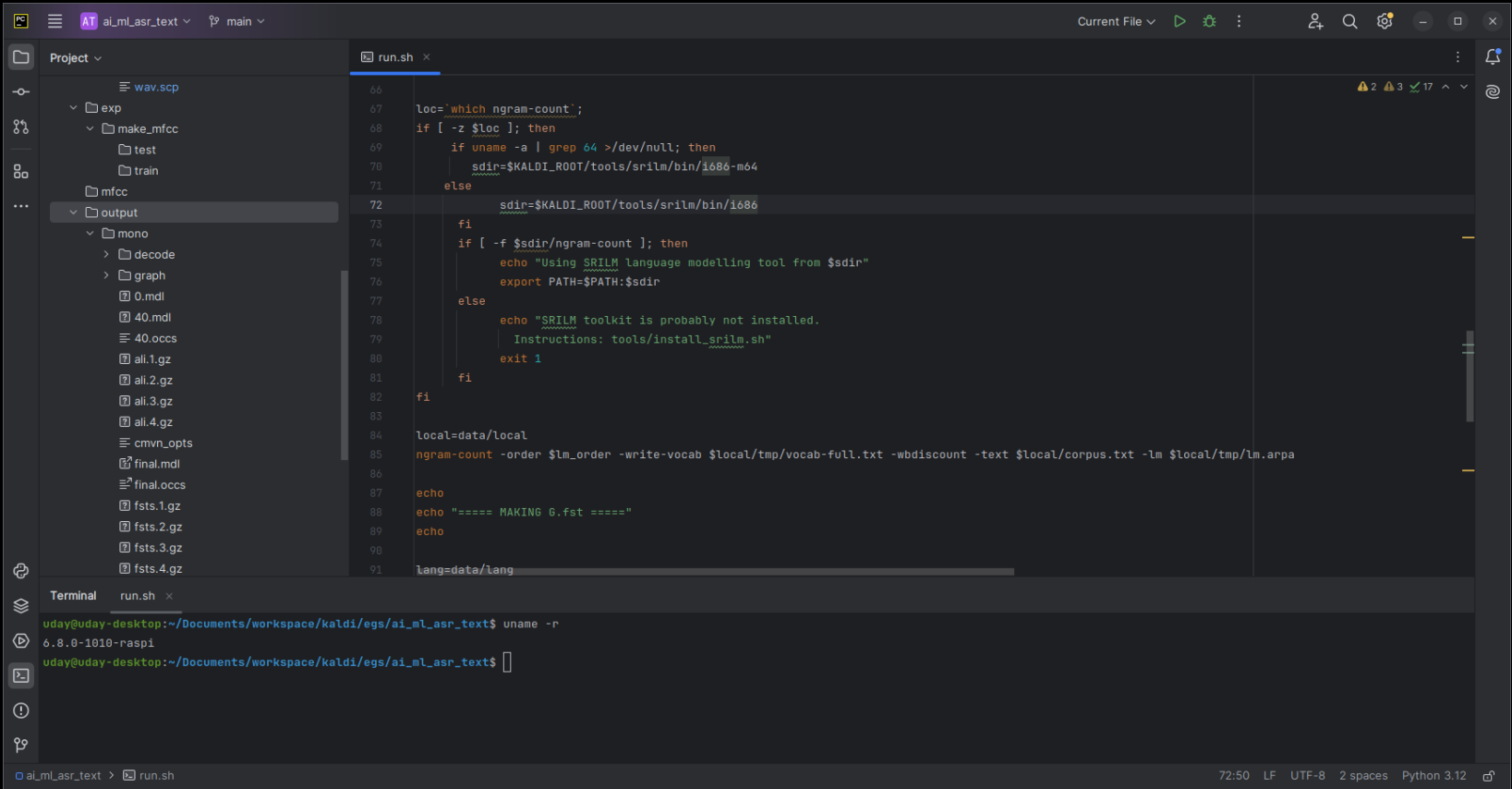
User consuming ASR system in a file upload mode

KEY MILESTONES

09

This project timeline are divided into key milestones, including platform design, development, testing, and deployment.

