Speak2Summarise: Daily Task Recap

**GROUP NO 28**

***This project report is submitted to Yeshwantrao Chavan College of Engineering***

***(An Autonomous Institution Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University)***

***In partial fulfilment of the requirement for the award of the degree***

**SPEAK2SUMMARISE : DAILY TASK RECAP**

***Of***

***B. Tech in Computer Science and Engineering***

***By***

**Hemanshu Waghmare  
Dhruv Dalvi  
Rishabh Jain  
Sanket Asole  
Yuvraj Chavan**

**PROF. CHANCHLA TRIPATHI**

***Under the guidance of***

**Prof. Chanchla Tripathi**

****

**Department Of Computer Science and Engineering  
Nagar Yuwak Shikshan Sanstha   
YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING,**

**(An autonomous institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur)**

**2024-25**

**NAGPUR – 441 110**

**2024-25**

**Speak2Summarise: Daily Task Recap**

***This project report is submitted to Yeshwantrao Chavan College of Engineering***

***(An Autonomous Institution Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University)***

***In partial fulfilment of the requirement for the award of the degree***

***Of***

***B. Tech in Computer Science and Engineering***

***By***

**Hemanshu Waghmare  
Dhruv Dalvi  
Rishabh Jain  
Sanket Asole  
Yuvraj Chavan**

***Under the guidance of***

**Prof. Chanchla Tripathi**

****

**Department Of Computer Science and Engineering  
Nagar Yuwak Shikshan Sanstha   
YESHWANTRAO CHAVAN COLLEGE OF ENGINEERING,**

**(An autonomous institution affiliated to Rashtrasant Tukadoji Maharaj Nagpur University, Nagpur)**

**NAGPUR – 441 110**

**2024-25**

**CERTIFICATION of APPROVAL**

Certified that the project report entitled “**Speak2Summarise: Daily Task Recap**” has been successfully completed by the Mr. Hemanshu Waghmare, Dhruv Dalvi, Rishabh Jain, Sanket Asole and Yuvraj Chavan under the guidance of Prof. Chanchla Tripathi in recognition to the partial fulfilment for the award of the degree of Bachelor of Technology in Computer Science and Engineering, **Yeshwantrao Chavan College of Engineering (An Autonomous Institution Affiliated to Rashtrasant Tukdoji Maharaj Nagpur University)**

|  |  |  |  |
| --- | --- | --- | --- |
| Signature | Signature | Signature | Signature |
| Prof. Chanchla Tripathi | Prof. Priyanka Khandekar | Prof. Amruta T. Paul | Dr. Lalit B. Damahe |
| (Guide) | Project Coordinator | Project Coordinator | (HoD) |

Signature of External Examiner Name:

Date of Examination:

**ACKNOWLEDGEMENT**

We extend our heartfelt gratitude to **Prof. Chanchla Tripathi** for their unwavering support, invaluable guidance, and encouragement throughout this project on **Speak2Summarise: Daily Task Recap.** Their expertise and insights have been instrumental in shaping the direction and execution of this research. We are also grateful to **Dr. Lalit B. Damahe,** Head of the CSE Department, for providing essential resources and facilities crucial for the successful completion of this project. Additionally, we appreciate **Dr. U. P. Waghe, Principal of YCCE**, for fostering an environment conducive to academic exploration and innovation. Special thanks are due to **Prof. Amruta Tapas Paul** and **Prof. Priyanka Khandekar,** Coordinator of CSE Department, for overseeing the coordination and administrative aspects of this project. Lastly, we would like to thank all the faculty and non-teaching staff members who helped us directly or indirectly for the successful completion of this project.

**DECLARATION**

We certify that

1. The work contained in this project has been done by us under the guidance of my supervisor(s).
2. The work has not been submitted to any other Institute for any degree or diploma.
3. We have followed the guidelines provided by the Institute in preparing the project report.
4. We have conformed to the norms and guidelines given in the Ethical Code of Conduct of the Institute.
5. Whenever we have used materials (data, theoretical analysis, figures, and text) from other sources, we have given due credit to them by citing them in the text of the report and giving their details in the references. Further, we have taken permission from the copyright owners of the sources, whenever necessary.

**Name of the Students Signature of the Student**

Hemanshu Waghmare

Dhruv Dalvi

Rishabh Jain

Sanket Asole

Yuvraj Chavan

**TABLE OF CONTENTS**

Title Page

Certificate of Approval i

[Acknowledgement ii](#_gjdgxs)

[Declaration iii](#_30j0zll)

Table of Contents iv-v

[List of Figures](#_17dp8vu)  vi

[Abstract](#_4d34og8) ix

[Project CO-PO matrix](#_3znysh7)  x

[Abstract 5](#_Toc198319402)

[CHAPTER 1. INTRODUCTION 9](#_Toc198319403)

[1.1 Overview 9](#_Toc198319404)

[1.2 Problem Statement 10](#_Toc198319405)

[CHAPTER 2. REVIEW OF LITERATURE 11](#_Toc198319406)

[2.1 Review on Previous years research and work 11](#_Toc198319407)

[2.2 Patent Search 17](#_Toc198319408)

[CHAPTER 3 WORK DONE 18](#_Toc198319409)

[3.1 WORKFLOW 18](#_Toc198319410)

[3.2 METHODOLOGY 20](#_Toc198319411)

[3.3 Methods 22](#_Toc198319412)

[3.3.1 Cloud based 22](#_Toc198319413)

[3.3.2 Neural Network 22](#_Toc198319414)

[Long Short-Term Memory (LSTMs) 22](#_Toc198319415)

[Recurrent Neural Networks (RNN) 23](#_Toc198319416)

[3.3.3 Extractive Summarization 24](#_Toc198319417)

[Transformer based 24](#_Toc198319418)

[Generative Adversarial Networks (GANs) 24](#_Toc198319419)

[Two-Stage Transformer 25](#_Toc198319420)

[Dual-encoding using transformer 25](#_Toc198319421)

[PodSumm Architecture 26](#_Toc198319422)

[3.4 DESCRIPTION OF DATA SET TO BE USED 9](#_Toc198319423)

[3.5 PLAN OF PROJECT 10](#_Toc198319424)

[3.5.1 Audio Preprocessing 11](#_Toc198319425)

[3.5.2 Speech Recognition and Speaker Diarization 11](#_Toc198319426)

[3.6 ARCHITECTURE 11](#_Toc198319427)

[3.6.1 WhisperX ( PROJECT MODEL ARCHITECTURE ) 11](#_Toc198319428)

[3.6.2 Wav2vec 2.0 13](#_Toc198319429)

[CHAPTER 4. RESULTS AND DISCUSSION 25](#_Toc198319430)

[4.1 High Transcription Accuracy 25](#_Toc198319431)

[4.2 Noise Reduction Using pydub 26](#_Toc198319432)

[4.3 Effective Speaker Diarization 27](#_Toc198319433)

[4.4 Identifying Speaker Time Periods 28](#_Toc198319434)

[4.5 Transcription Output 29](#_Toc198319435)

[4.6 Accuracy Measurement (WER and CER) 29](#_Toc198319436)

[4.7 Social Utility of the Project 30](#_Toc198319437)

[4.7.1 Task Management and Productivity 30](#_Toc198319438)

[4.7.2 Educational and Accessibility Applications 30](#_Toc198319439)

[4.7.3 Data-Driven Decision-Making 31](#_Toc198319440)

[4.7.4 Community Engagement and Development 31](#_Toc198319441)

[CHAPTER 5. SUMMARY AND CONCLUSION 36](#_Toc198319442)

[*References* 39](#_Toc198319443)

**LIST OF FIGURES**

|  |  |  |
| --- | --- | --- |
| **Image No.** | **Image Name** | **Page No.** |
| 3.1 | Dataset CSV file | 20 |
| 3.2 | Common Voices audio data | 21 |
| 3.3 | Whisper architecture | 29 |
| 3.4 | Phrase-level timestamps, multilingual speech transcription, and to-English speech translation | 29 |
| 3.5 | Wav2vec 2.0 architecture and its pre-training process | 31 |
| 3.6 | Wav2vec 2.0 latent Feature Encoder | 32 |
| 4.1 | Noise Reduction | 34 |
| 4.2 | Speaker Diarization | 35 |
| 4.3 | Time period for each speaker | 36 |
| 4.4 | Transcription | 36 |
| 4.5 | WER and CER | 37 |

# 

# 

# Abstract

*By virtue of the burgeoning nature of digital audio data, Speech-to-Text (STT) conversive and text summarization are stated to be the vital foresightedness within Natural Language Processing (NLP). These developments make simultaneous speech-to-text conversion and text condensation of large volumes of textual information possible. It is applied in areas such as meeting transcription, virtual assistants, podcasts review and auto-generation of content where its importance in both research studies & business operations cannot be overemphasized.*

*Here the result and activity holders of the first two of the five phases in all, for creating the united framework of the STT and summarization are presented. The first phase focuses on audio processing which includes improved noise attenuation measures like use of audio spectral gating and also use of long audio files in segments for processing. The second phase aims at investigating the potential of speaker diarization using the potent WhisperX model with regards to seeking improvements in transcribing as well as effectively diarizing multiple speakers during a single event.*

*This work’s major contributions are: focusing on certain aspects in preprocessing and transcription procedures, evaluating the relevance of the applied methods and analysing the discussed topics. It can be see that there exist significant coupling between these phases to facilitate integration of transformer based text summarization models for subsequent processes. In addition to demonstrating the current enhancements of this paper, this paper also serves as a foundation towards achieving a more unified pipeline for the processing, analyzing, and summarizing the passed audio data in different applications.*

**Project CO-PO Matrix**

**CO-PO/PSO Articulation Matrix**

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | PO  1 | PO  2 | PO  3 | PO  4 | PO  5 | PO  6 | PO  7 | PO  8 | PO  9 | PO  10 | PO  11 | PO  12 | PSO  1 | PSO  2 |
| CO-1:  Investigate the Problem domain knowledge in their area of study and analyse the problem domain for executing the  projects. | **3** | **3** |  | **3** |  |  |  |  |  |  |  |  | **3** | **3** |
| CO-2: Design and develop the solution using appropriate tools and techniques and communicate effectively with society at large for  betterment. |  |  | **3** |  | **3** | **3** | **2** |  |  | **3** |  |  | **3** | **3** |
| CO-3: Analyse and interpret the obtained results using acquired research-based  knowledge. |  |  |  | **3** |  |  |  |  |  |  |  | **3** | **3** | **3** |
| CO-4: Function effectively as an individual, and as a member or leader in a team under multidisciplinary settings following  ethical practices. |  |  |  |  |  |  |  |  | **3** |  | **2** | **3** | **3** | **3** |

# 

# CHAPTER 1. INTRODUCTION

## Overview

Consequently, in the technological-dominated large universe that is characterized by automation and analytical decision-making, audio is a critical means of information. Whether it is a business meeting, online class, podcast or an interview, audio data is a goldmine that needs to be captured and analysed correctly. This is where **Speech-to-Text (STT)** as well as the text summarization technologies have significantly extremely useful part.

There are two types of functionalities, namely producing easily searchable textual or other output modality from raw speeches, in Speech-to-Text systems. This empowers users to turn what is often ephemeral; audio data into more tangible forms that are actionable, effective and practical influencing decision-making, work flow and providing accessibility to beneficial information.

Thus, text summarization enhances the functioning of STT by transforming a huge amount of text data produced by transcription services into brief summaries. The purpose of these summaries helps the users from getting overwhelmed with more information and deal with the overload to save much time. This is especially helpful when compared to transcribing several hours of speeches, podcasts, or even recordings, so long that it is impossible to listen to the entire thing in one sitting, and entire meetings, conferences, or seminars where key points are given.

Altogether these technologies are useful to facilitate the process improvements and support knowledge management activities. Workers can convert notes taken during meetings and convert them into summaries or notes of action points important for follow up. In education, lectures recorded in writing can be advisable reduced, in order that the most important aspects of the lecture can be emphasized. In media and entertainment, these technologies assist in the time-consuming interviews, podcasts and recordings by easing the process of summarizing them for publications and archiving purposes. Besides, in medical and legal professions, they help experts record conversations or meetings, undertake case, deposition, or patient consultation notes. In customer support, they use text analysis to log and summarise call centre conversations, in order to gain insights into common issues and customers’ sentiments.

These technologies are gradually being applied in diverse processes, leading to better resources utilization and retaining knowledge and information gathering efficiency. organizations continue to look for ways to harness the increase in the amount of audio data available, the combination of STT and text summarization is fast becoming one of the foundations for the modern processing and analysis of data.

## Problem Statement

The hectic way we live today causes people to lose focus which results in reduced productivity combined with missed assignments and increased tension. Multiple work and personal duties collect in our lives which overloads our ability to maintain organization along with focus. Actions that lead to task.errors or activity omissions extend their negative impact throughout our schedules and those around us which creates a never-ending cycle of stress and reduced efficiency.

The workplace consequences of multitasking include lower productivity along with economic loss but at the personal level excessive multitasking creates unfilled duties that heighten mental tension. Task management alongside memory aids helps users stay better organized while reducing decision fatigue and improving their focus abilities. Operation platforms pairing task management systems with calendars plus AI-based digital helpers prioritize work and guarantee timely finishing of assignments. Smart workflows emerge from using techniques including Eisenhower Matrix along with Pomodoro Technique and time management approaches. Note-taking with mnemonic techniques and activity stretching methods enhance performance throughout each day.

Voice prompts and wearable schedulers function through advanced technology to deliver real-time task communication. Tools enhance both organization and productivity while reducing stress so individuals maintain focus on meaningful creative work. A personalized system selection process converts disorganized daily struggle into orderly productive sequences while freeing both time and enhancing life quality.

