A Project Report

On

# Abstractive Summarization and Automated Minutes of Meeting for Virtual Meetings

Submitted in partial fulfillment of the requirement of
the Degree of

Bachelor of Technology
In
Computer Engineering

Submitted By

**Alashi Harsh Kishor**

**Sankpal Shreyas Shashikant**

**Sase Omkar Ramkrishna**

**Zade Chaitanya Pramod**

Supervisor
**Prof. K.S. Charumathi**

**Department of Computer Engineering**

**PILLAI COLLEGE OF ENGINEERING**

**New Panvel – 410 206**

**UNIVERSITY OF MUMBAI**

**Academic Year 2023-24**

DEPARTMENT OF COMPUTER ENGINEERING

PILLAI COLLEGE OF ENGINEERING

New Panvel – 410 206

# CERTIFICATE

This is to certify that requirements for the B.Tech. Project Report A entitled **‘Abstractive Summarization and Automated Minutes of Meeting for Virtual Meetings’** have been successfully completed by the following students:

| **Name** | **Roll No.** |
|---|---|
| Alashi Harsh Kishor | DALP1703 |
| Sase Omkar Ramkrishna | DALP1731 |
| Zade Chaitanya Pramod | DALP1737 |
| Sankpal Shreyas Shashikant | NIS710 |

in partial fulfillment of Bachelor of Technology in the Department of Computer Engineering, Pillai College of Engineering, New Panvel – 410 206 during the Academic Year 2023 – 2024.

______________________________

**Supervisor**
**Prof. K.S. Charumathi**

________________________ ________________________

**Head of Department** **Principal**
**Dr. Sharvari Govilkar** **Dr. Sandeep Joshi**

DEPARTMENT OF COMPUTER ENGINEERING
PILLAI COLLEGE OF ENGINEERING
New Panvel – 410 206

# PROJECT REPORT APPROVAL

This **Fourth Year Project A** report entitled "**Abstractive Summarization and Automated Minutes of Meeting for Virtual Meetings**" by **Alashi Harsh Kishor, Sankpal Shreyas Shashikant, Sase Omkar Ramkrishna and Zade Chaitanya Pramod** are approved for the degree of B.Tech. in **Computer Engineering.**

**Examiners:**

**1.** ______________________
**2.** ______________________

**Supervisors:**

**1.** ______________________
**2.** ______________________

**Chairman:**

**1.** ______________________

**Date:**

**Place:**

# Declaration

We declare that this written submission for B.Tech. Project Report Declaration entitled **"Abstractive Summarization and Automated Minutes of Meeting for Virtual Meetings"** represents our ideas in our own words and where others' ideas or words have been included. We have adequately cited and referenced the original sources. We also declared that we have adhered to all principles of academic honesty and integrity and have not misrepresented or fabricated or falsified any ideas / data / fact / source in our submission. We understand that any violation of the above will cause disciplinary action by the institute and also evoke penal action from the sources which have thus not been properly cited or from whom paper permission has not been taken when needed.

Project Group Members:

Alashi Harsh Kishor: ____________

Sankpal Shreyas Shashikant: ____________

Sase Omkar Ramkrishna: ____________

Zade Chaitanya Pramod: ____________

**Date:**

**Place:**

# Table of Contents

# Abstract

For professional meetings, it is common practice to produce meeting minutes which get distributed to the meeting participants containing information on meeting agenda, attendee list and action item list with task owner names and task deadlines. The aim is to automate the process of producing meeting minutes and meeting summary with the help of AI and Deep Learning. The audio from meeting recording will be converted to text using a unique approach for Speech to Text which involves detecting accent of speaker and transcribing their audio using model trained for their specific accent improving the Speech to Text accuracy. The transcript summarization task will be done by BART pre-trained model for abstractive summarization, fine-tuned using conversational text datasets like AMI and ICSI corpus. Sentiment analysis is to be used to determine action items, tone of speaker and overall tone of the meeting. Meeting bot/extension will send live statistics about attendees and meeting chat. The summary, action items and Minutes of Meeting document will be made available to the dashboard of the host. The objective of easing and automating the management of virtual meetings will be achieved with this project and better decisions can be made in organizations that rely on virtual meetings.

# List of Figures

# List of Tables

# Chapter 1

# Introduction

## 1. Fundamentals

Abstractive summarization of minutes of meetings for virtual meetings involves using natural language processing (NLP) techniques to automatically generate a concise and coherent summary of the key points discussed during a meeting. Here are some fundamentals of abstractive summarization for virtual meetings:

1. Pre-processing the data: The first step is to pre-process the data by converting the audio or video recordings of the meeting into text format. This can be done using automatic speech recognition (ASR) technology or by manually transcribing the recording.

