International Research Journal of Engineering and Technology (IRJET) e-ISSN: 2395-0056
Volume: 10 Issue: 05 | May 2023 www.irjet.net p-ISSN: 2395-0072

MULTILINGUAL SPEECH TO TEXT CONVERSION USING HUGGING FACE FOR DEAF PEOPLE

Dr.N.V Sailaja, Billakanti Sushma, Aredla Likitha Reddy, Charitha Parachuri, Chandra Akash

[1]*Assistant professor, Dept. Of Computer Science Engineering, VNRVJIET college, Telangana, India*
[2,3,4,5] *Student, Dept. Of Computer Science Engineering, VNRVJIET college, Telangana, India*

Abstract - *Internet has taken the place of traditional communication channels. The use of online communication has resulted in a rapid rise in audio and visual information. Although it has been beneficial for the majority of people, those with special needs, such as the deaf, have few resources at their disposal. A speech-to-text Conversion programme is written.The audio input is converted to text using speech recognition technology. Algorithms for Natural Language Processing are used to extract root words and segment words and translate to different languages.*

Our goal is to create a multilingual speech-to-text conversion system employing Hugging Face for hearing-impaired people. The technology will make it possible for deaf individuals to instantly translate spoken language into text, assisting them in a variety of tasks including listening to lectures, participating in meetings, or even having conversations with others. Hugging Face's[11] cutting-edge neural network models and natural language processing algorithms will improve the suggested system's precision and effectiveness. The system will also support many languages, enabling a wide range of people throughout the world able to utilise it. Overall, the suggested technology would enable seamless communication and engagement in many contexts, greatly enhancing the quality of life for those with hearing impairments.

Key Words*: *transformers, whisper, Machine Learning, Automatic Speech recognition, web page, tokenizer, pipeline.

1.INTRODUCTION

With the development of machine learning and deep learning algorithms, automated voice recognition has become a major study area. Using pre-trained language models, like Hugging Face, for fine-tuning is one such method.

Hugging Face, a well-known NLP library, offers pre-trained models for a variety of NLP tasks, including text categorization, question-answering, and language production [11]. Access to pre-trained Transformer architecture-based speech recognition models is also made available by the library.

In order to increase the model's precision and performance on a certain speech recognition task, fine-tuning Hugging Face models [11] for voice recognition entails training the pre-trained models on a particular dataset. When trained on domain-specific datasets, this method has been proven to greatly increase the accuracy of voice recognition systems.

In this study, our efforts with honing Hugging Face models for voice recognition were discussed. Outlined the training dataset, the fine-tuning procedure, and the assessment measures employed to gauge the model's effectiveness. Also included front end to record voice from microphone in order to test the model's efficiency and accuracy. The front-end enables easy use of the model and provides good user-experience.

Our findings show the value of fine-tuning pre-trained language models like Hugging Face[11] for speech recognition, and thought this strategy has a great deal of promise for raising the precision and efficiency of speech recognition systems across a variety of domains.

2.LITERATURE SURVEY

Applications that target rescue were the ones that were most prevalent when some comparable ones entered the research sector. They are:

In [1], The model proposed in this paper uses standard input speech to text Conversion engine to take the input speech. With the use of cutting-edge artificial intelligence methods like Deepnets and Q learning, the search space is reduced using the HMM (Hidden Markov) model, and the optimised results are then used in real time. The accuracy of the suggested phonetic model is 90%.

In [2], The Proposed approach deals with recognition of two different languages – kannada and English. A Deep learning voice recognition model is used in conjunction with a word prediction model. When evaluated on a multilingual dataset, the accuracy is 85%, and when the user tests it in real time, the accuracy is 71%. Cosine similarity model was employed. When predictions were made using the average similarity of each class, a 59% average accuracy for the cosine similarity model was attained.

In [3] This paper presents a complete speech-to-text conversion system for Bangla language using Deep Recurrent Neural Networks. Possible optimization such as Broken Language Format has been proposed which is based on

properties of the Bangla Language for reducing the training time of the network. It was trained with collected data and which yielded over 95% accuracy in case of training data and 50% accuracy in case of testing data.