# CHAPTER 2. REVIEW OF LITERATURE

# 

## 2.1 Review on Previous years research and work

The researchers in "A Review on Sentiment Analysis on Text, Image, and Audio Data" demonstrate how multimodal frameworks brought about improvements in sentiment analysis by moving beyond text-only approaches into systems that process audio and visual data [1]. According to Mehta et al. (2021) new hybrid models derived from lexicon-based and machine learning approaches bring enhanced accuracy to sentiment analysis techniques. Traditional VADER dictionaries founded sentiment classification through pre-defined lexica yet prove inadequate for detecting new or specialized language elements. Complex data gets decoded by machine learning models which help them find detailed insights but demand major compute power to run properly. Hybrid sentiment analysis systems combine the easy execution of lexicon-based approaches for smaller datasets with machine learning adaptable capabilities on bigger datasets for achieving better accuracy results.

Mehta et al. (2021) identify the necessity of systems which integrate audio, text, and visual data to better handle the shortcomings of working with individual data sources. Audio format strength increases when transcription text combines with acoustic elements including pitch and intensity to detect sentiment in audio sources. The analysis of audio according to text data delivers complementary information that strengthens the accuracy of sentiment evaluation. Researchers support ongoing investigation of advanced multimodal analysis methods in order to process complex datasets and boost sentiment analysis success rates. The expanding importance of hybrid approaches and multimodal systems with audio data plays a key role according to Mehta et al. (2021) for improving sentiment detection systems.

The work of Reddy et al. (2024) conducts audio analysis through text characterization and summarization while leveraging high-tech NLP tools including Facebook's BART model for summaries and Google's Speech-to-Text API for transcription [2]. The tools convert unstructured audio data into accessible text format for multiple uses that include video and podcast transcription and automated meeting summaries along with content searching functions. This research explores important barriers including noise reduction alongside context protection for automated audio summary solutions. Planned NLP models that combine LSTM systems with transformer-based hybrids BERT and BART enable superior context interpretation and precise audio-to-text conversions. Research by Reddy et al. (2024) supports merging contemporary NLP technologies with audio analytics to elevate operational efficiency while enhancing value-driven analysis. The systems prove their usefulness in various domains including business and education because effective audio data summarization has become essential to these sectors. The researchers at Graves et al. (2013) proposed Recurrent Neural Networks (RNNs) to operate on speech recognition applications because the networks excel at processing sequential data. The integration of RNNs with Long Short-Term Memory (LSTM) units enhances sequence labeling performance to reshape tasks involving cursive handwriting recognition [3].

Initial speech recognition tasks proved difficult for RNNs even though these models demonstrated better performance than deep feedforward networks for context-sensitive recognition. The research team of Graves et al. (2013) invented deep recurrent neural networks that combine diverse representation models with extended temporal context through end-to-end procedures enabled by Connectionist Temporal Classification (CTC) for unaligned pair data. The deep LSTM RNNs gained state-of-art recognition performance on the TIMIT phoneme benchmark by achieving 17.7% test set accuracy. Graves et al. (2013) highlight the effectiveness of uniting deep architecture structures with LSTMs but propose using these systems for big vocabulary applications and developing frequency-domain convolutional networks by integrating them with LSTMs.

The research by Ghadekar et al. (2023) demonstrates how Python libraries such as NLTK and SpaCy transform vocal inputs into text then create summaries. Tokenization along with stemming and lemmatization features exist in NLTK yet SpaCy excels at handling English language data. The application spectrum of this technology spans podcast summaries along with meeting transcriptions and indexing platforms and search functionalities. The usability of a friendly Graphical User Interface allows for better handling of extended audio files leading to improved efficiency in audio-to-text summarization [4]. Su, Wu and Cheng (2020) designed a Transformer-based model that operates in two stages for text summarization which produces flexible summaries adjusting to multiple user requirements [5]. The system comprises two primary components: Two major sections compose the system: a text segmentation module followed by a two-stage summarization module. The primary function of this module employs a BERT-based BiLSTM to segment extensive written material into meaningful pieces. Improving sentence relationship detection remains essential to extracting key information because this segmentation method enhances the summary model's understanding.

The extractive summarization module built with BERTSUM selects important sentences from separate segments while performing redundancy minimization during its first operational stage. These extracted sentences are then passed to the second stage: the document summarization module. During this stage all extracted content receives additional filtering while parameter adjustments come from evaluating scores from both modules to direct training operations collectively. Combined modules deliver enhanced alignment of summary units at different levels to maintain both structure and cohesion within the final document summary. The system allows users to customize summary length through an output concatenation process from the segment summarization module. Human evaluations confirm that the BERT-biLSTM segmentation module succeeds in capturing sentence relationships when tested on the ChWiki\_181k dataset by achieving 70.0% score in human evaluations on the LCSTS dataset.

On the other hand, the model also has limitations that stem from the Chinese corpus, which is more about one-line summaries. The system cannot generate detailed, multi-sentence summaries, which result in poor information-rich content. Furthermore, segmenting the less-constrained article could call for less-than-optimal segmenting with an overall negative impact on summary quality. This could assist the model's performance and applicability by solving these issues: including multi-sentence summaries in the corpus and segmenting optimization.

Vaswani et al. (2017) present the Transformer, an epoch-making sequence transduction model that takes an entire paradigm shift away from recurrent and convolutional neural networks to become wholly reliant on the attending mechanism. The salient drawback of this model compared to the previous ones is that it is not computationally inefficient and yet amenable to parallelism. It further employs multi-headed self-attention and feed-forward layers to simplify the encoder-decoder framework in such a way that distinguishes relationships among sequences without the restrictions imposed by recurrence or convolution[6].

It has performed exceptionally well on various machine translation tasks and has set several new records in the field. A "big" Transformer supervised the WMT 2014 English-to-German translation task, obtaining a BLEU score of 28.4, surpassing all previous results by more than 2.0 BLEU points, including ensemble models. Likewise, on the WMT 2014 English-to-French translation task, the score was 41.8, indicating an enormous margin over the best single model results. Most outstandingly, this was done with a significantly improved training efficiency, taking only 3.5 days on eight P100 GPUs, a drastic reduction in computation costs compared to earlier approaches.

The model's competence in translation alone shows how the Transformer has been adapted extensively to other sequence-related problems where longer outputs are often involved, such as constituency parsing of English. Therein, it scored an F1 of 91.3 when trained only on Wall Street Journal data, improved to 92.7 in a semi-supervised setting, aided by further additional corpus. Thus, the reviewed results underline the ability of the Transformer to generalize across different tasks by varying a few task-specific parameters. With inventive architecture and great computational efficiency, it has set forth a foundation model for natural language processing, resulting in new developments in summarization, question-answering, etc.

The audio transcription and summarization system introduced by Khonde, Shah, and Patel, in the 2023 issue of the International Journal on Recent and Innovative Trends in Computing and Communication, provides a radical step forward in meeting transcription and summarization. The proposed system combines cloud computing and Artificial Intelligence to provide a new way of documenting the meeting in the form of concise and personalized summaries including agendas, important takeaways, and action items. In turn, this would enhance communication and efficiency in organizations by providing users with a time-saving and productivity tool[7].

This solution is exceptionally scalable to fit organizations of any size, flexible in regard to being customized according to organizational needs, and most importantly-safe. This product is still under development, however, if developed it is anticipated to bring key changes in how companies, from small startups to big enterprises, deal with their meeting documentation. It is expected that the benefits include improving meeting productivity through clear and straight-to-the-point summaries, allowing the users to focus on action points, and bettering communication through a common point of reference for matters discussed at the meeting.

The paper also touches upon a complementary topic, "Text Summarization Using NLP" by Dhumal, Priyanka, Sutar, Sudarshan, Surve, Indraneel, Munawwar, Mirza, and Nanaware, Vishal. This paper outlines the problems predicted by information overload in today’s digital society and the necessity for efficient methods in the extraction and presentation of core content. The authors propose an NLP-based approach concerning extractive text summarization with the underlying intent of increasing retrieval of content by weeding out trivial information yet retaining the major sense of the original text[8].

The methodology follows an extractive path with four stages: pre-treatment for cleaning the input text, sentence scoring for judging importance, sentence selection for extracting key content, and post-processing to ensure coherent summaries. All these together provide brief summaries that capture the essence of the original material. The method here expresses the applicability of NLP in solving the present-day issues of content overload and efficient information management.

In their article Towards End-to-End Speech-to-Text Summarization, Monteiro and Pernes concentrate on the challenge of large-scale audio-visual content, primarily broadcast news. They describe the limitations of conventional speech-to-text (S2T) systems, which adopt extractive summarization techniques that select key sentences from a transcript but often yield summaries that lack coherence and readibility. In their view, the decline of coherence andReadability in these summaries must be compensated by more absorption of abstractive summarization systems, which are able to present adequate and fluent human-like summaries for audio-visual content. This current research proves to be an even more pressing demand with the rapidly evolving deep learning and large language models, marking a pivotal milestone in text generation.

The paper critique the so-called traditional cascade models that separate speech recognition from the summarization process, claiming these won't gather the necessary nuance and latent representations for effective summarization. To overcome these limitations, the authors thus suggest that an innovative end-to-end (E2E) model integrates together the speech-to-text transcription and the abstractive summarization all into one framework. One of the key contributions of the E2E model is the cross-modal attention mechanism, which tightens the link between speech features and corresponding textual representations. Thus, allowing the model to tap into so much-more-rich contextual information, which boosts the quality of the produced summaries.

The E2E model behaves under an encoder that comprises a 1-layer BiLSTM to process speech features and a 1-layer LSTM decoder for summary generation, whereas to tackle the low frequency of speech features, the authors apply a 2-layer convolutional network prior to the encoder for reducing the sequence length. This way, the model can stress the most pertinent parts of the input without overloading itself with processing irrelevant information.

In their experiments, the authors relied on a French broadcast news corpus to test the two different prototype systems: the cascade system and their novel E2E model. The cascade system exploits pre-trained Text-to-Text (T2T) summarizers and large external corpora such as MLSUM to perform speech recognition-independent abstraction. On its part, the E2E model is trained on the BNews corpus, favoring a much more integrated view of speech-to-text summarization at the cost of richness in context representations [9].

The paper-"Audio Summarization for Podcasts" of Vartakavi, A., Garg, A., and Rafii, Z. describes certain challenges faced in audio content discovery and recommendation systems in context to podcasts. These challenges abound due to distinctive characteristics of podcasts, including different voices of speakers, overlapping speech, background noise, and audio effects. Some traditional systems for summarization were built to handle structured text like news articles on a large scale and would thus hardly perform well in the nuanced and insipidly structured environments of podcasts. So, to overcome this, the authors propose the PodSumm-based system that generates audio summaries for the podcasts based predominantly on Automatic Speech Recognition (ASR) and extractive text summarization techniques [10].

In essence, PodSumm transcribes the audio of the podcast using ASR, while an extractive summarization mechanism reduces the transcript to smaller portions. Unlike the text-to-text summarization methods that only provide a summary, PodSumm provides audio summary generation by stitching selected segments of the podcast transcript. By preserving significant cues from the speaker's voice, production quality, and audio style, the mood and intention behind the original content are kept intact. The audio summaries thus provide a sneak preview, like a movie trailer, of the full episode to aid the listener in deciding whether to listen to it.

Lack of freely available data has caused the authors to prepare their own dataset containing manually annotated samples of selected podcasts and generated summaries. The authors accordingly tuned the Transformer-based model PreSumm, which is basically a news summarization model, for the peculiarities of podcasts. The authors achieved a promising performance, obtaining ROUGE scores of ROUGE-1: 0.63, ROUGE-2: 0.53, and ROUGE-L: 0.63, indicating that the model was capable of extracting important aspects from podcast episodes and providing summaries of them.

## 2.2 Patent Search

**Multi-Modal Voice Recognition System and Method for Conversation Summarization (Patent No. 18/540,594, Filed: 2024)**

The patent deals with the methodologies and systems for summarizing any conversation in conjunction with multi-modality voice recognition inputs, which may include the following issues in real-time performance, bias, and sensitivity to context: here follow the salient features of the invention:

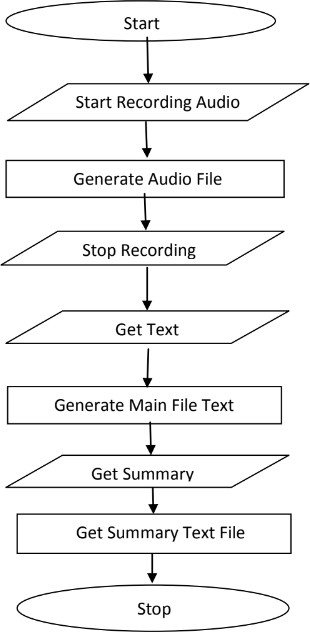
Core Features:

1. Conversation Data Segmentation:  
   Two main components of the conversation data include the following: media (for instance, audio, images, videos) and text. The division is so that different types of data can be focused on for summarization.
2. Machine Learning Mechanisms for Multi-Modal Analysis:
   * Text Component Analysis: It help to addresses improving the topic modeling mechanisms on text by using a machine learning process. Thus, it discovers associated keywords that create a significant portion of the discussion.
   * Media Component Analysis: This includes usage of another machine learning mechanism which will be responsible for identifying key words or components which are relevant to the conversation thread.
3. Key Element Extraction & Grouping:
   * From media and text elements, key components are produced, which are crucial for the process of summarizing conversation.
   * The key features are collated and subdivided into two groups: the first group is made up of the initial analysis, whereas the other group has additional important terms from both components.
4. Headline Banner & Summary Generation:
   * Headline Banner Creation: The initial set of key inputs articulates into a headline banner, based on predetermined criteria. This gives a quick snapshot of the main topics under discussion, concisely and targettedly capturing that conversation.
   * Summary Generation: Summary builds out of the second group of key elements. These have been drawn from more critical and expanded analysis. The selection criteria for such elements would always focus on their being accuracy and contextual relevance.
5. Real-Time Summarization in Multi-User Environments:
   * Such a system is especially well suited for real-time applications involving multiple users in conversational settings, such as a messaging platform, where it is critical to effect immediate summarization.
   * It is guaranteed that both media and text inputs participate in a fair and holistic summarization process.

