CLAP

WALKTHROUGH HIGHLIGHTING BACKEND AND FRONTEND INFRASTRUCTURE

Backend : USER SPECIFIC TABLES

speechtotextbackend_user : User id(key referenced elsewhere) , password,email , mode (demo/random) and other user specific detail

speechtotextbackend_appuser : references above , one to one correspondance – contains other user details like name, number of conversions, verifications ,scores

No of text verifications : updated when verify task allocated

No of conversions : updated when speak task allocated

speechtotextbackend_Loginlog – Timestramp record of logins

Backend : text (task) tables

speechtotextbackend_textfile : id , name, language, text , assigned , conversions, verified conversions, conversion passes

Stores generated text file **tasks -**details

Assigned : +1 when assigned to user/ -1 when user converts/passes the task

Conversions - +1 if user converts the text

Conversin_passes - +1 if user doesn't convert/ clicks on pass

Example data :

65294 | Regular29 | Tamil | மின்புத்தகங்கள் எவ்வாறு அமைய வேண்டும் | 1 | 1 | 0 | 0 | 2019-04-15 09:21:15.632059+05:30 | eruzhvali-short-stories-poems_5nshNVP | freetamilebooks | Regular

Speechtotextbackend_textfileappuser : References both textfile and user tables- i.e user-> task mapping details

This table adds data as soon as a user converts text to audio in the app, it contains user id , text file id, id of audio file recorded by user, verifications, audio length, verificaation passes (updated after another user verifies this task created),assigned (updated after this task is assigned for verification to another user)

Backend : text (task) tables

Speechtotextbackend_textfileappuserverify :

Contains id, text ,verificationStatus,
verificationDateTime,t extFileAppUser_id,
verifyingAppUser_id ,audio_quality

Textfileappuser_id references the task from
textfileappuser table and verifingappuser_id is the user
id . It creates the task according to the user verification
details. Audio quality refers to comment by user eg
"audio is fine"

Example data :

28490 | थोड़ा इंतज़ार करो फिर तुम्हें एक अनोखा फल
चखने को मिलेगा
| 3 | 2019-08-19 22:43:59.580215+05:30 |
2019-08-19 22:43:43.810465+05:30 | 100 |
        22270 | 445 | Inaudible;

Backend : Text (tasks) tables

Speechtotectbackend_uploadtextfile:

Two fields : id and textfile

Contains the list of text files uploaded

Others ⋮

No data recorded in any of these tables (for audio files conversion)

Speechtotextbackend_audiofile
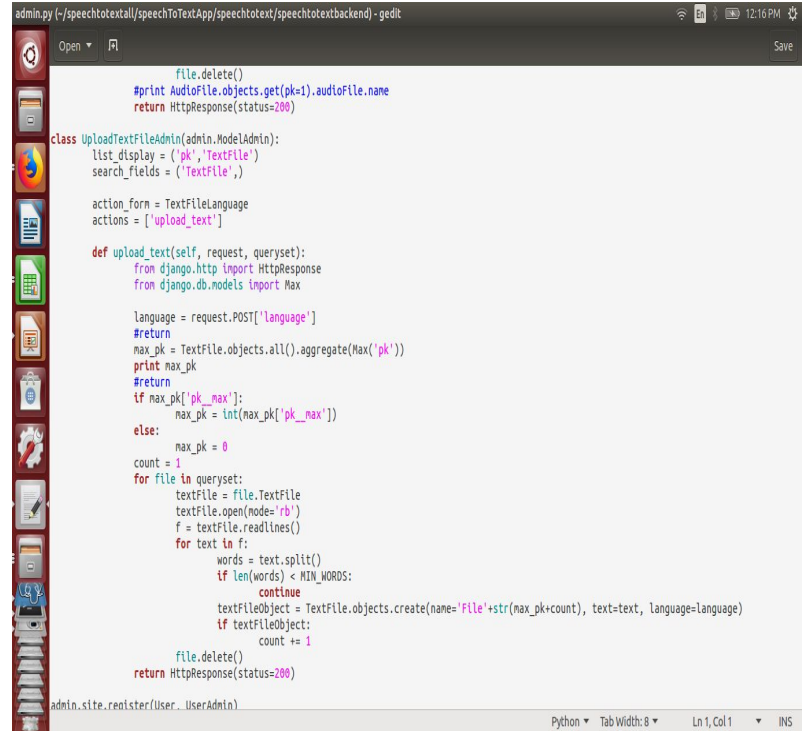
Speechtotextbackend_audiofileappuser