In [4] The open-sourced Sphinx 4 framework, which is built in Java and offers the necessary procedural coding tools to construct an acoustic model for a unique language like Bengali, is required for the proposed model. To edit the recorded data, we utilised Audacity, a free digital audio workstation (DAW). Used an audio dataset of recorded speech from 10 different speakers, including both male and female speakers, and produced their own unique transcript files to test the performance. Experimental findings show that the suggested model had an average accuracy of 71.7% for the dataset they analysed.

In [5] It is an encoder-decoder model based on Transformers, commonly known as a sequence-to-sequence model. A series of audio spectrogram properties are translated into a series of text tokens using this approach. First, the feature extractor transforms the raw audio input into a log-Mel spectrogram. The spectrogram is then encoded using the Transformer encoder to create a series of encoder hidden states. Finally, the decoder predicts text tokens autoregressively based on both the hidden states of the encoder and the preceding tokens.

In [6] This model used Recurrent Neural Networks to convert Speech to text. This model includes prosodic modelling method. Prosodic modelling is done at the audio decoding process post-processing stage with the goal of identifying word-boundary cues to help with language decoding. To comprehend the input prosodic information and the output text-boundary, the RNN Model has three layers. After the training of RNN Model then it is used to generate the word -boundary cues to solve the problem of word-boundary ambiguity. Two Schemas were proposed. First Schema directly takes RNN outputs and directly adds to the word-sequence hypothesis. Schema 2 is extension of Schema 1 by adding Finite State Machines for setting path constraints. The Character accuracy were 73.6%,74.6% using the schemas 1 and 2.

3.METHODOLOGY

The suggested system uses a Transformer-based encoder-decoder as its foundation. The encoder receives a log-Mel spectrogram as input. Cross-attention techniques provide the decoder with the final concealed states of the encoder. Text tokens are conditionally predicted by the decoder autoregressively based on hidden states from the encoder and previously predicted tokens.

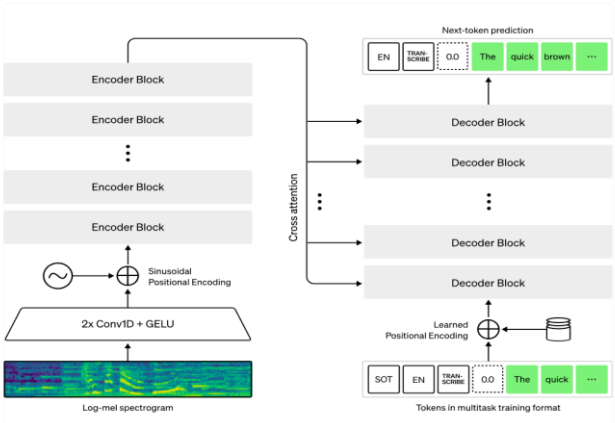

Fig. 1. Architecture diagram

3.1 Data Collection

A number of crowd-sourced dataset called Common Voice in which people record Wikipedia text in a variety of languages. Version 11 of the Common Voice dataset is used in this instance. As for our chosen language, Hindi, an Indian language used in the northern, central, eastern, and western parts of India, serve as the basis for further developing of our model.

3.2 Feature Extraction and Tokenizing

Speech is represented as a temporally varying 1-dimensional array. The signal's amplitude at every particular time step is represented by the value of the array at that instant. Speech includes an endless variety of amplitude values since it is continuous. Computer hardware that relies on finite arrays will have issues. As a result, discretized our voice signal by taking samples of its values at predetermined time intervals. The sampling rate, which is often expressed in samples per second or Hertz (Hz), is the interval with which sounds are sampled.

Although storing more information per second is necessary, sampling with a greater sample rate produces a better approximation of the continuous speech input.

The feature extractor carries out two tasks. A series of audio samples are originally padded or truncated so that each one has a 30-second input time. By adding zeros to the end of the sequence, samples that are less than 30 seconds are stretched to 30 seconds (zeroes in an audio transmission signify a silent or no sound). No need to use attention mask as every batch is padded or truncated to maximum length in input. Tokenizer has been pre-trained using the transcriptions for 96 pre-training languages [11]. Load the pre-trained check point and train it on the data set and fine tune it.

3.3 Training and Evaluation

Define a data collator: This programme takes the data that has already been processed and creates PyTorch tensors that are required for the model.