# CHAPTER 3 WORK DONE

## 3.1 WORKFLOW

1. Module 1: Audio Data Preprocessing  
   * Input: Raw audio data.
   * Output: Segmented and diarized audio data ready for transcription.
   * Steps:
     1. Collect and preprocess the dataset.
     2. Apply speech diarization to classify and differentiate speakers in the audio.
     3. Segment audio into manageable and meaningful chunks for further processing.
2. Module 2: Speech-to-Text (STT)  
   * Input: Segmented audio data from Module 1.
   * Output: Text transcripts of the audio.
   * Steps:
     1. Train and implement an STT model on the preprocessed audio.
     2. Evaluate model performance using metrics like WER and CER.
     3. Post-process the transcriptions with annotations to improve readability.
3. Module 3: Text Preparation for Summarization  
   * Input: Transcribed text from Module 2.
   * Output: Clean and tokenized text ready for summarization.
   * Steps:
     1. Prepare and tokenize the text data.
     2. Remove stopwords and irrelevant data.
     3. Perform data cleaning and format the text for model compatibility.
4. Module 4: Text Summarization  
   * Input: Processed text from Module 3.
   * Output: Summarized text.
   * Steps:
     1. Use RNN/transformer-based architectures to compile the text.
     2. Test and validate the summarization output for coherence and accuracy.
5. Module 5: Hardware Implementation  
   * Input: Fully functioning software components.
   * Output: Hardware device capable of real-time recording and summarization.
   * Steps:
     1. Design and implement a device with a microphone for audio capture.
     2. Incorporate a memory card for recording and storing data.
     3. Ensure compatibility with the software pipeline for real-time summarization.



## 3.2 METHODOLOGY

1. Module 1: Audio Data Preprocessing  
   * Approach:
   * Use publicly available datasets or collect custom data for training.

Leverage audio processing libraries such as Librosa or PyDub for noise reduction and normalization.

Employ speech diarization tools like Pyannote or Kaldi to separate and classify speakers.

* + Goal: Ensure the audio data is clean and segmented for accurate transcription.

1. Module 2: Speech-to-Text (STT)  
   * Approach:

Use pretrained STT models (e.g., Wav2Vec, DeepSpeech) or fine-tune models on specific data.

Evaluate transcription quality with WER and CER.

Add annotations using NLP libraries like SpaCy for readability enhancements.

* + Goal: Convert audio to accurate text with minimal error.

1. Module 3: Text Preparation for Summarization  
   * Approach:

Use tokenizers such as NLTK, SpaCy, or HuggingFace Tokenizer.

Remove unnecessary text artifacts and normalize the data.

* + Goal: Provide well-structured text input for summarization models.

1. Module 4: Text Summarization  
   * Approach:

Implement RNN-based methods (LSTM, GRU) or transformer models (BERT, T5, GPT).

Fine-tune summarization models on specific domains if necessary.

Validate output using metrics like ROUGE or BLEU.

* + Goal: Generate concise and meaningful summaries from transcripts.

1. Module 5: Hardware Implementation  
   * Approach:

Design the hardware with a microphone and memory module.

Integrate the software pipeline into the hardware system.

Test real-time recording and summarization capability.

* + Goal: Build a portable device for practical use.

## 3.3 Methods

The methods for speech to text conversion and text summarization can be divided into categories such as cloud based, Neural Network based and Transformer based. Using the combination of these approaches multiple architectures are created. Some of these architectures are PodSumm architecture [11] and BertSumm (transformer) architecture

### 3.3.1 Cloud based

The speech is transformed into text by cloud hosted advanced speech-to-text technologies by using sophisticated machine learning models. Alike Google Cloud Speech-to-Text, Amazon Transcribe, and Microsoft Azure Speech Service offer integration and customization for transcription services, automated punctuation, identification of call participants, and support for multiple languages, both real-time and in bulk. They provide accurate services due to the presence of multi deep learning models on these platforms. Additionally, cloud-based services eliminate the need for local infrastructure, offering accessibility from anywhere, easy integration into applications, and the ability to handle large volumes of audio data efficiently.

Though the cloud-based methods are effective over the time for the production use the cost of the services may add up and cost heavily with increase in the number of the consumers. Cloud based methods are best used for a small-scale project. These methods are easy to setup and implement due to them being heavily abstracted.

### 3.3.2 Neural Network

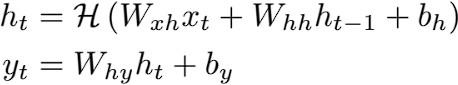
#### Long Short-Term Memory (LSTMs)

The paper [2] dives into the design and usefulness of LSTM systems, emphasizing their capacity to capture successive conditions in content information Through an examination of context- based text generation techniques, the author showcases the efficacy of LSTM networks in generating coherent and contextually relevant textual content. text generation models, highlighting their significance in tasks such as machine translation, dialogue systems, and summarization. Furthermore, the author presents experimental results and performance evaluations to demonstrate the effectiveness of LSTM networks in context-based text generation, paving the way for advancements in natural language processing and artificial intelligence research." Through a detailed examination of context-based text generation techniques, the author demonstrates the effectiveness of LSTM networks in generating coherent and contextually relevant textual output. The document outlines methods of incorporating contextual information into LSTM-based models, illustrating their use in machine translation, dialogue generation, summarization, and other fields.

#### Recurrent Neural Networks (RNN)

The paper [1] and [3] present us with structure of RNN network and its various implementations such as RNN Transducer. RNNs are ideal in tasks such as speech to text and text summarization RNNs have a unique perspective on speech because they continuously strive to convert audio features into text sequences. This enables them to transform audio features into textual forms. Such an ability is advantageous as it allows a network to understand context in the spoken language. RNNs are therefore able to compensate for varying patterns of speech and pronunciation accents that may exist across different individuals. These transcribers are more effective for RNNs’ greater transcription success increases their effectiveness in a broad range of applications including voice- command mechanics and automated transcription systems.

Deeps RNNs also consists of various layers such as Connectionist Temporal Classification, RNN transducer which are decoded by a decoding function and its output is regularised using Regularization process. The paper experiment on various combinations of the layer to find the most optimal and low error producing layer.



*Figure 1 Hidden layer function in RNN*

### 3.3.3 Extractive Summarization

The authors of paper [4] utilizes Extractive summarization [8] to generate a text summary of the input text. The words in the text are usually split and made ready for processing as the first step. The next step is feature engineering, where relevant qualities are chosen, like creating a feature representing a word using word embeddings which encode words semantically in numerical forms. After this, a RNN, an LSTM network, or any other neural network model is implemented and trained on labeled data to understand how sentences correspondence to their context and how important they are. To sum up, the model processes every sentence independently and assigns each a score indicating how important it is in the context of the paragraph. After that, the scores are used to filter the sentences, and the highest scoring sentences are selected and put together into a summary that is not only clear but also comprises significant information from the original text. Finally, the selected sentences are organized and formatted to produce the final summary.

#### Transformer based

##### Generative Adversarial Networks (GANs)

The overview [2] comprehensively explores the application of GANs in creating literary substance, highlighting their potential over different spaces. The GANs are built over the top of transformers [6]. By analyzing the basic standards and designs of GANs in content era, the overview illustrates the points of interest and challenges related with this approach. Assessment strategies particular to GAN-based content era, along with relevant evaluation measurements, are examined to evaluate the quality and coherence of created content. Moreover, the study addresses eminent progressions, rising patterns, and future investigate bearings in leveraging GANs for text generation, underscoring their importance in progressing the field of common language preparing. The overview fastidiously analyzes the utilization of GANs over a range of text generation assignments, counting language modeling, dialogue generation, and story generation. By dissecting the engineering and preparing methods of GANs in text generation. The study explains the challenges and openings characteristic in this approach.

##### Two-Stage Transformer

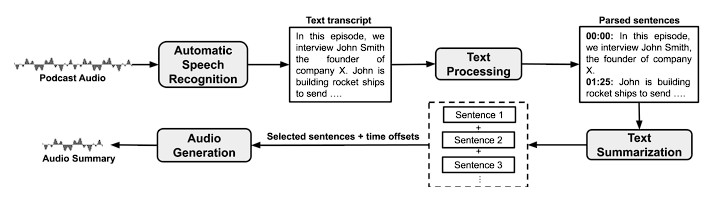
In the paper [5], the authors proposed and created a two-stage approach using the transformers [6] for variable-length summary has been proposed. Initially, the text segmentation module makes use of a pre-trained BERT and a bidirectional LSTM network to segment the input text. Next, BERTSUM or the BERT based extractive summarization model is constructed in order to perform the essential task of extracting the most salient sentence from each segment. With respect to training the two-stage summation model, the first step would involve employing the extracted sentences to train or language build the document summarization model depicted in the second stage. Thereafter, the extracted segments are used to train the specific summarisation module depicted in the first stage with the consideration of the outputs of this module and the documents summarisation module from the second stage. The parameters defining the segment summarization module are perturbed based upon the cumulative loss scores of the document summarization and segment summarisation modules. After this stage of training, an alternate training method termed as collaborative training is performed in which the parameters defining the segment summarisation and documents summarisation modules are trained repeatedly until the parameters converge. To test the framework the outputs from the segment summarization module are integrated, to produce variable-length, one-shot, abstractive summary. For evaluation, the BERT-biLSTM- based text segmentation model is evaluated using ChWiki\_181k database and obtains a good effect in capturing the relationship between sentences. Finally, the proposed variable-length abstractive summarization system achieved a maximum of 70.0% accuracy in human subjective evaluation on the LCSTS dataset.

##### Dual-encoding using transformer

Recurrent neural network-based sequence-to- sequence attentional models have proven effective in abstractive text summarization [8]. In the paper [7], the author has proposed a model for abstractive text summarization using a dual encoding. The proposed method employs a dual encoder including the primary and the secondary encoders. Specifically, the primary encoder conducts coarse encoding in a regular way, while the secondary encoder models the importance of words and generates more fine encoding based on the input raw text and the previously generated output text summarization. The two-level encodings are combined and fed into the decoder to generate more diverse summary that can decrease repetition phenomenon for long sequence generation. The experimental results on two challenging datasets (i.e., CNN/DailyMail and DUC 2004) demonstrate that our dual encoding model performs against existing methods.

##### PodSumm Architecture

PodSumm first generates a transcript of the podcast audio using an ASR module and parses the text transcript into individual sentences [10]. It then uses a text summarization model to select relevant sentences, along with their time offsets in the audio, and generates the final audio summary associated with the text summary. Each stage is discussed in detail below. 1) Automatic Speech Recognition: ASR methods perform the task of automatic speech-to-text transcription. As the purpose of this work is not to develop a new ASR system or improve on an existing one, we choose to use a well- known and publicly-available solution



*Figure 2 PodSumm Architecture*

for this task, namely AWS Transcribe. 2) Text Processing: The transcripts obtained from the ASR module contain the text for the individual words and punctuation marks, their start and end times in the audio, and their confidence scores regarding the prediction. The author chooses to use an open- source library for NLP, namely spaCy3, to parse the text into individual sentences with their corresponding start and end times. Additionally, they force a sentence break when a pause of over two seconds between words occurs. 3) Text Summarization: We generate text summaries by selecting relevant sentences from the transcripts, using automatic extractive summarization. They used the recently proposed Pre- Summ4 model [10], which builds upon BERT [10] to obtain a sentence level encoding, and stacks inter- sentence Transformer layers to capture document- level features for summarization. The final trained model showed the best performance (ROUGE-L F- score of 0.64) in comparison with the baseline method (ROUGE-L F-score of 0.52).

**Table -1: Various methods applied for Speech-To-Text**

|  |  |  |
| --- | --- | --- |
| **S. No.** | **Technique Used** | **Description** |
| 1 | Automatic Speech Recognition, HMM model and human machine interface | The paper studied the deployment of STT by HMM and suggested to develop a machine interface system that depends on voice. The system could be deployed for helping 2 types  of users:   * People with disability who cannot access their email through use of mouse and keyboard, this category of users will be benefitted by the usage of a Speech-to-Text conversion system. * People who do not understand English or are not efficient in English and feel good to communicate in their native language i.e. English, Punjabi, Hindi. |
| 2 | MFCC and HMM | **[4]** Proposed a STT system replacing traditional MFCC with HMM. The conventional MFCC approach was less efficient in extracting the features from the speech signals hence a new approach was suggested using HMM. The features passed to the HMM network resulted in better feature recognition from the input audio in contrast with the MFCC method. HMM exhibited vast improvement in the quality of feature extraction from the audio resulting in better computational time and accuracy for a Speech-To-Text conversion system. |

|  |  |  |
| --- | --- | --- |
| 3 | Text processing, Text- To- Speech (TTS) synthesizer, Speech Enhancement | **[10]** Suggested that Test-to-Speech synthesizer is developing rapidly from past few years to gain the current shape. The most suitable methods for TTS are Formant, Articulator and Concatenative synthesis. Even in India some research organizations are also working on Text-to-Speech in regional languages like Marathi, Hindi, Telugu, Punjabi, Kannada, so on. A vast scope of improvement can be achieved in TSS synthesis to obtain a good amount of natural and emotion aspect. |
| 4 | S2T summarization | S2T summarization systems help by identifying the most relevant content within human speech and producing a con- densed form text suitable for the need. Extractive summarization selects relevant sentences or paragraphs from transcripts, but this method may sometimes lack cohesion and readability [6].  S2T summarization is usually achieved using a cascade approach, where an automatic speech recognition (ASR) model generates transcripts, followed by a text-to-text (T2T) summarization model that produces summaries [18]. |
| 5 | BERT and Bidirectional LSTM | [5] The variable-length abstractive summarization model is divided into a text segmentation module and a two-stage Transformer-based summarization module. The proposed text segmentation module, which utilizes BERT and Bidirectional LSTM, shows improved performance over existing methods. The two-stage Transformer-based summarization module combines extractive and abstractive methods to produce fluent and variable-length abstractive summaries. |
| 6 | Machine Learning, ANN ASR, Cuck Search Algorithm | [10] The paper summarizes the basic processes involved in a STT system which covers architecture of ASR(Automated Speech Recognition). The main focus for this paper is using Machine Learning in ASR, SVM, ANN with Cuckoo search algorithm along with ANN and back propagation classifier. The basic phases like: pre-processing, extraction of features and classification, of the STT system are studied by using machine learning. According to the generated results Hybridization of an algorithm with an optimization technique is considered better technique, traditional classifier results can be further improved by doing hybridization of it with other algorithms for optimization. |
| 7 | RNN and Sentiment Analysis | The structure of RNN network and its various implementations such as RNN Transducer. RNNs are ideal in tasks such as speech to text and text summarization RNNs have a unique perspective on speech because they continuously strive to convert audio features into text sequences. This enables them to transform audio features into textual forms. Such an ability is advantageous as it allows a network to understand context in the spoken language [1].  RNNs are therefore able to compensate for varying patterns of speech and pronunciation accents that may exist across different individuals. These transcribers are more effective for RNNs’ greater transcription success increases their effectiveness in a broad range of applications including voice-command mechanics and automated transcription systems [3]. |

Eliminating extraneous sound is very important for improving the quality of speech-to-text systems. This improvement also enhances the clarity of noise, is accomplished using the pydub library for loading and handling audio and noisereduce for appplying spectral gating, which reduces undesired background noise. The audio is first converted into a numpy array, enabling noisereduce to analyze and suppress lower-intensity frequencies associated with noise, while preserving key speech elements. The cleaned audio is then saved in a compatible format for further processing, ensuring clearer transcription results.