- Kaldi Installation Completed
- Data Preparation Completed
- Development of Model:
- Training model:
- Test model :
- Demo:



ASR using ESPnet

10

```
+ Code + Text
Verifying transaction: done
Executing transaction: done
15m
./activate_python.sh && python3 -m pip install -U numba
Requirement already satisfied: numba in ./anaconda/envs/espnet/lib/python3.9/site-packages (0.60.0)
Requirement already satisfied: llvmlite<0.44,>=0.43.0dev0 in ./anaconda/envs/espnet/lib/python3.9/site-packages (from numba) (0.43.0)
Requirement already satisfied: numpy<2.1,>=1.22 in ./anaconda/envs/espnet/lib/python3.9/site-packages (from numba) (2.0.1)
touch numba.done
./activate_python.sh && ./installers/install_torch.sh "true" "1.12.1" "11.6"
2024-09-24T22:28:40 (install_torch.sh:146:main) [INFO] python_version=3.9.19
2024-09-24T22:28:40 (install_torch.sh:147:main) [INFO] torch_version=1.12.1
2024-09-24T22:28:40 (install_torch.sh:148:main) [INFO] cuda_version=11.6
2024-09-24T22:28:53 (install_torch.sh:90:install_torch) conda install -y pytorch=1.12.1 torchaudio=0.12.1 cudatoolkit=11.6 -c pytorch -c conda-forge
KeyError('active_prefix_name')
Traceback (most recent call last):
  File "/content/drive/MyDrive/asr/espnet/tools/anaconda/envs/espnet/lib/python3.9/site-packages/conda/exception_handler.py", line 18, in __call__
    return func(*args, **kwargs)
  File "/content/drive/MyDrive/asr/espnet/tools/anaconda/envs/espnet/lib/python3.9/site-packages/conda/cli/main.py", line 52, in main_subshell
    from .conda_argparse import do_call, generate_parser, generate_pre_parser
  File "/content/drive/MyDrive/asr/espnet/tools/anaconda/envs/espnet/lib/python3.9/site-packages/conda/cli/conda_argparse.py", line 51, in <module>
    from .main_create import configure_parser as configure_parser_create
  File "/content/drive/MyDrive/asr/espnet/tools/anaconda/envs/espnet/lib/python3.9/site-packages/conda/cli/main_create.py", line 16, in <module>
    from ..notices import notices
  File "/content/drive/MyDrive/asr/espnet/tools/anaconda/envs/espnet/lib/python3.9/site-packages/conda/notices/_init_.py", line 3, in <module>
    from .core import notices # noqa: F401
  File "/content/drive/MyDrive/asr/espnet/tools/anaconda/envs/espnet/lib/python3.9/site-packages/conda/notices/core.py", line 15, in <module>
    from .. import cache, fetch_views
```

```
[46] !./asr.sh --stage 4 --stop_stage 4 --train_set train_nodev --valid_set train_dev --test_sets "train_dev test"

2024-09-24T23:03:12 (asr.sh:283:main) ./asr.sh --stage 4 --stop_stage 4 --train_set train_nodev --valid_set train_dev --test_sets train_dev test
2024-09-24T23:03:15 (asr.sh:321:main) Info: The valid_set 'train_dev' is included in the test_sets. '--eval_valid_set true' is set and 'train_dev' is removed from the test_sets.
2024-09-24T23:03:15 (asr.sh:564:main) Skipped stages: 9 14 15
2024-09-24T23:03:15 (asr.sh:799:main) Stage 4: Remove long/short data: dump/raw/org -> dump/raw
utils/copy_data_dir.sh: copied data from dump/raw/org/train_nodev to dump/raw/train_nodev
utils/validate_data_dir.sh: no such file dump/raw/train_nodev/wav.scp (if this is by design, specify --no-wav)
```

Stage 5: Generate token_list from dump/raw/train_nodev/text using BPE.