2. Identifying key information: The next step is to identify the key information discussed during the meeting. This can include topics, decisions, action items, and important points made by participants.

3. Generating a summary: Once the key information has been identified, the abstractive summarization algorithm uses NLP techniques to generate a concise and coherent summary of the meeting. This involves understanding the context of the discussion, identifying important keywords and phrases, and using natural language generation (NLG) techniques to create a summary that is easy to read and understand.

4. Evaluating the summary: Finally, the generated summary is evaluated to ensure that it accurately reflects the key points discussed during the meeting. This can be done manually by a human reviewer or through automated metrics such as ROUGE (Recall-Oriented Understudy for Gisting Evaluation).

## 2. Objective

The objective of this project is to automate the process of producing meeting minutes and meeting summaries for professional virtual meetings using AI and Deep Learning techniques. The project aims to improve the efficiency and accuracy of the meeting minutes and summaries, reduce the need for manual transcription, and provide valuable insights into the meeting content and participants' sentiment. The ultimate goal is to ease and automate the management of virtual meetings, enabling better decision-making in organizations that rely on virtual meetings.

## 3. Scope

1. Speech to Text Conversion: Develop and implement a unique approach for Speech to Text conversion that can accurately transcribe the audio recordings of virtual meetings.

2. Abstractive Summarization: Fine-tune the BART pre-trained model for abstractive summarization using conversational text datasets to generate accurate and concise meeting summaries.

3. Sentiment Analysis: Implement sentiment analysis techniques to identify the tone of the meeting, tone of the speaker, and action items that need to be addressed.

4. Live Statistics: Develop a meeting bot that can provide live statistics about the attendees and meeting chat, such as the number of attendees, how long they were present, and what was discussed in the chat.

5. Dashboard Integration: Integrate the meeting summary, action items, and Minutes of Meeting document with the dashboard of the host, making it easy for the host to review and share the documents with other participants.

6. Testing and Validation: Test and validate the system on different virtual meeting recordings and datasets to ensure the accuracy and efficiency of the system.

7. Deployment and Maintenance: Deploy the system on a cloud-based platform and provide ongoing maintenance and support to ensure the system's reliability and scalability.

## 4. Outline

I. Introduction

- Background on the need for automating the production of meeting minutes and summaries for virtual meetings
- Objective of the project

II. Speech to Text Conversion

- Explanation of the unique approach for speech to text conversion
- Details on how accents are detected and how the model is trained for specific accents
- Importance of speech to text accuracy for meeting summaries

III. Abstractive Summarization

- Overview of the BART pre-trained model
- Description of the fine-tuning process using conversational text datasets
- Importance of accurate and concise meeting summaries

IV. Sentiment Analysis

- Explanation of how sentiment analysis is used to determine action items, tone of speaker, and overall tone of the meeting
- Importance of identifying important information from the meeting content

V. Meeting Bot/Extension

- Details on how the meeting bot/extension provides live statistics about attendees and meeting chat
- Importance of having a system that can provide real-time updates during virtual meetings

VI. Dashboard Integration

- Explanation of how the summary, action items, and Minutes of Meeting document are made available to the dashboard of the host
- Importance of having an easily accessible system for review and sharing of meeting documents

VII. Testing and Validation

- Overview of the testing process to ensure accuracy and efficiency of the system
- Importance of testing on different virtual meeting recordings and datasets

VIII. Deployment and Maintenance

- Explanation of how the system will be deployed on a cloud-based platform
- Details on ongoing maintenance and support to ensure reliability and scalability

IX. Conclusion

- Recap of the objectives and outcomes of the project
- Future directions for the project

# Chapter 2

# Literature Survey

## 2.1 Introduction

In recent times, virtual meetings have become an integral part of professional organizations. To manage these meetings effectively, producing accurate meeting minutes and summaries is essential. This abstract proposes the use of AI and deep learning to automate the process of producing meeting minutes and summaries. The proposed system involves converting meeting recordings to text using a unique speech-to-text approach that considers the speaker's accent. The text data will be summarized using a pre-trained BART model fine-tuned with conversational text datasets. Additionally, sentiment analysis will be used to determine the tone of the meeting and identify action items. The meeting bot/extension will provide live statistics about attendees and meeting chat. The system aims to make the summary, action items, and minutes of meeting document available to the host's dashboard. This project's primary objective is to ease and automate the management of virtual meetings and improve the decision-making process in organizations that rely on virtual meetings.