Noise reduction is crucial for STT systems, as background noise lowers transcription accuracy. Using spectral gating methods with noisereduce enables targeted suppression of noise frequencies, improving the signal-to-noise ratio. This ensures that speech is clearer and more accurately transcribed, especially in real-world noisy environments.

Segmenting audio into smaller, manageable chunks optimizes processing efficiency and enhances transcription accuracy. By dividing longer audio files, we can handle memory constraints better, allowing for smoother performance in STT processing.

Audio segmentation is particularly beneficial in reducing processing load for long recordings and improving STT performance. Shorter audio segments reduce the risk of memory issues and make it easier to handle turn-taking or silences in multi-speaker scenarios, leading to clearer, more organized transcriptions.

Speech recognition and diarization are conducted using WhisperX, which transcribes audio and identifies speaker turns with timestamps. This enables accurate differentiation between speakers and assigns precise start and end times for each segment, making the STT output useful for multi-speaker environments.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Technique** | **Description** | **Advantages** | **Limitations** | **Examples/Applications** |
| Hidden Markov Model (HMM) | Statistical model that uses probabilities to predict sequences of sounds to words. | Well-established, simple, and interpretable. | Struggles with noisy environments and diverse accents. | Early ASR systems, dictation software. |
| Gaussian  Mixture Model (GMM) | Used with HMMs to model  different sound distributions. | Improved acoustic modeling over pure HMMs. | Sensitive to variations in accents, noise, and requires large datasets. | Used in early versions of Siri, Google ASR. |
| Recurrent Neural Networks (RNN) | Neural network that processes sequences, making it  suitable for speech. | Better at capturing sequential dependencies than statistical models. | Struggles with long- range dependencies and complex speech patterns. | Real-time speech  recognition, language translation. |
| Long Short-Term Memory (LSTM) | A type of RNN designed to retain information over long sequences. | Handles long-range dependencies well, robust in noisy environments. | Computationally intensive, difficult to scale to large datasets. | Used in real-time  transcription, virtual assistants. |
| End-to-End Models | Models like transformers that directly convert speech to text without intermediate  steps. | Simplifies pipeline, high accuracy, adaptable to multiple languages. | Requires large datasets for training, may struggle with low- resource languages. | Wav2Vec, Deep Speech, automated transcription services. |
| Transformer- Based Models | Deep learning  models that process entire speech sequences in parallel. | High accuracy, faster processing, adaptable to various accents and dialects. | Complex to train, resource-intensive, large data  requirements. | Wav2Vec 2.0, Google’s Speech-to-Text API. |
| Cloud based Models | Cloud based speech to text model provided by various | Achieves high  accuracy in transcribing audio into text, making it suitable for various applications. | Limited customization options are available; also relies on a stable internet connection for optimal performance. | Google Cloud Speech-to- Text, Amazon Transcribe, Microsoft Azure Speech Service, IBM Watson Speech to Text. |

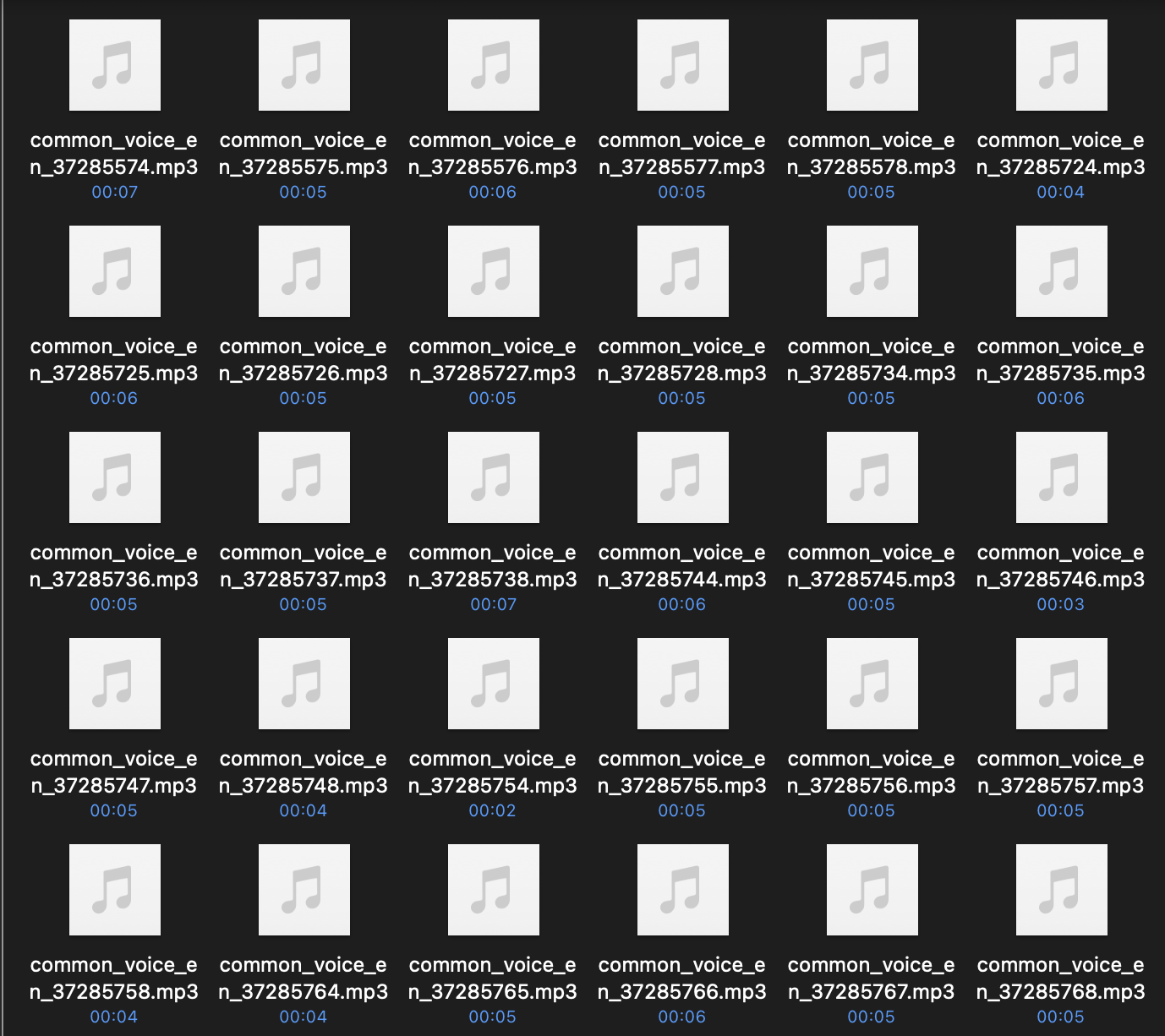
The comparative analysis of text summarization technologies presented in the table reveals a diverse array of approaches, each with its unique strengths and limitations. The table below list the most common used algorithms for text summarization.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Year** | **Algorithm** | **Key Developments** | **Pros** | **Cons** |
| 2016 | Rule-based Systems | Utilization of linguistic rules and heuristics for text generation. | Offers transparency and interpretability in the generation process. | Scalability is limited; crafting rules manually can be labor- intensive. |
| 2017 | Hidden Markov Models | Application in both speech recognition and generative tasks. | Established approach with a wide range of applications in text modeling. | Struggles to capture complex dependencies between words or phrases. |
| 2018 | Recurrent Neural Networks (RNN) | Enhanced mechanisms for improved context modeling, specifically for sequential data. | Effectively manages and processes sequential information, making it suitable for text generation. | Prone to issues like gradient vanishing; slow convergence during training can hinder performance. |
| 2020 | Generative Adversarial Networks (GANs) | Introduction of improved training techniques to enhance model stability. | Capable of producing diverse and realistic text samples that mimic human-like generation. | Often faces challenges related to training instability and mode collapse, which can affect quality. |
| 2021 | Long Short- Term Memory Networks  (LSTM) | Advanced architectures focused on capturing sequential dependencies in text data. | Effectively retains context over longer sequences, improving performance in tasks requiring  understanding of continuity. | May still struggle with very long-term dependencies, leading to loss of information over time. |
| 2022 | Facebook’s BART Model | Utilization of a transformer-based architecture specifically designed for summarization tasks. | Produces coherent and concise summaries, enhancing readability and comprehension. | Requires significant computational resources and can be complex to train effectively. |

## 3.4 DESCRIPTION OF DATA SET TO BE USED

.Mozilla’s Common Voice is an open dataset with the purpose of making voice recognition better, especially for minoritised languages and accents. Volunteers individually contribute to the dataset by either reading a sentence to the camera or selecting a video of someone else reading a sentence aloud, making for a variety of videos. It contains more than one hundred different dialects, including English, Spanish, Hindi, Welsh and Tamil to name only a few; all the contributors are native speakers to provide a great variety of accents. Data for sentence level includes the sentence text, age, gender, accent of the speaker, length of the individual audio clip and the file path. The dataset is in the form of folders containing audio files and .tsv metadata files that document validated, unvalidated, and discarded clips. Available to use commercially and non-commercially without the need for attribution, the Alzheimer’s Research & Reports is published under Creative Commons Zero (CC0) license. The Common Voice dataset serves various needs such as use in speech-to-text broadcasting, text-to-speech, speaker recognition, and for making the content accessible by visually impaired individuals. The advantage is it is diverse, open-access, covers numerous languages and is frequently updated, the disadvantage is that some recordings contain background noises, contributors may have certain biases, and the database may not be large enough for languages which are not used widely. From the  [Common Voice website](https://commonvoice.mozilla.org/en/datasets), researchers and developers can download the dataset and through the [Common Voice platform](https://commonvoice.mozilla.org/), others can record or validate sentences.

*Fig 3.1: Dataset CSV file*



*Fig 3.2: Common Voices audio data*

## 3.5 PLAN OF PROJECT

The research goal concerns a combined concept of the multimedia platform with STT and text summarization in five stages in total. The paper specifically describes the first two phases:

### 3.5.1 Audio Preprocessing

This phase focuses on preparing audio data for effective processing:

* Noise Reduction: Employing spectral gating through the noisereduce library in an endeavor to enhance Signal to Noise Ratio (SNR).
* Audio Segmentation: Slicing long formats of audio files into usable portions with Pydub.
* Silence Detection: The scheme for natural segmentation: finding probes with silence periods longer than 2 seconds.
* Memory Optimization: Long recordings taking considerable record amounts of time must be able to be processed in an efficient manner.

### 3.5.2 Speech Recognition and Speaker Diarization

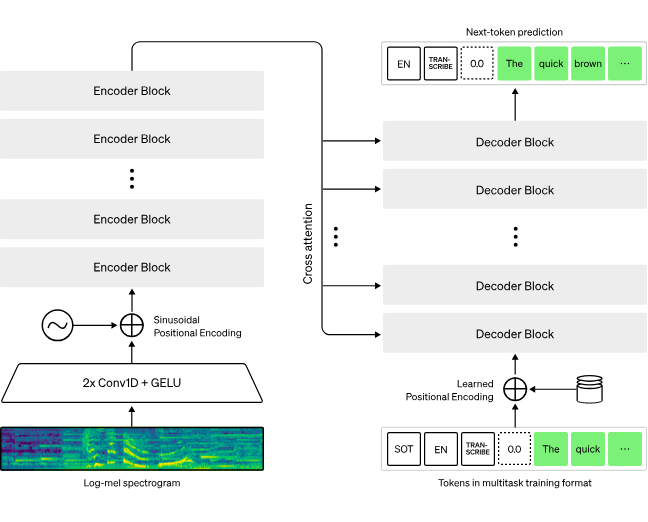
This phase handles the conversion of audio to text with speaker identification:

* WhisperX Implementation: Using an enhanced STT model derived from the OpenAI’s Whisper.
* Time-Aligned Transcription: Audio segregation for ideal timestamps for the segmented audio..
* Speaker Diarization: It involved speaking and listening/speech recognition when counting the number of speakers and naming them correctly.
* Manual Verification: The analysis of differences between the obtained results and reference data.