This is important for text processing. Here, we make a dictionary simply using the English characters. We use the `sentencepiece` toolkit developed by Google.

```
!./asr.sh --stage 5 --stop_stage 5 --train_set train_nodev --valid_set train_dev --test_sets "train_dev test"

2024-09-24T23:03:23 (asr.sh:283:main) ./asr.sh --stage 5 --stop_stage 5 --train_set train_nodev --valid_set train_dev --test_sets train_dev test
2024-09-24T23:03:26 (asr.sh:321:main) Info: The valid_set 'train_dev' is included in the test_sets. '--eval_valid_set true' is set and 'train_dev' is removed from the test_sets.
2024-09-24T23:03:26 (asr.sh:564:main) Skipped stages: 9 14 15
2024-09-24T23:03:26 (asr.sh:877:main) Stage 5: Generate token_list from dump/raw/org/train_nodev/text using BPE
Traceback (most recent call last):
  File "/content/drive/MyDrive/asr/espnet/tools/sentencepiece_commands/spm_train", line 9, in <module>
    import sentencepiece as spm
ModuleNotFoundError: No module named 'sentencepiece'
```

ESPnet example code:

https://colab.research.google.com/drive/1F5IXqzljBrJr3N_6UgW-EPTmfZIOGby?usp=drive link

Reason for Failure:

- Limited GPUS and frequent disconnects from Data
- Issues in installing dependency python packages
- **Blocker** : `!./asr.sh --stage 3 --stop_stage 3 --train_set train_nodev --valid_set train dev --test_sets "train dev test" --nj 4`

```

2024-09-24T23:01:54 (asr.sh:614:main) Stage 3: Format wav.scp: data/ -> dump/raw
utils/copy_data_dir.sh: copied data from data/train_nodenv to dump/raw/org/train_nodenv
utils/validate_data_dir.sh: Successfully validated data-directory dump/raw/org/train_nodenv
2024-09-24T23:01:55 (format_wav_scp.sh:46:main) scripts/audio/format_wav_scp.sh --nj 4 --cmd run.pl --audio-format flac --fs 16k
2024-09-24T23:01:55 (format_wav_scp.sh:118:main) [info]: without segments
run.pl: 4 / 4 failed, log is in dump/raw/org/train_nodenv/logs/format_wav_scp.*.log
# pyscripts/audio/format_wav_scp.py --fs 16k --audio-format flac --multi-columns-input false --multi-columns-output false dump/
# Started at Tue Sep 24 23:01:59 UTC 2024
#
Traceback (most recent call last):
  File "/content/drive/MyDrive/asr/espnet/egs2/an4/asr1/pyscripts/audio/format_wav_scp.py", line 8, in <module>
    import humanfriendly
ModuleNotFoundError: No module named 'humanfriendly'
# Accounting: time=1 threads=1
# Ended (code 1) at Tue Sep 24 23:02:00 UTC 2024, elapsed time 1 seconds
# pyscripts/audio/format_wav_scp.py --fs 16k --audio-format flac --multi-columns-input false --multi-columns-output false dump/
# Started at Tue Sep 24 23:01:59 UTC 2024
#
Traceback (most recent call last):
  File "/content/drive/MyDrive/asr/espnet/egs2/an4/asr1/pyscripts/audio/format_wav_scp.py", line 8, in <module>
    import humanfriendly
ModuleNotFoundError: No module named 'humanfriendly'
# Accounting: time=1 threads=1
# Ended (code 1) at Tue Sep 24 23:02:00 UTC 2024, elapsed time 1 seconds
# pyscripts/audio/format_wav_scp.py --fs 16k --audio-format flac --multi-columns-input false --multi-columns-output false dump/
# Started at Tue Sep 24 23:02:00 UTC 2024
#
Traceback (most recent call last):
  File "/content/drive/MyDrive/asr/espnet/egs2/an4/asr1/pyscripts/audio/format_wav_scp.py", line 8, in <module>
    import humanfriendly
ModuleNotFoundError: No module named 'humanfriendly'
# Accounting: time=1 threads=1
# Ended (code 1) at Tue Sep 24 23:02:01 UTC 2024, elapsed time 1 seconds
# pyscripts/audio/format_wav_scp.py --fs 16k --audio-format flac --multi-columns-input false --multi-columns-output false dump/
# Started at Tue Sep 24 23:02:00 UTC 2024
#