## 2.2 Literature Review

### 2.2.1 Summarization [1][2][3][4][5][6][7]

The literature review presents five papers that discuss different approaches to summarizing lengthy conversations, including pre-training models, multi-stage summarization frameworks, and strategies for abstractive summarization of code-switched and meeting dialogues. The papers introduce new datasets, methods, and techniques for achieving better summarization performance, such as incorporating sentiment analysis, named entity recognition, and visual data. However, some limitations and challenges are also highlighted, such as the complexity and computing requirements of the models, the lack of human-annotated datasets, and potential biases in the evaluation process. These insights can inform future research on dialogue summarization and advance our understanding of how to capture and convey the essence of long conversations effectively.

### 2.2.2 Action Items [8][9][10][11]

The first article proposes a system for generating to-do items from emails by identifying task sentences and extracting context using supervised algorithms. The second article presents a model for action item detection in meetings, which improves state-of-the-art performance but still faces challenges such as small and unbalanced datasets. The third article focuses on automatically rephrasing action items in meeting transcripts for clearer communication and collaboration, while the fourth article presents a dataset preparation method for extracting decision elements from meeting transcripts, but may not be effective for other communication types.

### 2.2.3 Speech to Text [12][13][14]

In summary, the research papers present pyannote.audio, a comprehensive framework for speaker diarization using end-to-end neural implementations. The framework provides pre-trained models for various tasks and supports on-the-fly data augmentation. The proposed approach achieved state-of-the-art performance in most of these tasks, but its applicability to datasets in other languages has not been tested. The suggested framework may require significant processing resources, which could limit its use in some applications.

### 2.2.4 Emotion Sentiment [15]

The papers presents an approach to neural machine translation (NMT) that takes into account not only the previous and next sentences but also a wider context of the entire document. The approach also utilizes a novel attention mechanism that focuses on the relevant parts of the source text during the translation process.

The proposed method achieves state-of-the-art performance on several benchmark datasets for machine translation, indicating its effectiveness in capturing contextual information and improving translation quality. The method is adaptable and can be used for various languages and domains.

However, the approach requires a large amount of training data and may be computationally expensive. Additionally, the attention mechanism may not always correctly identify the most relevant parts of the source text, potentially leading to inaccuracies in the translation.

### 2.2.5 Topic Segmentation [16]

The paper proposes an unsupervised approach to topic segmentation using BERT embeddings, which results in more accurate representations of the text's semantic meaning. It is a flexible and scalable method, as it does not require annotated data or prior knowledge of the topics. The approach is thoroughly evaluated on various benchmark datasets and demonstrates its effectiveness in comparison to other state-of-the-art techniques. However, the approach assumes a consistent topic structure, and it has only been tested on English-only meetings. Additionally, the complexity of the computations might make real-time segmentation impractical.

### 2.2.6 Automated Minutes of Meeting [17][18][19]

For the three reviewed papers, they all propose approaches to automate the generation of meeting minutes, reducing the workload of human participants and saving time. They rely on crystal-clear audio recordings of the meeting and may give rise to ethical issues. However, it is uncertain how well they would perform in meetings of different types. The first paper and the third paper use deep learning techniques, while the second paper uses a bot with an algorithm that ensures precision and consistency. The second and third papers can generate meeting minutes in real-time.

## 2.3 Limitations of Existing System or Research Gap

Based on the limitations highlighted in the literature reviews, there are several potential research gaps that could be explored:

i. Develop methods that can perform well on a wide range of meeting types and languages, rather than being limited to specific scenarios or languages.
ii. Investigate methods for generating meeting minutes that are more interpretable, allowing users to understand how the system is making decisions.
iii. Explore ethical issues related to the use of automated meeting minute generation, such as the possibility of excluding certain participants or introducing bias into the minutes.
iv. Develop methods that are more robust to noisy or low-quality audio recordings, which are common in real-world meeting settings.

v. Investigate the use of other forms of data, such as video or chat logs, in combination with audio recordings to improve the accuracy and completeness of automated meeting minute generation.

### 2.4 Literature Summary

| TITLE | ADVANTAGES | LIMITATIONS |
| --- | --- | --- |
| **1. Summarization** | | |
| a. DIALOGLM: Pre-trained Model for Long Dialogue Understanding and Summarization (2022) | 1. Offers a pre-training approach for interpreting and summarising lengthy conversations that beats cutting-edge models on a variety of datasets and tasks.<br>2. Introduces a window-based denoising method for generative pre-training that is meant to deal with the special traits of lengthy talks.<br>3. Adds sparse attention to the model to enable it to handle lengthier inputs. | 1. Even if the model performs better than cutting-edge models, it could still have trouble with subtler and more intricate dialogue patterns.<br>2. The suggested method is only relevant to text-based discussions and might work with other modalities, like speech. |
| b. SUMMN: A Multi-Stage Summarization Framework for Long Input (2022) | 1. It helps to maintain context information for interpreting utterances by first producing a coarse summary in several phases, followed by a final fine-grained summary. | 1. The SUMMN multi-stage architecture, particularly when the number of stages is considerable, may lead to an increase in |