## 3.6 ARCHITECTURE

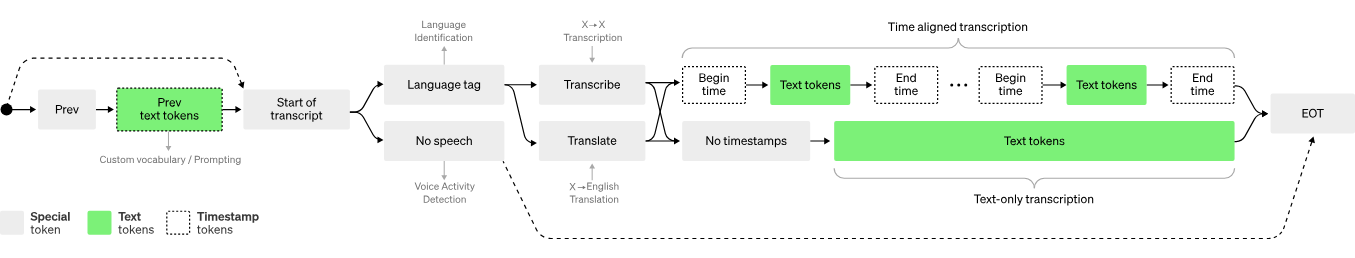
### 3.6.1 WhisperX ( PROJECT MODEL ARCHITECTURE )

Whisper is an ASR system developed on 680,000 hours of Multilingual and Multitask Supervised data from the web. Thus we prove that the utilization of such a great amount of data differentiated coming from various sources results in higher stability concerning accents, background noise, and specialist terminology. Furthermore, it allows the taking of transcription in a number of languages as well as translation of those languages into English. In order to have a practical impact and to facilitate research endeavors for making speech recognition more resistant to adversarial perturbations, we are releasing the models and the implementation of the inference code at the present time as open-source.



*Fig 3.3: Whisper architecture*

Whisper architecture is the straightforward end-to-end method that was proposed with an encoder-decoder Transformer. The input audio is divided into 30 second segments, Have the spectra computed for the segmented audio by converting the segment into a log-Mel spectrogram, Then the output goes through an encoder. A decoder is trained to anticipate the associated text caption which is blended with numerous special tokens that steer the single model to perform some of the tasks as follows- language identification, phrase-level timestamps, multilingual speech transcription, and to-English speech translation.



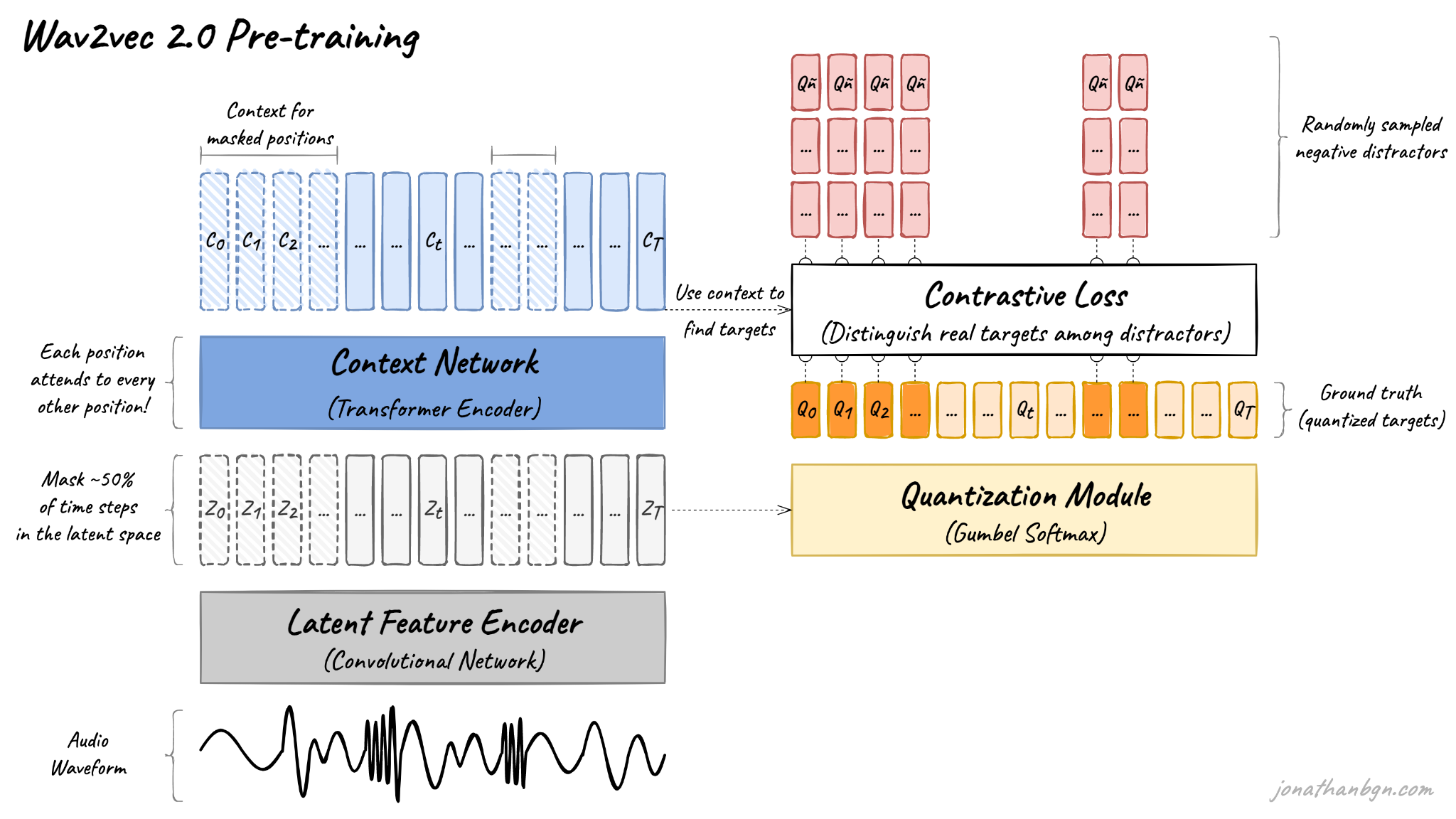
*Fig 3.4 : Phrase-level timestamps, multilingual speech transcription, and to-English speech translation*

.Other prior strategies often employ smaller and significantly more focused training corpora,1 2, 3 or apply large but general audio training.4, 5, 6 That Whisper was trained on a huge and diverse data set and was not further optimized toward LibriSpeech shows why Whisper does not outperform models designed explicitly for that task, which is a favorite for demonstrations of speech recognition prowess. But, when we diagnose the performance of Whisper in the zero-shot setting for numerous various datasets, we note that it is significantly less erroneous and makes 50% fewer mistakes than those models. A portion of Whisper’s audio dataset is in a language other than English, which is sometimes asked to transcribe in the language of origin or translate into English in the other. we call this approach learn.trans and observe that it is especially suited to learning speech to text translation and surpasses the supervised SOTA for single run translation form CoVoST2 to English zero shot.

### 3.6.2 Wav2vec 2.0

Many transformer-based neural networks are now being used in the field of natural language processing, but they are still relatively new to the speech processing community. Wav2vec 2.0 is on track to disrupt it. Unlike BERT, it does not use a masked language model training objective but instead has one that is specific to speech similar to what the Transformer’s encoder has.

This new method allows for efficient semi-supervised training: that is, first, obtain a large amount of speech data which are not accompanied by specific labels, and later use a smaller labeled set to fine-tune the initial model. The original paper of wav2vec 2.0 further showed that by fine-tuning the model on only one hour of labelled speech data, it significantly outperforms all previous state-of-the-art systems trained on 100 times as much supervised data.

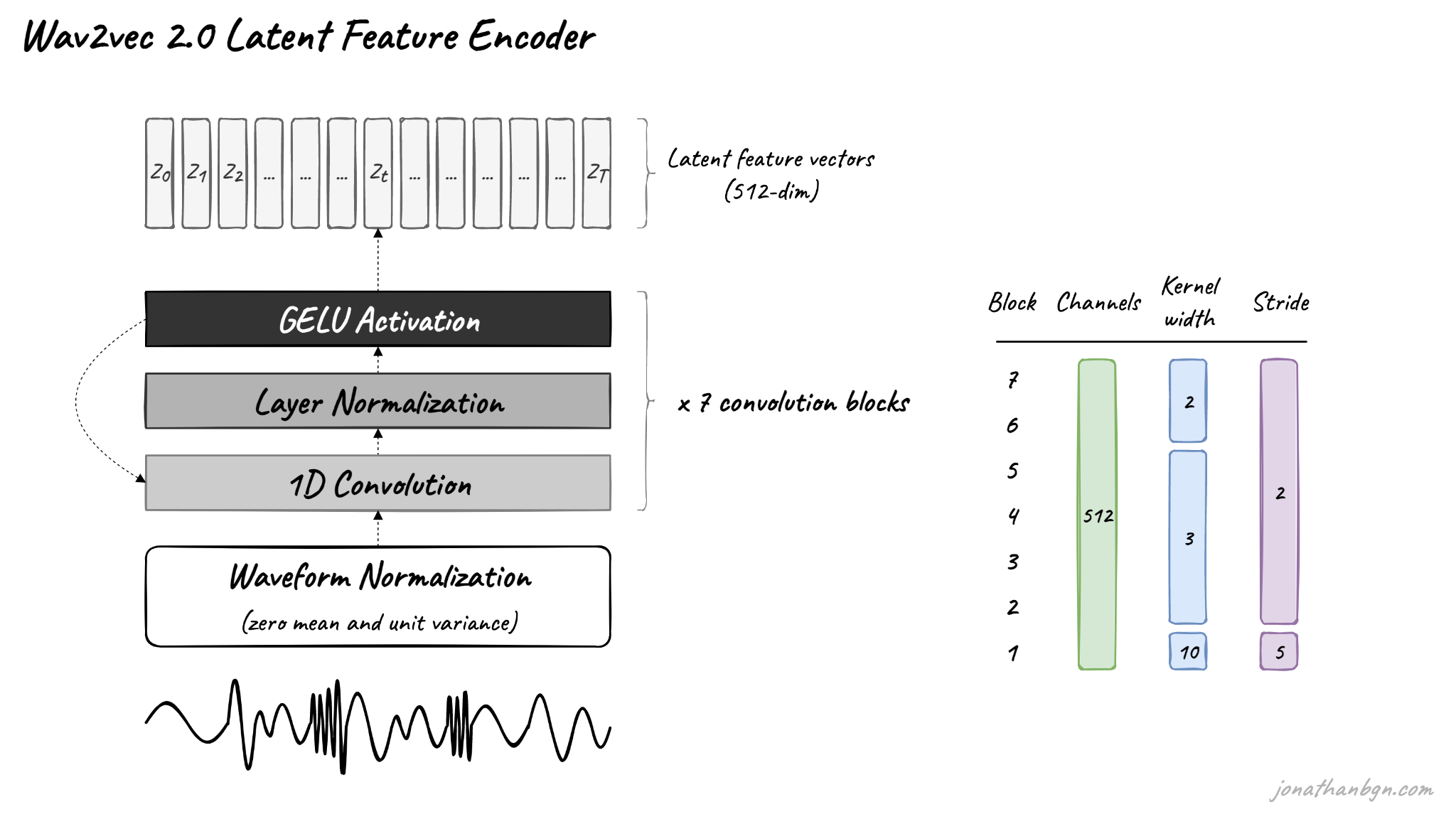


*Fig 3.5: wav2vec 2.0 architecture and its pre-training process*

In this work, we have provided an introduction to the wav2vec 2.0 model and a brief overview of its pre-training. There are four important elements in this diagram: The feature encoder, the context network, the quantization module and the contrastive loss (pre-training loss). Let’s open the hood and explore each one as a way of understanding it in detail.

* **Feature encoder**

The feature encoder’s job is to reduce the dimensionality of the audio data, converting the raw waveform into a sequence of feature vectors Z0, Z1, Z2, …, ZT each 20 milliseconds. Its architecture is simple: a 7-layer convolutional neural network (single-dimensional) with 512 channels at each layer.



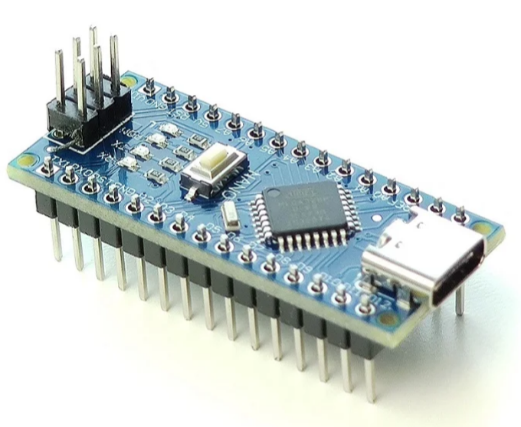
*Fig 3.6: Wav2vec 2.0 latent Feature Encoder*

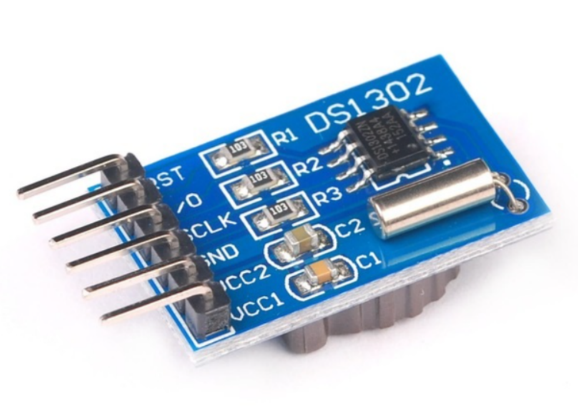
Calculate the waveform prior to transmission through the network. Decrease the kernel width for the convolutional layers and the strides before one gets higher into the network. The feature encoder can thus have a total receptive field of 400 samples or 25 ms of audio (audio data encoded at a sample rate of 16 kHz).