```

My result is shown below:

```
## asr_train_asr_demo_stochasticformer_raw_bpe30
### WER

|dataset|Snt|Wrd|Corr|Sub|Del|Ins|Err|S.Err|
|---|---|---|---|---|---|---|---|---|
|decode_asr_asr_model_valid.acc.ave/test|130|773|85.4|11.3|3.4|0.4|15.0|46.9|
|decode_asr_asr_model_valid.acc.ave/train_dev|100|591|77.3|15.7|6.9|0.7|23.4|62.0|

### CER

|dataset|Snt|Wrd|Corr|Sub|Del|Ins|Err|S.Err|
|---|---|---|---|---|---|---|---|---|
|decode_asr_asr_model_valid.acc.ave/test|130|2565|93.5|2.9|3.6|1.2|7.8|46.9|
|decode_asr_asr_model_valid.acc.ave/train_dev|100|1915|87.8|5.0|7.2|1.8|14.0|62.0|

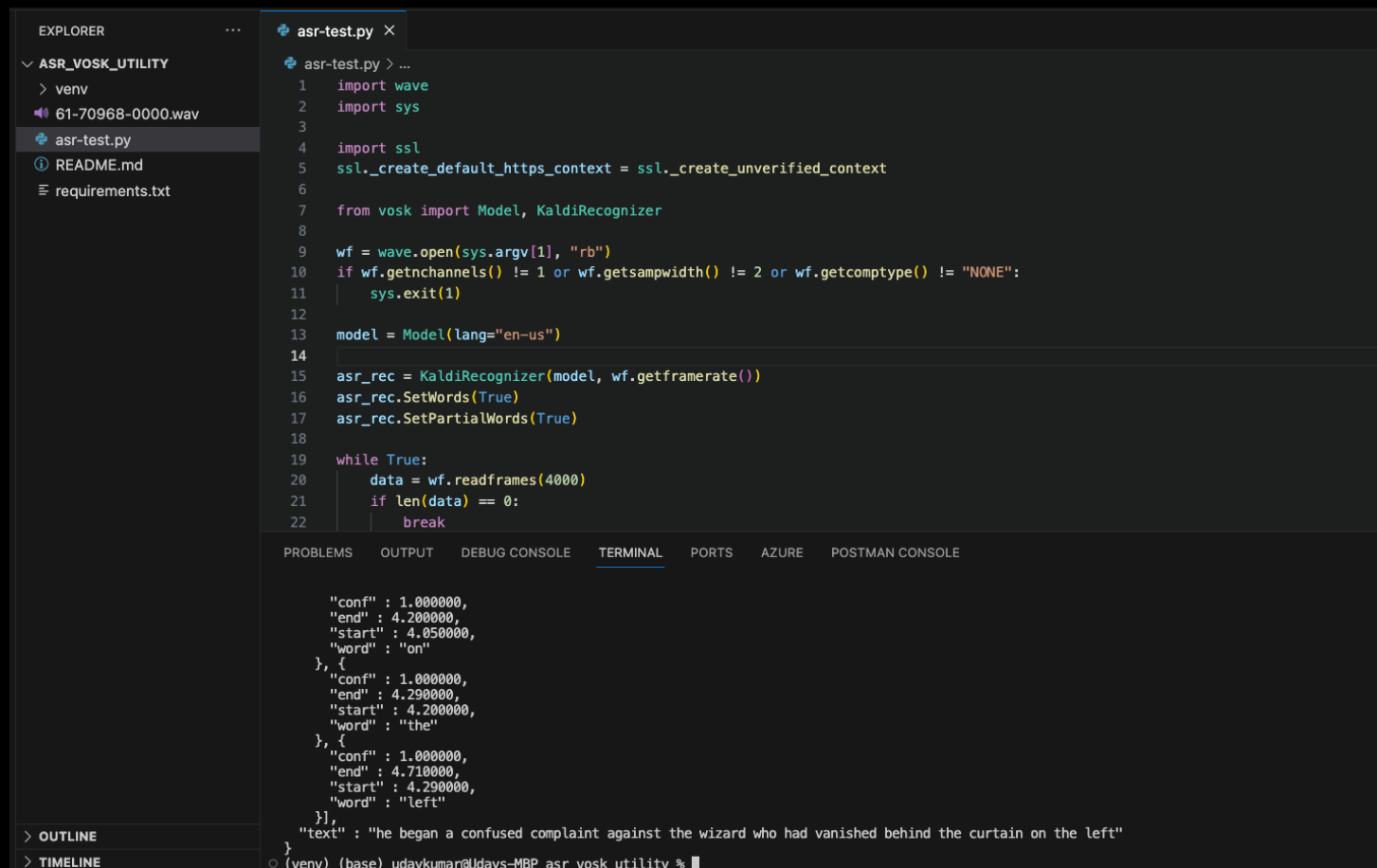
### TER

|dataset|Snt|Wrd|Corr|Sub|Del|Ins|Err|S.Err|
|---|---|---|---|---|---|---|---|---|
|decode_asr_asr_model_valid.acc.ave/test|130|2695|93.8|2.8|3.4|1.2|7.4|46.9|
|decode_asr_asr_model_valid.acc.ave/train_dev|100|2015|88.4|4.8|6.8|1.7|13.3|62.0|
```