| | | |
|---|---|---|
| | 2. SUMMN can handle larger texts by breaking them up into smaller parts and creating summaries for each section, utilising the full potential of pretrained language models. | computing complexity.<br>2. Additionally, SUMMN's application to other tasks, such extractive summarization, is constrained because it is primarily intended for abstractive summarization jobs. |
| c. An Exploratory Study on Long Dialogue Summarization: What Works and What's Next (2021) | 1. Explores the topic of long dialogue summary.<br>2. Presents a thorough analysis of the summarising techniques currently in use.<br>3. Outlines the main difficulties and shortcomings of the current approaches.<br>4. New approaches and strategies for summarising lengthy dialogue are suggested. | 1. Other types of text summarising are not included by the study, which is restricted to summarising lengthy discussions.<br>2. The study is based on a particular dataset, which might not be typical of all long conversations. |
| d. GupShup: An Annotated Corpus for Abstractive Summarization of Open-Domain Code-Switched Conversations (2021) | 1. The article offers a brand-new task of abstractive summarization of open-domain code-switched written discussions that may be used as input to | 1. The absence of significant datasets with human-annotated summaries has been one of the main obstacles to accurately |

| | | |
|---|---|---|
| | other downstream NLP models, like intent classification, question answering, and item suggestion.<br>2. The authors introduce GupShup, a brand-new corpus with over 6,800 code-switched Hindi-English dialogues and associated human-annotated summaries in both Hindi and English. To train and test models for the task of abstractive summarization of code-switched talks, researchers can make effective use of the dataset. | summarising textual interactions.<br>2. The Hindi-English language pair is the main emphasis of the paper, limiting the applicability of the suggested strategies to other code-switched language combinations. |
| e. Abstractive Summarization of Meeting Conversations (2020) | 1. The research suggests a novel method for abstractive summarization of meeting dialogues that incorporates various NLP approaches, such as sentiment analysis and named entity recognition.<br>2. In terms of ROUGE scores—a metric frequently used to assess the quality of summarization—the | 1. The proposed approach is not thoroughly evaluated in the research using other criteria, such as the coherence, fluency, or readability of the generated summaries.<br>2. The research does not address the constraints or potential biases of the experimental dataset, which would |

| | | |
|---|---|---|
| | proposed strategy outperformed a number of baseline models. | limit the applicability of the suggested strategy to different meeting contexts or domains. |
| f. On Faithfulness and Factuality in Abstractive Summarization (2020) | 1. Due to the use of sophisticated sequence-to-sequence structures, including attention and copy mechanisms and fully attention-based Transformer topologies, conditional text generation models are able to produce language that is human-like, fluent, and has a high degree of coherence.<br>2. The resulting text's accuracy and reliability have greatly increased thanks to trained language modelling for natural language understanding. | 1. Extreme abstractive document summarization is the only task that the study focuses on, hence it might not be applicable to other text creation tasks.<br>2. Different human evaluators may have varying interpretations and criteria for evaluating text, making the study's human review procedure open to bias. |
| g. Keep Meeting Summaries on Topic: Abstractive Multi-Modal Meeting Summarization (2019) | 1. In the study paper, a unique method for creating conference summaries that integrates both written and visual data from the meeting is suggested.<br>2. In terms of ROUGE scores, the proposed model performs better than earlier | 1. The task-oriented meeting is the emphasis of the research paper, which may make it difficult to generalise to other sorts of meetings.<br>2. The suggested model requires a lot of |

| | | |
|---|---|---|
| | approaches to meeting summary.<br>3. The produced summaries are more thorough and educational since various modalities (text and pictures) were used in the summary process. | processing and might not be useful for real-time summaries of significant sessions. |
| **2. Action Items Extraction** | | |
| a. Smart To-Do: Automatic Generation of To-Do Items from Emails (2022) | 1. Using a commitment classifier that recognises action intents in the emails, the task sentence is identified.<br>2. utilises supervised algorithms to identify significant phrases in the email that can be used to provide context for the question. | 1. Inadequate architectural and design elements must be fixed before using structured meta data.<br>2. A multitask generation model that can jointly identify useful context for the task can replace the current two stage framework. |
| b. Action item detection in meetings using pretrained transformers (2021) | 1. The model created a baseline for the action item detection from the AMI meeting corpus and significantly improved the state-of-the-art performance on the ICSI simulated meeting corpus.<br>2. Evaluated the issues with the action item | 1. Action item detection as a span boundary detection problem has been presented, however it needs further work and might not be appropriate in all circumstances.<br>2. For action item detection models, the problems of tiny, |