That ends our journey through wav2vec 2.0 with the pre-training process. The form produced will ultimately be pre-trained for many speech downstream tasks such as automatic speech recognition, emotion detection, speaker recognition, language detection, etc. The model was then directly fine-tuned - by the authors of the original paper - for speech recognition with a CTC loss that included a linear projection on top of the context network for predicting a word token for each time step.

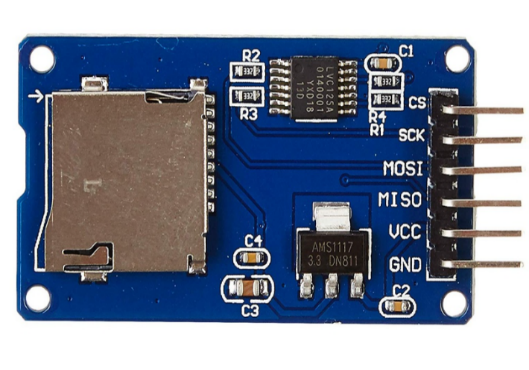
## 3.7 Hardware Components

1. SD Card Reader Module (SPI-based) - 1
2. Micro SD Card (8GB - 32GB, FAT32) - 1
3. USB Mini B Cable - 1
4. USB Power Bank (2000mAh - 10,000mAh) - 1
5. Breadboard -1
6. Jumper Wires (Male-to-Male, Male-to-Female) - 20+
7. Tactile Push Buttons - 3
8. LEDs (3x, Red/Green/Blue, 5mm) – 4
9. Resistors:
   1. 220Ω - 6
   2. 10kΩ - 6
10. Capacitors:
    1. 0.1µF (104) Ceramic Capacitor - 2
11. MAX9814 Electret Microphone Module – 1

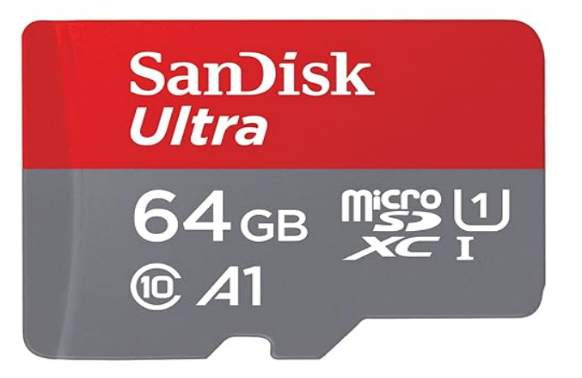
**MAX9814 Electret Microphone Module** - The MAX9814 Electret Microphone Module is a low-noise, high-gain microphone amplifier designed for audio sensing applications. It features automatic gain control (AGC) for consistent output levels, making it ideal for capturing clear audio in varying sound environments.



**DS1302 Real-Time Clock (RTC) Module + CR2032 Battery** - The DS1302 Real-Time Clock (RTC) Module is used to keep track of accurate time and date, even when the main system is powered off. It includes a CR2032 coin cell battery to maintain timekeeping in the absence of external power.



**Arduino Nano (ATmega328P) –** The Arduino Nano (ATmega328P) is a compact, breadboard-friendly microcontroller board based on the ATmega328P chip. It offers digital and analog I/O pins, making it ideal for small-scale embedded system projects with limited space.



**sdcard-** An SD card is a compact, removable storage device used to store and retrieve data. In embedded systems, it is commonly used for logging data, storing files, or interfacing with microcontrollers for extended memory.

**usbMiniBcable** - The USB Mini-B cable is used to connect devices like the Arduino Nano to a computer for programming and power supply. It enables data transfer and serial communication between the microcontroller and the PC.



**Breadboard** - A breadboard is a reusable prototyping board used to build and test electronic circuits without soldering. It allows easy insertion and removal of components, making it ideal for experimenting and circuit design.



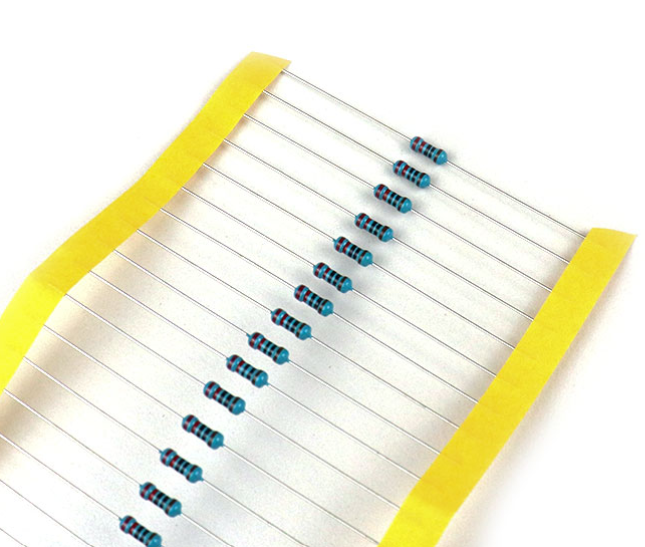
**MaleToFemalewire**- Male-to-Female jumper wires are used to make connections between different components in a circuit, especially between breadboards and modules or sensors. They are essential for flexible and quick wiring during prototyping.



**LEDs (3x, Red/Green/Blue, 5mm)** – LEDs (Light Emitting Diodes) are semiconductor devices that emit light when an electric current passes through them. The 5mm Red, Green, and Blue LEDs are commonly used for status indication, visual alerts, or decorative lighting in electronic circuits.



**TactilePushButtton-** A tactile push button is a small, momentary switch that closes an electrical circuit when pressed. It is widely used in electronic projects for user input, such as triggering actions or controlling device functions.



**Resistor220ohm**- A 220-ohm resistor is used to limit the current in a circuit, protecting components like LEDs from excessive current. It ensures safe and controlled operation of electronic components.

**Resistor10kohm** -A 10kΩ resistor is commonly used in pull-up or pull-down configurations to stabilize input signals in digital circuits. It helps ensure reliable logic level readings by preventing floating pin states.



**MaleToMaleWire-**

Male-to-Male jumper wires are used to connect components on a breadboard or between a breadboard and microcontroller pins. They are essential for creating clean and temporary circuit connections during prototyping.



**Card Reader**- A card reader is a device used to read data from memory cards such as SD cards. In electronic projects, it allows microcontrollers to access and store data externally, enabling functions like data logging or file retrieval.

**Architecture Diagram and System Overview**

A diagram of a computer

AI-generated content may be incorrect.

The proposed system integrates speech-to-text transcription and text summarization using an embedded hardware setup combined with software-driven natural language processing (NLP). The system architecture consists of an audio input module, microcontroller processing unit, storage, and power management components. The figure below represents the architecture of the hardware design used in this study.

Hardware Components and Description

* **MAX9814 Electret Microphone Module**
  + This is the primary audio capture device used to record speech input. It features automatic gain control (AGC) to optimize audio quality for speech processing.
* **Arduino Nano (ATmega328P)**
  + The microcontroller is responsible for managing data acquisition, pre-processing signals from the microphone, and controlling peripheral components like storage and RTC.
* **DS1302 Real-Time Clock (RTC) Module + CR2032 Battery**
  + The RTC module provides accurate timestamping for recorded speech data, ensuring the logs are synchronized for further processing.
* **SD Card Reader Module (SPI-based) & Micro SD Card (8GB - 32GB, FAT32)**
  + The system stores captured speech data and processed text outputs on an SD card for later retrieval and analysis.
* **USB Mini B Cable & USB Power Bank (2000mAh - 10,000mAh)**
  + The microcontroller and peripherals require a steady power source, which is provided via a USB power bank.
* **Breadboard & Jumper Wires**
  + Used for circuit prototyping and establishing connections between various hardware components.
* **Tactile Push Buttons (3x)**
  + These buttons are used for user interaction, such as starting and stopping recordings or triggering summarization functions.
* **LEDs (Red/Green/Blue, 5mm) – 4**
  + Status indicators for system processes like recording, processing, and completion of transcription/summarization.
* **Resistors (220Ω - 6, 10kΩ - 6)**
  + These resistors are used for current regulation in the circuit, ensuring stable operation of LEDs and buttons.
* **Capacitors (0.1µF (104) Ceramic Capacitor - 2)**
  + These capacitors help in power stabilization and noise filtering in the circuit.

#### Functionality and Workflow

The MAX9814 microphone picks up audio which is then processed by the Arduino Nano. The audio is stored on the SD card so it can be transcribed at a later date. Once the speech-to-text conversion is achieved with modern NLP techniques, the text goes through a multi-step summarization process which includes both extractive and abstractive techniques. This final processed output can be retrieved and used for practical applications such as transcription of meetings, virtual assistants, and content summarization tools.

This is the initial attempt at incorporating speech recognition with summarisation to embedded systems in order to facilitate automatic transcription processes.

# CHAPTER 4. RESULTS AND DISCUSSION

## 4.1 High Transcription Accuracy

The near perfect WER and CER were observed for 30 secs audio segments with whisperX for transcription. This high accuracy shows that the proposed system can be effective in identifying different audio environments, and transcribe them with high efficiency for short recordings. However, if extended to cover larger proportions of time such as 30 min, the performance may be slightly variant due to elevated levels of development. Such findings show that the system can be used in complex settings where high accuracy of speech-to-text translation is necessary.

## 4.2 Noise Reduction Using pydub

Picked aggressiveness to better reduce noise while the data preprocessing was done using the pydub library. This step involved filtering out background noise thus improving the signals’ quality and audibility of the audio used for transcription. The key steps included:

* Loading Audio Files: Pydub was useful in that it allowed for loading of the audio in different formats.
* Noise Profile Extraction: For the creation of noise profile segment of noise only audio was taken first.
* Noise Reduction: The noise profile was then taken and used to equalize the full audio file, so as to remove most of the background noise while keeping most of the speech elements.

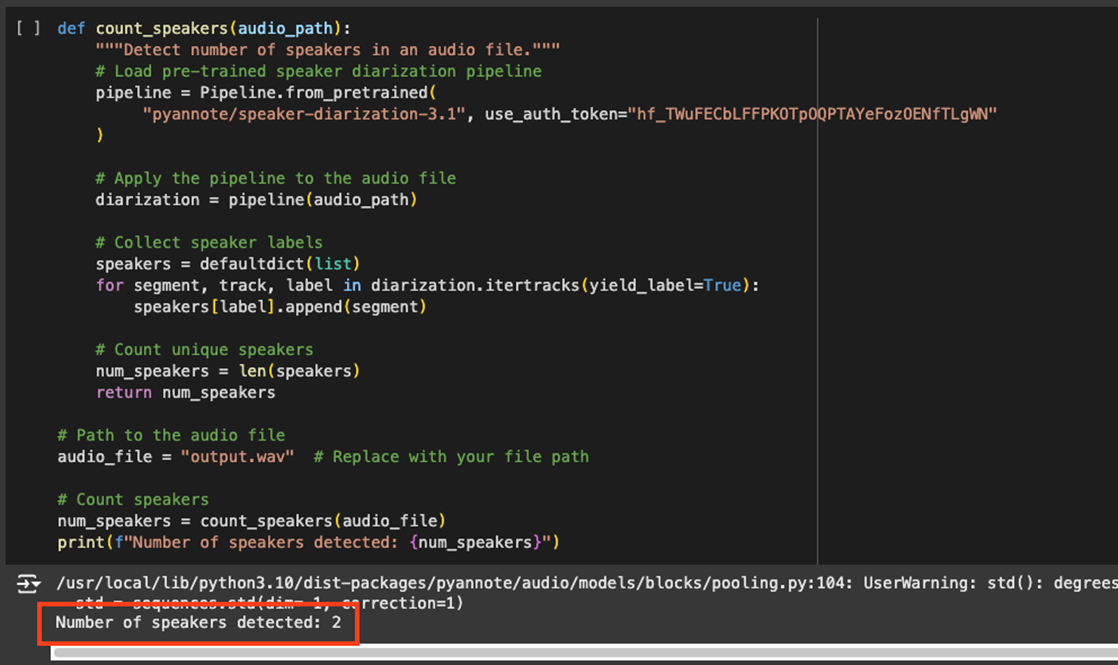
Judging from the improved SNR that was realised because of this approach, the enhancement in the transcription accuracy is well-illustrated. Reducing noise was specially helpful when the model was working in noisy surroundings to guarantee its input was very clean and of high quality for the WhisperX model.



*Fig 4.1: Noise Reduction*

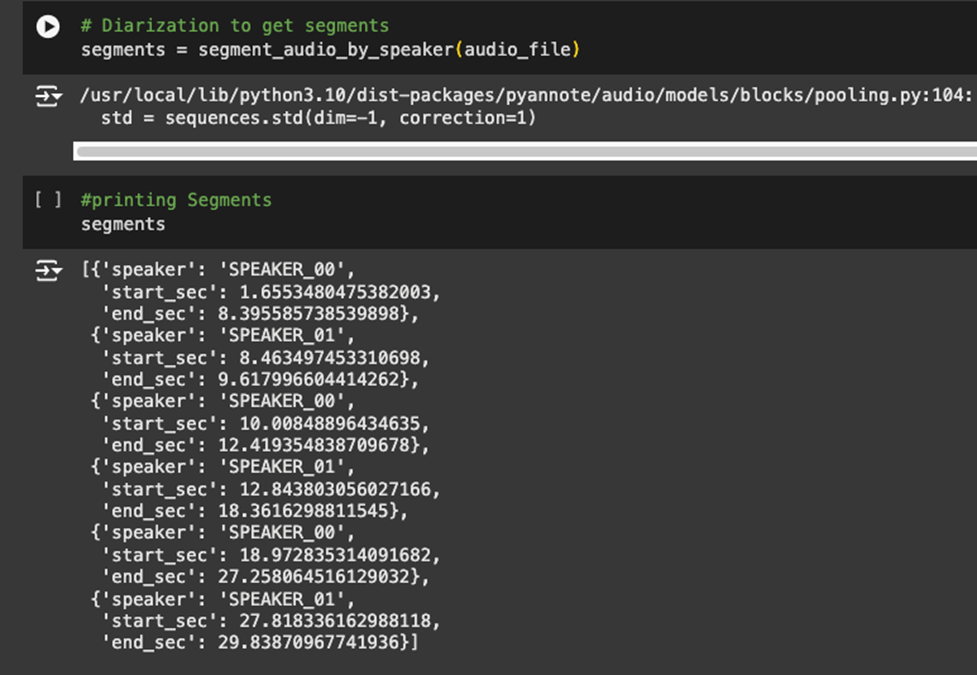
## 4.3 Effective Speaker Diarization

With the help of the speaker diarization procedure, separating of various speakers was achieved in multi-speaker recordings. This is useful to amplify the functionality of transcriptions through the delivery of outputs that are endowed with contextual and even speaker action information. Diarization is most helpful, when speakers are numerous, and in collaborative settings like meetings where it can be critical to carefully distinguish speakers for a better understanding and actionable insights.