Evaluation Metrics: Used word error rate (WER) measure to assess the model.

Load pre-trained checkpoint: Load the checkpoint that has already been trained and properly set it up for training.

Define the training arguments: The Hugging Trainer[11] will use them to build the training schedule.

3.4 Web Application

For the project's purposes, a web application is created that provides complete control over the fine-tuned model. The web application serves as a user interface for the model and is written using Gradio.

An interface page with controller choices makes up the application. On the page, options like recording, clearing, and submitting are provided. The end-user can manually enter voice from microphone with the help of these control choices.

The backend support of the application is fine-tuned model. The View portion is a straightforward front-end component with only one page. The computer and model must be linked to a Wi-Fi network in order for the web application to work. No extra setup is required. First clicking on record button enables microphone and records the voice that is spoken. Stop button is provided in order to stop the recording. After recording, submit and clear buttons are provided. When submit is clicked the model transcribes it and displays the text result in the text box provided. You can use Flag button to store that recording.

3.5 Results

The procedure goes as follows. The machine learning model is directly connected to the microphone. When an audio is recorded through the microphone the machine learning model processes the audio and displays the text corresponding to the recorded audio on the screen. The recognized speech is then segmented into 30 seconds and if it is less than that then 0's appended at the end. These segments then are converted into Log-Mel Spectrum inside the model and through several layers it divides speech into tokens and prints the resulted text. While transcribing the added 0's will be deleted. The WER of the model is 32.42% which is very less.

Fig. 2 Testing the model

Fig. 3. Application demo

3. CONCLUSIONS

Our work helps the deaf people in understanding others speech by converting it to text by using hugging face[11]. The motivation for this kind of study and experiments is the fact that whisper [12] is a simple ASR system for its perfect understandability in human face-to-face communication. Speech Recognition for Hindi Language and Conversion into Hindi Text is done. The implemented model is trained on a dataset of 3500 audios and achieved a word error rate of 32.42%. It is shown that whisper signals, can give high scores in word recognition for speech. For Open AI whisper small the WER is 87.30%[12] and now our work fine-tuned the model and reduced the Word Error rate to 32.42%.

REFERENCES

[1] Gulbakshee Dharmale, Dipti D. Patil and V. M. Thakare, "Implementation of Efficient Speech Recognition System on Mobile Device for Hindi and English Language" International Journal of Advanced Computer Science and Applications(IJACSA), 10(2), 2019.

[2] Multilingual Speech to Text using Deep Learning based on MFCC Features P Deepak Reddy, Chirag Rudresh and Adithya A S, PES University, India.

[3] M. T. Tausif, S. Chowdhury, M. S. Hawlader, M. Hasanuzzaman and H. Heickal, "Deep Learning Based Bangla Speech-to-Text Conversion," *2018 5th International Conference on Computational Science/ Intelligence and Applied Informatics (CSII)*, Yonago, Japan, 2018, pp. 49-54, doi: 10.1109/CSII.2018.00016.

[4] A. U. Nasib, H. Kabir, R. Ahmed and J. Uddin, "A Real Time Speech to Text Conversion Technique for Bengali Language," *2018 International Conference on Computer, Communication, Chemical, Material and Electronic Engineering (IC4ME2)*, Rajshahi, Bangladesh, 2018, pp. 1-4, doi: 10.1109/IC4ME2.2018.8465680.

[5] https://arxiv.org/abs/2212.04356

[6] Wern-Jun Wang, Yuan-Fu Liao, Sin-Horng Chen,

RNN-based prosodic modeling for mandarin speech and its application to speech-to-text conversion,Speech Communication,Volume 36, Issues 3–4,2002,Pages 247-265,ISSN 0167-6393,https://doi.org/10.1016/S0167-6393(01)00006-1.

[7] Software Engineering - A Practitioner's Approach, Roger S. Pressman, McGraw-Hill

[8] Machine Learning, Tom M. Mitchell, McGraw-Hill International Edition, 6th Edition, 2001

[9] https://huggingface.co/datasets/mozilla-foundation/common_voice_11_0

[10] https://huggingface.co/blog/fine-tune-whisper

[11] https://huggingface.co/docs

[12] https://openai.com/research/whisper