| | | |
|---|---|---|
| | detection task's binary classification formulation and found connections to span boundary tasks. | noisy, and highly unbalanced datasets still present difficulties.<br>3. Effective action item detection may need for longer context models, which can be computationally expensive and time-consuming. |
| c. Automatic Rephrasing of Transcripts-based Action Items (2021) | 1. The approach is to provide improved communication and collaboration by providing clear and concise language for action items, the system can help team members understand and complete tasks more efficiently.<br>2. The system can be trained on specific domains or vocabularies, allowing it to generate more accurate and relevant phrasings for specific industries or teams. | 1. The performance of the system may be constrained by the language it was trained on, and it may not perform as well with dialects or other languages.<br>2. For teams that don't regularly record or transcript their meetings, the system's requirement for transcripts may provide a problem. |
| d. Preparing a Dataset for Extracting Decision Elements from Meeting Transcript Corpus (2018) | 1. Identified elements in natural conversations and developed an associated annotation scheme. | 1. In the suggested annotation scheme, decision analysis components like expressions of |

| | | |
|---|---|---|
| | 2. The system can be trained on specific domains or vocabularies, allowing it to extract decision elements more accurately and relevantly for specific industries or teams. | constraints and trade-offs are not included.<br>2. The method is primarily designed to extract decision-making components from meeting minutes, therefore it might not be as effective for other kinds of documents or communication. |
| **3. Speaker Accent, Speaker Diarization & Speech to Text** | | |
| a. Robust Speech Recognition via Large-Scale Weak Supervision (2022) | 1. In the paper, a unique weak supervision voice recognition method is presented.<br>2. The method is scalable and supports huge training data sets.<br>3. The method is suited for use in real-world applications since it is resistant to noise and other types of interference.<br>4. On a number of benchmark datasets, the suggested technique produces state-of-the-art results.<br>5. | 1. It is uncertain how well the suggested strategy would perform on other datasets because it has only been tested on a small subset of benchmark datasets.<br>2. Weak supervision may add bias into the training data, which may reduce the model's accuracy.<br>3. The proposed solution needs a lot of computer power, which might make it impractical for some applications.<br>4. It is challenging to grasp how the model |

| | | |
|---|---|---|
| | | generates its predictions because the methodology is not easily comprehensible. |
| b. Linguistic-Acoustic Similarity Based Accent Shift for Accent Recognition (2022) | 1. The research provides an innovative method for accent identification based on linguistic-acoustic similarities.<br>2. When compared to conventional methods, the suggested method provides great accuracy in accent recognition.<br>3. The method is suited for use in real-world applications since it is resistant to noise and other types of interference.<br>4. The suggested methodology to accent recognition is adaptable and can be used with various languages. | 1. It is unclear how well the proposed technique would perform on other languages because it has only been tested on a small set of languages.<br>2. The suggested solution needs a lot of computer power and can be challenging to use in real-time applications.<br>3. It is challenging to grasp how the model generates its predictions because the methodology is not easily comprehensible.<br>4. The usage of linguistic-acoustic similarity might skew the training |

| | | |
|---|---|---|
| | | set of data, which would reduce the model's precision. |
| c. Pyannote.audio: neural building blocks for speaker diarization (2019) | 1. pyannote.audio provides end-to-end neural implementations for each building block required for speaker diarization, which can be combined and jointly optimized to build speaker diarization pipelines.<br>2. pyannote.audio comes with pre-trained models for voice activity detection, speaker change detection, overlapped speech detection, and speaker embedding, which have achieved state-of-the-art performance for most of these tasks.<br>3. pyannote.audio supports on-the-fly data augmentation, which is very convenient for training neural networks as it generates a virtually infinite number of augmented versions of each original audio file. | 1. The suggested framework necessitates significant processing resources, such as strong GPUs and lots of memory, which may prevent some users from adopting it.<br><br>2. The suggested framework is only tested on datasets in English; results obtained with datasets in other languages may differ. |