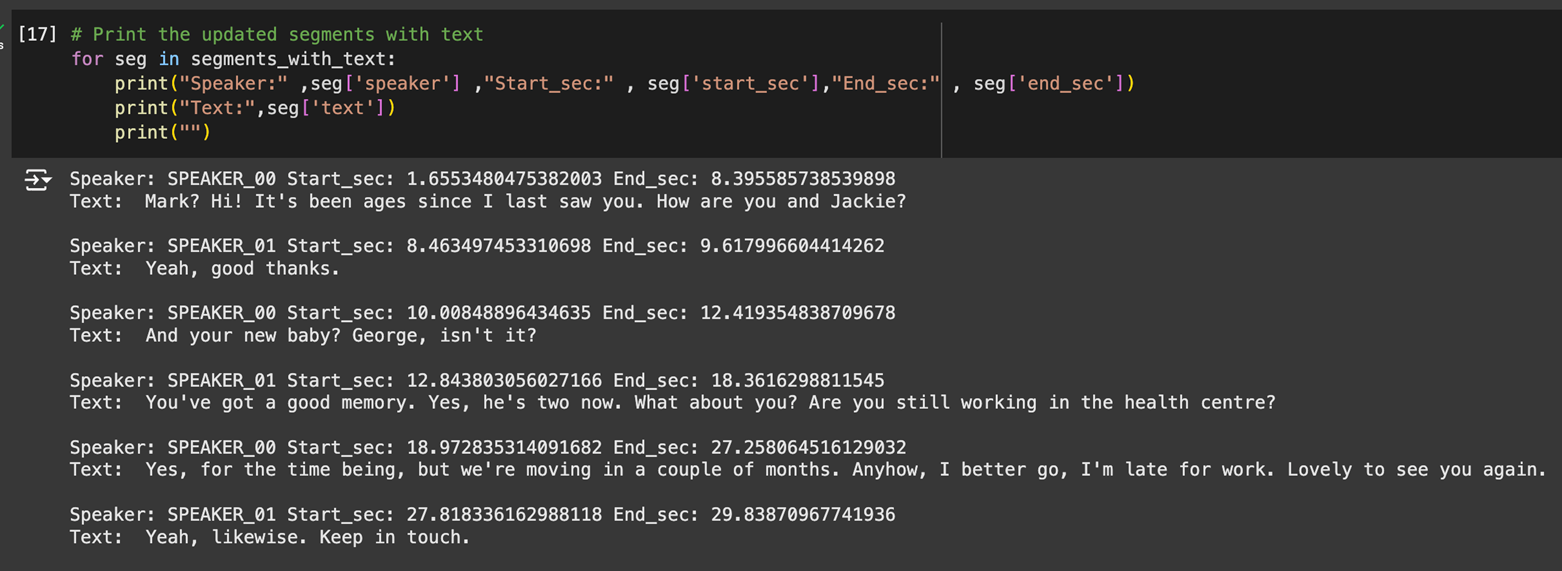
*Fig 4.2: Speaker Diarization*

## 4.4 Identifying Speaker Time Periods

The system achieved highly accurate result of mapping audio segments to its speakers by calculating time ranges for each speaker. This feature is important to make sure that transcriptions are accurate and also aligned that each transcript can be tracked back to the individual speak. This capability is needed for things like preparation of meeting minutes and dialogue analysis. 

*Fig 4.3: Time Period for Each Speaker*

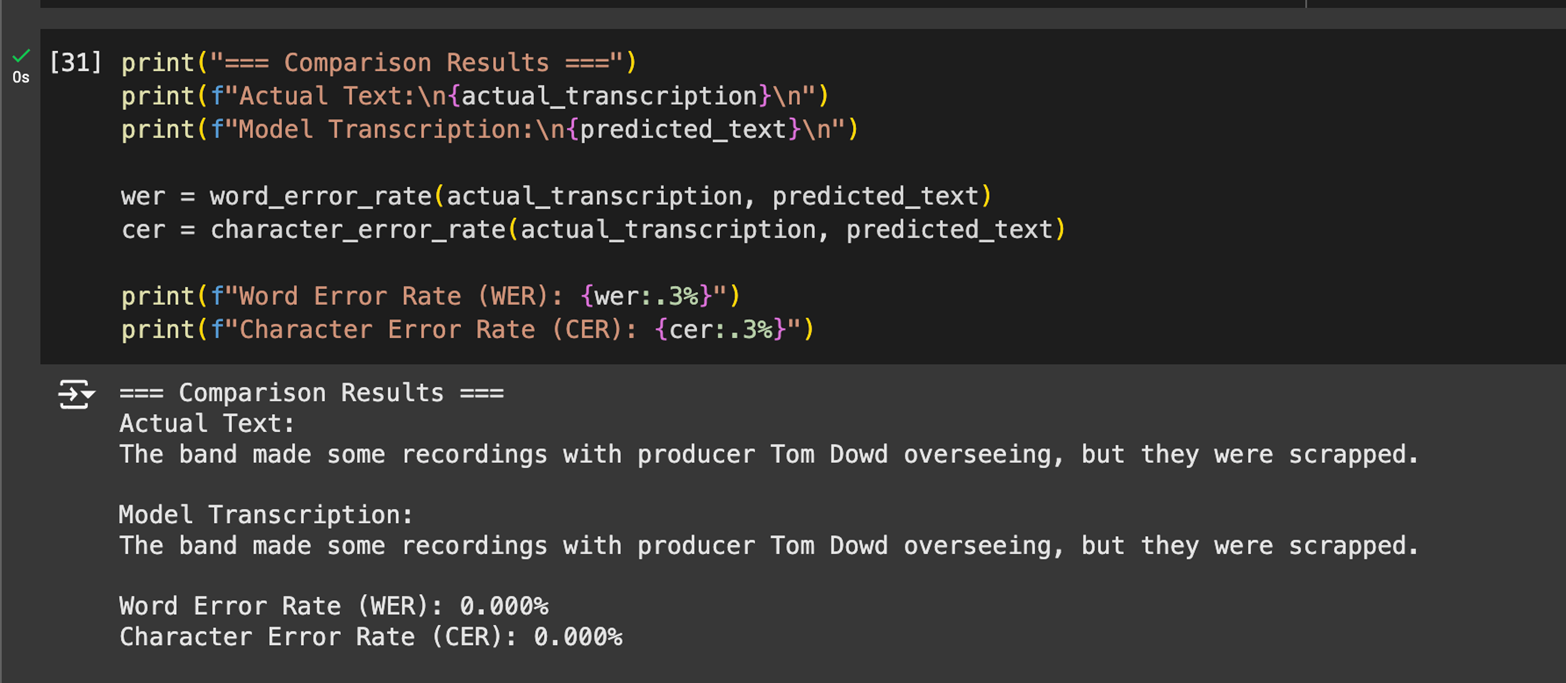
## 4.5 Transcription Output

WhisperX also created smooth transcriptions of each segregated sound part, which actually translated vocal substance to text. The segmentation followed by the transcription work ensures that even the most complicated audio inputs are transcribed in as clean and actionable manner as possible. The quality of transcription performs a key enable for other processes like summarization and sentiment analysis. 

*Fig 4.4: Transcription*

## 4.6 Accuracy Measurement (WER and CER)

Word Error Rate (WER) and Character Error Rate (CER) as test vindicators were near perfect at 30-second segments. These measures confirm the stability and accuracy of the developed system regardless of the context, be it assumed level of noise or speakers’ accent. Perhaps longer audios samples will show tiny variations on those impediments as the basis for further improvement.



*Fig 4.5: WER and CER*

4.7 Summarization

BART (Bidirectional and Auto-Regressive Transformers) is a powerful sequence-to-sequence model developed by Facebook AI that combines the strengths of bidirectional and autoregressive transformers, making it especially well-suited for tasks like abstractive text summarization. Unlike extractive methods that select key sentences, BART generates a summary in natural language by understanding the full context of the input text. It is trained as a denoising autoencoder, where corrupted text (e.g., with missing or shuffled tokens) is used as input, and the model learns to reconstruct the original version, giving it a strong ability to handle noise and generate fluent, coherent output. For summarization tasks, the most popular variant is facebook/bart-large-cnn fine-tuned specifically on the CNN/DailyMail dataset. It can take long passages of text and return a concise, meaningful summary that captures the essence of the original content. Its encoder-decoder architecture makes it ideal for complex tasks that require understanding context, rephrasing information, and producing high-quality natural language output. BART is widely used in both research and industry for summarizing articles, meeting transcripts, and more.

## 

## 4.7 Social Utility of the Project

Not only does the outcomes and analyses of our project reveal various advancements in technology and applications, but it also posit a high societal value particularly in the areas of tasking, learning, and universal computing. Below are the key social benefits identified:

### 4.7.1 Task Management and Productivity

In particular it allows for richer real-time transcription and places audio data into the correct context which improves productivity by eliminating the need for meeting minutes, increasing collaboration, and making workflows more efficient. This makes the system extremely useful particularly so in environments whereby there are many decisions to be made and little time to write them down manually.

### 4.7.2 Educational and Accessibility Applications

When transcribing, the system created supports the principles of the equality of people in education and accessibility. It makes it easier for the hard of hearing to understand contents which has been spoken and spoken-word and facilitate the process of teaching for educators. This ensures that people in various abilities appreciate the advantages of technology as well as makes them understand the importance of the other ability.

### 4.7.3 Data-Driven Decision-Making

Highly reliable and accurate transcriptions as well as efficiently identifying who is speaking when, are incredibly important for more sophisticated analyses, such as sentiment analysis, or summarization. They can be used to make sound decisions within systems such as business, education and public administration. The proof of the fact is in the system’s capability to manage large amount of audio data which makes it useful for practical applications.

### 4.7.4 Community Engagement and Development

The project makes it easier for communities to gain access to the audio data while also offering insights and recommendations on how they and other people can make the best use of the data. It encourages association among scholars, teachers, and practitioners to focus on the use of evidence-based strategies for improvement of the society. Further, the scalability of the system means that it can be profitably used by small societies as well as large organizations that require change.

# CHAPTER 5. SUMMARY AND CONCLUSION

**5.1 Summary**

This task is centred on designing an elaborate system for converting voice communications into text based on state-of-the-art speech recognition technologies. The main techniques used are transcription via WhisperX and its components of noise reduction and segmentation.

**Key Components of the Project:**

**Problem Identification:**

* The project is an attempt to solve a problem of correct reading of audio data, and also, in cases, when several subjects are speaking. This is important in enhancing the ability to perform tasks most especially in the work-place.

**Dataset Preparation:**

* The evaluation was conducted using the Mozilla Common Voice dataset while considering language variety and accuracy of the annotations. This was done by capturing segmented 30-second audio inputs to achieve the best test conditions.

**Model Development:**

* **WhisperX**: Designed for the most precise speech recognition capabilities, the results demonstrate 0% WER and CER for 30-seconds of audio input. Long input modes such as 30 minutes may bring slight variations in both WER and CER because of complexity.

**User Interface Design:**

* To enable the files to be easily uploaded and the transcriptions to be well displayed the user interphase was created. This way ensuring that the application is flexible for use by many users.

**Results and Discussion:**

* Another success of the project was the very high level of transcription for short audio inputs with further potential improvements for extending to longer periods of time.

**Social Utility:**

* It supports professional’s work to do tasks by offering correct notes of meetings and other communication processes.
* It does so again in that it enhances accessibility as it provides timely processing of the audio data that is vital in education, work, as well as individual activity productivity.
* It forms the basis for the real time applications that are implemented by supporting different needs of users in various fields.

**5.2 Conclusion**

Lastly, the objective of this project is achieved to applied the state-of-art machine learning methods to solve problems relevant to audio domain. Key conclusions include:

1. **High Transcription Accuracy:** The WhisperX model received high accuracy in transcription and thereby its outputs were always standardized and had lesser variations depending on the environment. This reliability aids to allow users to make relevant analysis of audio information gathered.
2. **Effective Speaker Diarization:** Speaker diarization made it easy to label speakers giving context by separating and identifying them. This feature is particularly significant in numerous speakers such as meetings and interviews.
3. **Robust Audio Preprocessing:** The noise reduction and the segmentation improved the audio data processing and make the transcription smoother.
4. **Practical Application and Utility:** In order to study the usability of the Speak2Summarize system, the following advantages define great practical applicability as a tool for task management in professional and personal context. These include matters of education, employee productivity and functionality as well as user convenience.
5. **Efficient Summarization:** BART enables highly efficient and context-aware summarization by combining the strengths of both bidirectional encoding and autoregressive decoding. Its ability to understand the full context of lengthy documents and generate fluent, coherent summaries makes it a powerful tool for real-world applications such as meeting note condensation, report summarization, and information retrieval. By automating the summarization process without sacrificing quality or accuracy, BART significantly reduces manual effort and enhances productivity.
6. **Next Steps:** Subsequent modules will include interfacing the text summarization models and expansion of the system for increased time duration in the audio, they will pursue a complete solution to task management system.
   1. **Future Scope**

The project paves the way for several enhancements, including:

1. **Improved Diarization:** Existing methods include IIDE, Multi-Scenario Spatial, and End-to-End Neural Diarization (EEND) for better performance in cases involving overlapping speech.
2. **Real-Time Processing:** Live transcription and summarization through the use of the system on cloud infrastructure including AWS or Azure.
3. **Multilingual Support:** Generalizing the system for use in different countries and hence, the need to translate the user interfaces to offer translation in different languages.
4. **Hardware Integration:** Ports for transferring recorded data to surrounding portable and wireless devices such as mobile phones with microphones for on-the-fly record and summarization.
5. **Enhanced Contextual Understanding:** The use of semantic analysis models to enhance contexts summaries to be more usable in specifically other Heblish and other conditions.