VOSK API

11

- Vosk is a speech recognition toolkit. The best things in Vosk are:
 - Supports 20+ languages and dialects - English, Indian English, German, French, Spanish, Portuguese, Chinese, Russian, Turkish, Vietnamese, Italian, Dutch, Catalan, Arabic, Greek, Farsi, Filipino, Ukrainian, Kazakh, Swedish, Japanese, Esperanto, Hindi, Czech, Polish, Uzbek, Korean, Breton, Gujarati, Tajik.
- Works offline, even on lightweight devices - Raspberry Pi, Android, iOS
- Provides streaming API for the best user experience (unlike popular speech-recognition python packages)
- There are bindings for different programming languages, too - java/csharp/javascript etc.
- Allows quick reconfiguration of vocabulary for best accuracy.
- Supports speaker identification beside simple speech recognition.



```
EXPLORER
  ASR_VOSK_UTILITY
    > venv
    61-70968-0000.wav
    asr-test.py
    README.md
    requirements.txt

asr-test.py
1  import wave
2  import sys
3
4  import ssl
5  ssl._create_default_https_context = ssl._create_unverified_context
6
7  from vosk import Model, KaldiRecognizer
8
9  wf = wave.open(sys.argv[1], "rb")
10 if wf.getnchannels() != 1 or wf.getsampwidth() != 2 or wf.getcomptype() != "NONE":
11     sys.exit(1)
12
13 model = Model(lang="en-us")
14
15 asr_rec = KaldiRecognizer(model, wf.getframerate())
16 asr_rec.SetWords(True)
17 asr_rec.SetPartialWords(True)
18
19 while True:
20     data = wf.readframes(4000)
21     if len(data) == 0:
22         break

PROBLEMS OUTPUT DEBUG CONSOLE TERMINAL PORTS AZURE POSTMAN CONSOLE

{"conf": 1.000000,
 "end": 4.200000,
 "start": 4.050000,
 "word": "on"
 }, {
 "conf": 1.000000,
 "end": 4.290000,
 "start": 4.200000,
 "word": "the"
 }, {
 "conf": 1.000000,
 "end": 4.710000,
 "start": 4.290000,
 "word": "left"
 }
 }, {
 "text": "he began a confused complaint against the wizard who had vanished behind the curtain on the left"
 }
 }
 (venv) (base) udaykumar@Udays-MBP asr_vosk_utility %
```

Models used:

[RNNLM](#)

Source Code:

[asr_vosk_utility](#)