| | | |
|---|---|---|
| **4. Emotion Sentiment Analysis** | | |
| a) Sentiment Analysis and Emotion Recognition from Speech<br>Using Universal Speech Representations (2022) | 1. Sentiment analysis and emotion recognition from speech are made more effective and accurate with the usage of universal speech representations.<br>2. The method is more adaptable and simpler to use across many datasets and languages because it doesn't necessitate explicit feature engineering or domain-specific knowledge.<br>3. The suggested approach is thoroughly evaluated in this study on a number of benchmark datasets, confirming its superiority over other state-of-the-art techniques. | 1. To train the universal speech representations, the method depends on the availability of a lot of speech data, which might not always be possible in specific applications or domains.<br>2. It's possible that the paper won't apply to other languages or dialects because it concentrates on English language speaking.<br>3. As it relies on preset categories and might not take into account for individual differences in expression or perception, the proposed approach might not be able to capture subtle variations in mood or feelings. |
| **5. Topic Segmentation** | | |
| a. Unsupervised Topic | 1. The use of BERT embeddings allows for a | 1. It assumes a consistent topic structure, which |

| | | |
|---|---|---|
| Segmentation of Meetings with BERT Embeddings (2021) | more accurate representation of the semantic meaning of the text, leading to better segmentation results.<br>2. The unsupervised nature of the approach means that it does not require annotated data or prior knowledge of the topics being discussed, making it a more flexible and scalable method.<br>3. The paper presents a comprehensive evaluation of the proposed approach on several benchmark datasets, demonstrating its effectiveness in comparison to other state-of-the-art methods. | is not necessarily the case.<br>2. Only in English-only meetings tested.<br>3. Due to the complexity of the computations, real-time segmentation may not be appropriate. |
| **6. Automated Minutes of Meeting** | | |
| a. Automated Generation of Meeting Minutes Using Deep Learning Techniques (2022) | 1. The suggested approach lessens the workload of human participants by automating the generation of meeting minutes.<br>2. The suggested solution reduces errors by utilising deep learning techniques to guarantee the accuracy and completeness of the generated minutes. | 1. It is uncertain how well the suggested strategy would perform in meetings of various types given that it is tested on a particular kind.<br>2. The suggested approach is reliant on the accessibility of crystal-clear audio |

| | | |
|---|---|---|
| | 3. When opposed to manually taking notes and summarising, the automatic compilation of the meeting minutes saves time.<br>4. The suggested approach is adaptable to many meeting forms and types. | recordings of the meeting, which might be constrained in some circumstances.<br>3. The method is not easily comprehensible, making it challenging to comprehend how the model generates the minutes.<br>4. The automated creation of meeting minutes may give rise to unethical issues, such as the possibility of excluding specific participants or slanting the minutes in favour of particular viewpoints. |
| b. MoM: Minutes of Meeting Bot (2021) | 1. The meeting minutes-taking process is automated by the bot, which lessens the workload for the human attendees.<br>2. The algorithm of the bot ensures the precision and thoroughness of the minutes taken, minimising errors.<br>3. Compared to manually taking notes and | 1. It is uncertain how well the suggested strategy would perform in meetings of various types given that it is tested on a particular kind.<br>2. The suggested approach is reliant on the accessibility of crystal-clear audio recordings of the |

| | | |
|---|---|---|
| | summarising the meeting minutes, the bot's automatic transcription and summation of the minutes saves time.<br>4. The format and organisation of the meeting minutes are consistent thanks to the bot's algorithm. | meeting, which might be constrained in some circumstances.<br>3. The method is not particularly interpretable, making it challenging to comprehend how the bot chooses which portions of the conference to record and summarise.<br>4. The automation of taking minutes may give rise to ethical questions, such as the possibility of excluding some attendees or slanting the minutes in favour of particular viewpoints. |
| c. A Sliding-Window Approach to Automatic Creation of Meeting Minutes (2021) | 1. The suggested solution lessens the workload of human participants by automating the process of writing meeting minutes.<br>2. The suggested approach is adaptable and can accommodate various meeting configurations. | 1. It is uncertain how well the suggested strategy would perform in meetings of various types given that it is tested on a particular kind.<br>2. The suggested approach is reliant on the accessibility of |

| | | |
|---|---|---|
| | 3. The suggested approach can produce meeting minutes in real-time, making it possible to distribute the minutes right away after the meeting.<br>4. The suggested approach is flexible enough to accommodate various participants' accents and speaking styles. | crystal-clear audio recordings of the meeting, which might be constrained in some circumstances.<br>3. The method is not easily comprehensible, making it challenging to comprehend how the model generates the minutes.<br>4. The automated creation of meeting minutes may give rise to unethical issues, such as the possibility of excluding specific participants or slanting the minutes in favour of particular viewpoints. |

Table 2.4 Summary of Literature Survey

# Chapter 3

# Proposed System

## 3.1 Overview

An end-to-end web application software based on microservices architecture including client dashboard and online meeting attender bots. An interactive dashboard allows user to obtain meeting summary and other analytics related to the meeting in the following three scenarios:

1. Meeting is scheduled but not started
2. Meeting has already started
3. Meeting is over

The proposed solution addresses all three scenarios. The first two scenarios are handled by meet attender bots and the last scenario is handled by transcript of recording upload feature on the dashboard.