Speechtotextbackend_audiofileappuserverify

Speechtotextbackend_uploadaudiofile

TEXT CONVERSION WORKFLOW

-Text files exist under different sources (in different folders nameed Hindi, Marathi,Malyalam etc)

-Uploaded individually via dashboard (Admin interface)

- The code has been modified to filter sentences(remove special characters, filter by length -not included in this screenshot) but task creation is done in the final stage and uploaded to table speechtotextbackend_textfiles

TEXT CONVERSION WORKFLOW

TASK CREATION

- After upload , file gets deleted and tasks created in textfiles db, can be seen in the dashboard as well from textfiles
- Each task assigned a unique id while creation
- This id currently generates duplicates when a bulk of text files are uploaded and converted at the same time due to key issues

TASK ALLOCATION

MAX_TEXT_CONVERSIONS=2 -each text to speech task is given to two users atmost (constant can be changed)

MAX_TEXT_VERIFICATIONS = 3 each verfication task is given to two users currently, if there is a mismatch in comments then it is given to third user

Allocation of these tasks are assigned in batches per user filtered by selected language, task completed or not , not already assigned and some other variables.

Picks up the tasks such that max tasks assigned per user does not exceed the constants assigned

Currently max conversion tasks per user at a time is set to 5 and max verification task per user at a time is set to 3

MAX_ASSIGNED_TEXT_CONVERSION = 5

MAX_ASSIGNED_TEXT_VERIFICATION = 3

Algo in utils.py

def get_new_text_conversion_list(appUser, language):

def get_new_text_verification_list(appUser, language):

TASK ALLOCATION

TEXT TO SPEECH :

ASSIGNED BATCH SIZE = 10 Currently

The filtered files discussed previously are the valid files that can be assigned to users for conversion
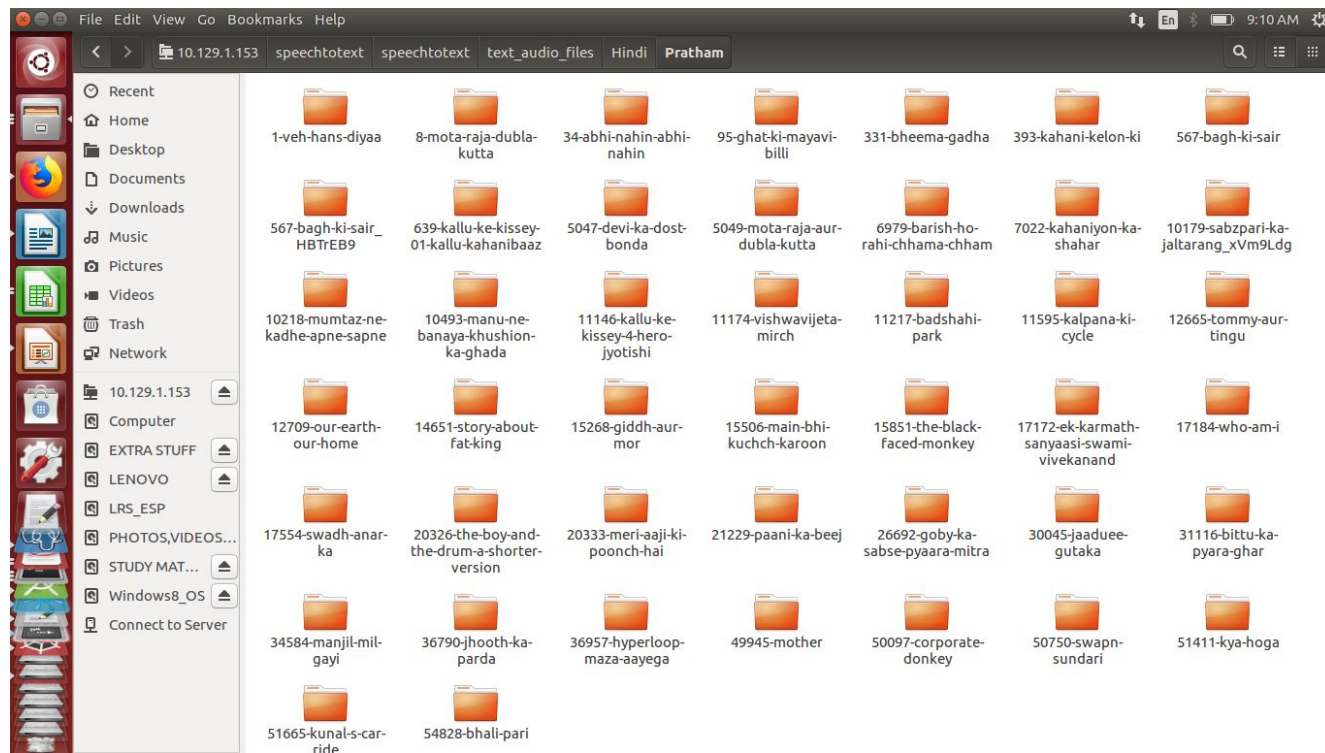
All the valid files are divided into different batches as per the Batch size allowed