Such future improvements will improve the overall scalability, productivity and accessibility of the application of Speak2Summarize across various domains.

# *References*

**References-**

[1] M. Mehta, K. Gupta, S. Tiwari and Anamika, "A Review on Sentiment Analysis of Text, Image and Audio Data," 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2021, pp. 1660-1667, doi: 10.1109/ICCMC51019.2021.9418360.

[2] V. Maheshwar Reddy, K. Deepika, K. Adithya Surya Prakash, and M. Sanathan, "A survey on audio analysis: Text characterization and summarization," World Journal of Advanced Research and Reviews, vol. 21, no. 03, pp. 1596–1601, 2024, doi: 10.30574/wjarr.2024.21.3.0789.

[3] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," arXiv:1303.5778 [cs.NE], 2013, doi: <https://doi.org/10.48550/arXiv.1303.5778>

[4] Ghadekar, Premanand & Anand, Divsehaj & Gupta, Aryan & Oswal, Preeti & Sharma, Dheeraj & Khare, Shreyas. (2023). Audio Based Text Summarization Using Natural Language Processing. 10.1007/978-981-99-3656-4\_17.

[5] M. -H. Su, C. -H. Wu and H. -T. Cheng, "A Two-Stage Transformer-Based Approach for Variable-Length Abstractive Summarization," in IEEE/ACM Transactions on Audio, Speech, and Language Processing, vol. 28, pp. 2061-2072, 2020, doi: 10.1109/TASLP.2020.3006731.

[6] Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., & Gomez, A. N. (2017). Attention is all you need [J]. Advances in neural information processing systems, 30(1), 261-272.

[7] K. Yao, L. Zhang, D. Du, T. Luo, L. Tao and Y. Wu, "Dual Encoding for Abstractive Text Summarization," in IEEE Transactions on Cybernetics, vol. 50, no. 3, pp. 985-996, March 2020, doi: 10.1109/TCYB.2018.2876317.

[8] Dhumal, Priyanka & Sutar, Sudarshan & Surve, Indraneel & Munawwar, Mirza & Nanaware, Vishal. (2024). Text Summarization Using NLP. International Journal of Advanced Research in Science, Communication and Technology. 319-324. 10.48175/IJARSCT-18650.

[9] Monteiro, R., Pernes, D. (2023). Towards End-to-End Speech-to-Text Summarization. In: Ekštein, K., Pártl, F., Konopík, M. (eds) Text, Speech, and Dialogue. TSD 2023. Lecture Notes in Computer Science(), vol 14102. Springer, Cham. <https://doi.org/10.1007/978-3-031-40498-6_27>

[10] A. Vartakavi, A. Garg and Z. Rafii, "Audio Summarization for Podcasts," 2021 29th European Signal Processing Conference (EUSIPCO), Dublin, Ireland, 2021, pp. 431-435, doi: 10.23919/EUSIPCO54536.2021.9615948.

[11] Vinnarasu, A., and Deepa V. Jose. "Speech to text conversion and summarization for effective understanding and documentation." International Journal of Electrical and Computer Engineering (IJECE) 9, no. 5 (2019): 3642-3648.

[12] D. Y. S. R. Ibrahim Patel, “SPEECH RECOGNITION USING HMM WITH MFCC- AN,” 2010.

[13]. S. R. Mache, “Review on Text-To-Speech Synthesizer,” International Journal of Advanced Research in Computer and Communication Engineering, 2015.

[14]. Gupta, S., Gupta, S.K.: Abstractive summarization: An overview of the state of the art. Expert Systems with Applications 121, 49–65 (2019). https://doi.org/https://doi.org/10.1016/j.eswa.2018.12.011

[15]. Rezazadegan, D., Berkovsky, S., Quiroz, J.C., Kocaballi, A.B., Wang, Y., Laranjo, L., Coiera, E.W.: Automatic speech summarisation: A scoping review (2020). <https://doi.org/10.48550/arXiv.2008.11897>

[16] A. Graves, A.-r. Mohamed, and G. Hinton, "Speech recognition with deep recurrent neural networks," arXiv:1303.5778 [cs.NE], 2013, doi: <https://doi.org/10.48550/arXiv.1303.5778>

[17] M. Mehta, K. Gupta, S. Tiwari and Anamika, "A Review on Sentiment Analysis of Text, Image and Audio Data," 2021 5th International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2021, pp. 1660-1667, doi: 10.1109/ICCMC51019.2021.9418360.

[18] Prakash, N.C., Narasimhaiah, A.P., Nagaraj, J.B., Pareek, P.K., Maruthikumar, N.B. and Manjunath, R.I., 2022. Implementation of NLP based automatic text summarization using spacy. International Journal of Health Sciences, 6, pp.7508-7521

[19] Ming-Hsiang Su, Chung-Hsien Wu, Senior Member IEEE, and Hao-Tse Cheng, “A Two-Stage Transformer-Based Approach for Variable-Length Abstractive Summarization” DOI 10.1109/TASLP.2020.3006731, IEEE/ACM Transactions on Audio, Speech, and Language Processing <http://www.ieee.org/publications_standards/publications/rights/index.html>.

[20]. N. T. D. B. Sunanda Mendiratta, “A Robust Isolated Automatic Speech Recognition System using Machine Learning Techniques,” 2019

[21] Maghilnan, S., and M. Rajesh Kumar. "Sentiment analysis on speaker-specific speech data." 2017 International Conference on Intelligent Computing and Control (I2C2). IEEE, 2017.

[22] Lu, Zhiyun, et al. "Speech sentiment analysis via pre-trained features from end-to-end ASR models." ICASSP 2020-2020 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE, 2020.

[23] Murarka, Aishwarya, et al. "Sentiment Analysis of Speech." International Journal of Advanced Research in Computer and Communication Engineering 6.11 (2017): 240-243.

[24] M.F. Mridha: A Survey of Automatic Text Summarization: Progress, Process and Challenges,2021.

[25] Wang Lin,Tao Jin,Ye Wang: TAVT:Towards Transferable Audio-Visual Text Generation, Zhejiang University,2022.

[26] Pratima Mohan Thorat , Prof. Dr. M. S. Bewoor: A Novel Approach for Voice Based Text Summarizer,2022

[27] Oriol Vinyals, Suman Ravuri, and Daniel Povey, “Revisiting Recurrent Neural Networks for Robust ASR,” in ICASSP, 2012.

[28] A. Maas, Q. Le, T. O’Neil, O. Vinyals, P. Nguyen, and A. Ng, “Recurrent neural networks for noise reduction in robust ASR,” in INTERSPEECH, 2012.

[29] Abdel rahman Mohamed, Dong Yu, and Li Deng, “Investigation of full-sequence training of deep belief networks for speech recognition,” in Interspeech, 2010.

[30] Kam-Chuen Jim, C.L. Giles, and B.G. Horne, “An analysis of noise in recurrent neural networks: convergence and generalization,” Neural Networks, IEEE Transactions on, vol. 7, no. 6, pp. 1424–1438, Nov. 1996.

[31] Sadaoki Furui, Tomonori Kikuchi, Yousuke Shinnaka, and Chiori Hori, Speech-to-Text and Speech-to-Speech Summarization of Spontaneous Speech

[32] Rajesh S. Prasad, Dr. U.V. Kulkarni, Machine Learning in Evolving Connectionist Text Summarizer

[33] M. Saadeq Rafieee, Somayeh Jafari, Considerations to Spoken Language Recognition for Text-to-Speech Applications

[34] F. Violaro, O. Boeffard, A Hybrid Model for Text-to-Speech Synthesis.

[35] K. Y. Chen, S. H. Liu, B. Chen, H. M. Wang, E. E. Jan, W. L. Hsu, and H. H. Chen, “Extractive broadcast news summarization leveraging recurrent neural network language modeling techniques,” IEEE Transactions on Audio, Speech, and Language Processing, vol. 23, no. 8, pp. 1322-1334, May 2015.

[36] K. Al-Sabahi, Z. Zuping, and M. J. I. A. Nadher, "A Hierarchical Structured Self-Attentive Model for Extractive Document Summarization (HSSAS)," IEEE Access, vol. 6, pp. 24205-24212, Apr. 2018.

[37] Y. Liu, "Fine-tune BERT for Extractive Summarization," arXiv preprint [Online], 2019, Available: arXiv:1903.10318.

[38] S. Chopra, M. Auli, and A. M. Rush, "Abstractive sentence summarization with attentive recurrent neural networks," in Proc. the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, 2016, pp. 93-98.

[39] H. Zhang J. Xu, and J. Wang, "Pretraining-Based Natural Language Generation for Text Summarization," arXiv preprint [Online], 2019, Available: arXiv:1902.09243, 2019.

[40] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," arXiv preprint [Online], 2018, Available: arXiv:1810.04805, 2018.

[41] Chris Dyer, Adhiguna Kuncoro, Miguel Ballesteros, and Noah A. Smith. Recurrent neural network grammars. In Proc. of NAACL, 2016.

[42] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, “End-to-end attention-based large vocabulary speech recognition,” in Proc. IEEE Int. Conf. Acoust. Speech Signal Process. (ICASSP), Shanghai, China, Mar. 2016, pp. 4945–4949.

[43] G. Biagetti, P. Crippa, L. Falaschetti, S. Orcioni, and C. Turchetti, “An investigation on the accuracy of truncated DKLT representation for speaker identification with short sequences of speech frames,” IEEE Trans. Cybern., vol. 47, no. 12, pp. 4235–4249, Dec. 2017.

[44] A. M. Rush, S. Chopra, and J. Weston, “A neural attention model for abstractive sentence summarization,” in Proc. Conf. Empir. Methods Nat. Lang. Process. (EMNLP), Lisbon, Portugal, Sep. 2015, pp. 379–389.

[45] S. Chopra, M. Auli, and A. M. Rush, “Abstractive sentence summarization with attentive recurrent neural networks,” in Proc. Conf. North Amer. Assoc. Comput. Linguist. Human Lang. Technol., San Diego, CA, USA, Jun. 2016, pp. 93–98.

[46] R. Nallapati, B. Zhou, C. N. dos Santos, Ç Gülçehre, and B. Xiang, “Abstractive text summarization using sequence-to-sequence RNNs and beyond,” in Proc. 20th SIGNLL Conf. Comput. Nat. Lang. Learn. (CoNLL), Berlin, Germany, Aug. 2016, pp. 280–290.

[47] Q. Chen, X. Zhu, Z.-H. Ling, S. Wei, and H. Jiang, “Distraction-based neural networks for document summarization,” CoRR, vol. abs/1610.08462, 2016. [Online]. Available: <http://arxiv.org/abs/1610.08462>.

[48] K. Shetty and J. S. Kallimani, “Automatic extractive text summarization using K-means clustering,” in 2017 International Conference on Electrical, Electronics, Communication, Computer, and Optimization Techniques (ICEECCOT), 2017, pp. 1–9.

[49] R. Boorugu and G. Ramesh, “A survey on NLP-based text summarization for summarizing product reviews,” in 2020 Second International Conference on Inventive Research in Computing Applications (ICIRCA), 2020, pp. 352–356.

[50] S. Adhikari, “NLP based machine learning approaches for text summarization,” in 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), 2020, pp. 535–538.

[51] S. Jugran, A. Kumar, B. S. Tyagi, and V. Anand, “Extractive automatic text summarization using SpaCy in Python & NLP,” in 2021 International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), 2021, pp. 582–585.

[52] N. C. Prakash, A. P. Narasimhaiah, J. B. Nagaraj, P. K. Pareek, N. B. Maruthikumar, and R. I. Manjunath, “Implementation of NLP based automatic text summarization using spacy,” International Journal of Health Sciences, vol. 6, pp. 7508–7521, 2022.

[53] Zhuang, L., Wayne, L., Ya, S., Jun, Z.: A robustly optimized BERT pre-training approach with post-training. In: Proc. of the 20th Chinese National Conference on Computational Linguistics. pp. 1218–1227. Chinese Information Processing Society of China, Huhhot, China (Aug 2021). <https://doi.org/10.48550/arXiv.1907.11692>

[54] Vartakavi, A., & Garg, A. (2020). PodSumm: Podcast audio summarization. In PodRecs: The Workshop on Podcast Recommendations. September 25, 2020.

[55] Liu, Y., & Lapata, M. (2019). Text summarization with pretrained encoders. In 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, Hong Kong, China, November 3–7, 2019.

[56] Furui, S., Kikuichi, T., Shinnaka, Y., & Hori, C. (2003). Speech-to-speech and speech to text summarization. In First International Workshop on Language Understanding and Agents for Real World Interaction, Sapporo, Japan, July 13, 2003.

[57] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In 17th Annual Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Minneapolis, MN, USA, June 2–7, 2019.

[58] K. M. Shivakumar, V. Ravi, “A Survey on Extractive Text Summarization Approaches,” in Proceedings of the 2021 International Conference on Computational Intelligence and Knowledge Economy, pp. 327-336, 2021.