In the end, user is provided with meeting summary, action items list, keywords, analytics like no. of speakers, no. of question asked, no. of dates and times discussed. Finally, using all this information, an automated minutes of meeting document is generated which user can download, share or edit.

### 3.2 Existing Architecture

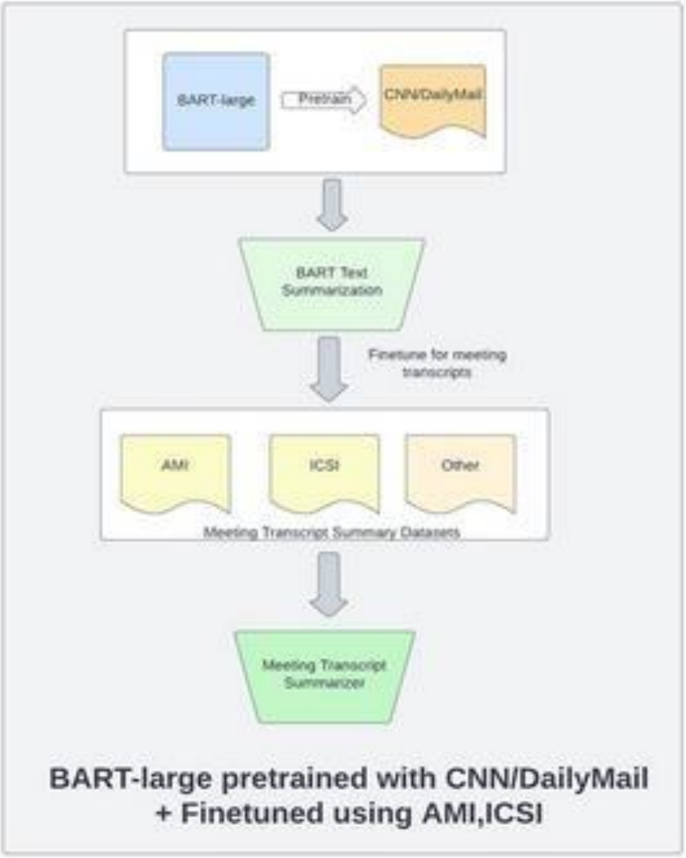


Fig. 3.2 (a) Existing Summarization Model

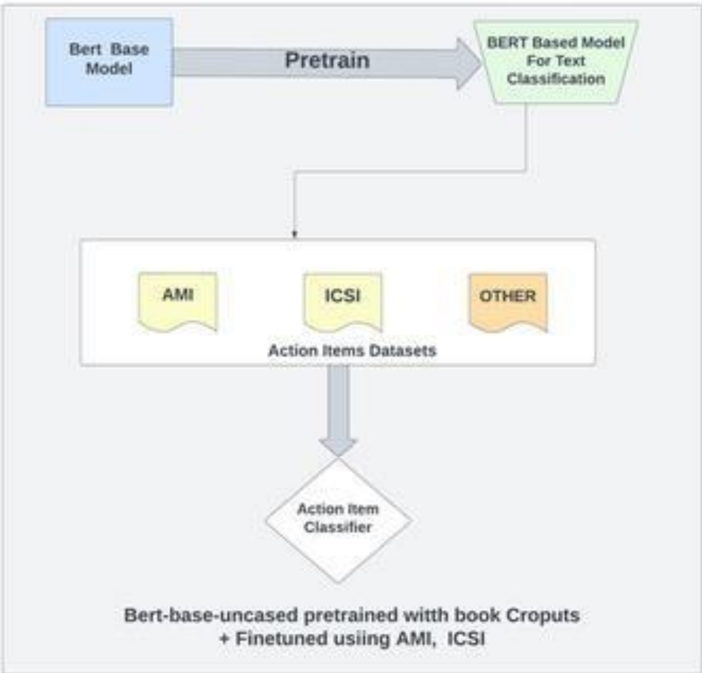


Fig. 3.2 (b) Existing Action Items Extraction Model

1. Audio Input: The system takes in the audio recording of the meeting as input.

2. Speech-to-Text Conversion: The audio recording is converted to text using a speech-to-text conversion tool such as Google Cloud Speech-to-Text, Amazon Transcribe or Microsoft Azure Speech Services.

3. Text Summarization: The text is then summarized using natural language processing (NLP) techniques such as extractive or abstractive summarization. Extractive summarization identifies the most important sentences in the text and combines them into a summary, while abstractive summarization generates a summary in a more human-like way, by rephrasing and paraphrasing the original text.