All files in the batch should be assigned first and then only the next batch is assigned
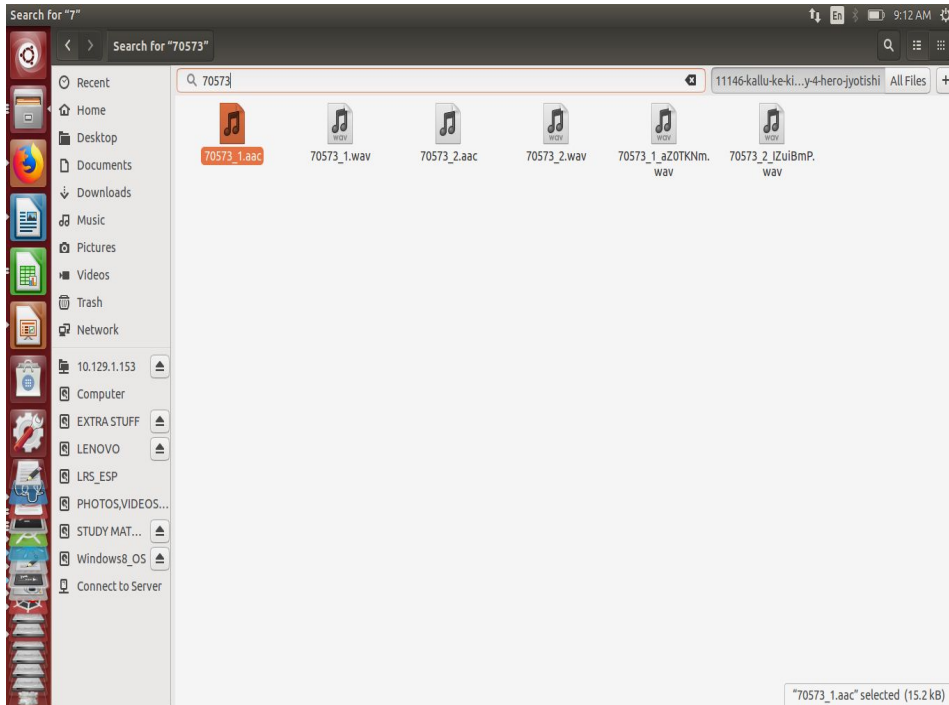
VERIFICATION :

Same algorithm followed

AUDIO FILE STORAGE

AUDIO FILE STORAGE

- 3 files created for each recording
- 1 orginal file, 1 faad generated aac converted file and 1 clean file
- 2 recordings for each task currently , so 6 files in total corresponding to each text file task
- Batch allocation followed so currently some tasks are in status complete (2 done) and in some tasks partially (1 done)
- Clean file picked up for validation

Overview of code structure

Utils.py -> contains all the algorithms for user task allocation and storage for both verify and speech tasks.

Admin.py-> code for uploading text files, files-> task conversion algorithm, csv changes

Models.py -> contains database structures, initialisations

constants.py-> constants like batchsizes, no of tasks assigned ,etc

Views.py -> code for dashboard modification