4. Sentiment Analysis: The system can analyze the sentiment of the text to identify the tone of the meeting and to understand the opinions and emotions expressed by the speakers. This information can be used to improve the accuracy of the summary and to identify action items.

5. Action Item Extraction: The system can identify action items discussed during the meeting by analyzing the text for certain keywords and phrases, or by using machine learning models to extract them.

6. Minutes of Meeting: The final output of the system is the minutes of meeting document, which includes the meeting summary, action items, attendee list and task owner names and task deadlines.

Overall, the existing architecture for automated meeting minutes and summaries involves several NLP techniques, such as speech-to-text conversion, summarization, sentiment analysis, and action item extraction, working together to produce the minutes of meeting document.

### 3.3 Proposed System Architecture

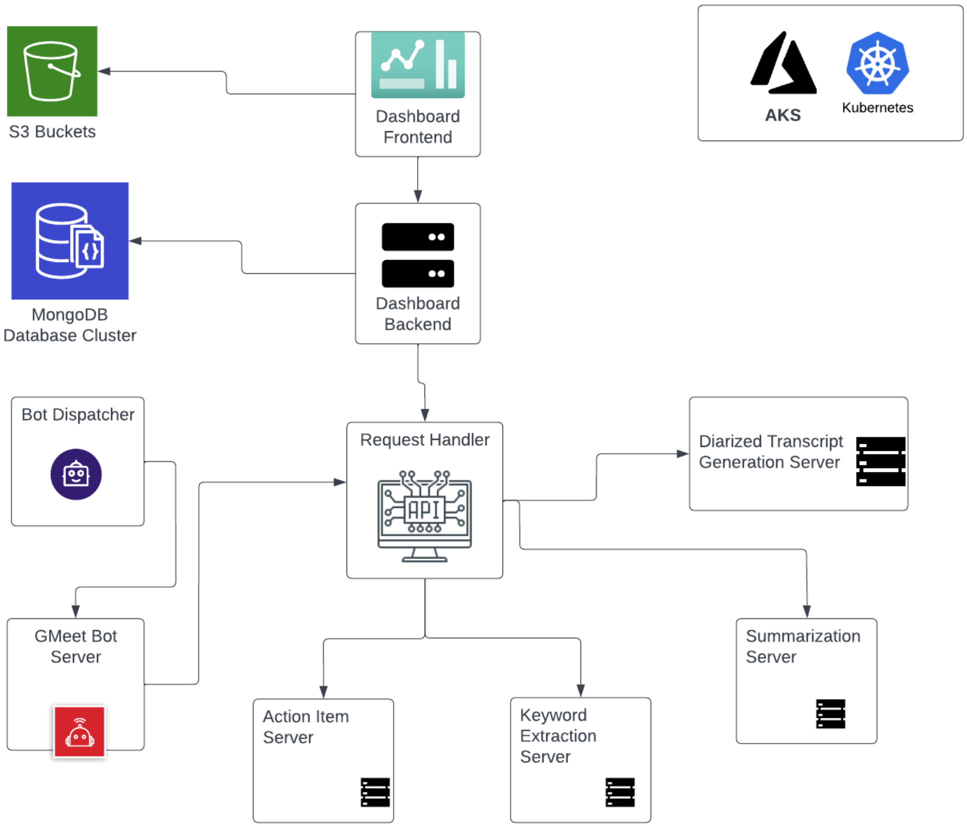


Fig. 3.3 Proposed System Architecture

#### 3.3.1 Implementation Details

The system is implemented on microservices architecture and deployed on Azure Kubernetes Service allowing it to rapidly scale as per demand. The components are as follows:

i. Dashboard Frontend (Load Balancer)
ii. Dashboard Backend
iii. Request Handler (API server)
iv. Bot Dispatcher
v. Bot Server
vi. Diarized Transcript Generation Server

vii. Summarization Server
viii. Action Items Server
ix. Keywords Extraction Server
x. MongoDB Database Cluster (Database)
xi. AWS S3 Buckets (Database)

### 3.3.2 Techniques and Algorithms

#### 3.3.2a Diarization

Diarization is the process of separating a multi-speaker audio recording into individual speaker segments. In our proposed system, we use a state-of-the-art diarization algorithm that analyzes the audio signal and identifies different speakers based on the characteristics of their voices. This diarization process enables the system to identify each speaker in the meeting and create separate audio segments for them.

#### 3.3.2b Audio Segmentation

After diarization, the system injects silence between audio segments to create separate audio files for each speaker. This is done to ensure that the audio segments for each speaker do not overlap, making it easier to transcribe them accurately. The duration of the silence can be adjusted depending on the length of the speaker segments and the level of noise in the recording.

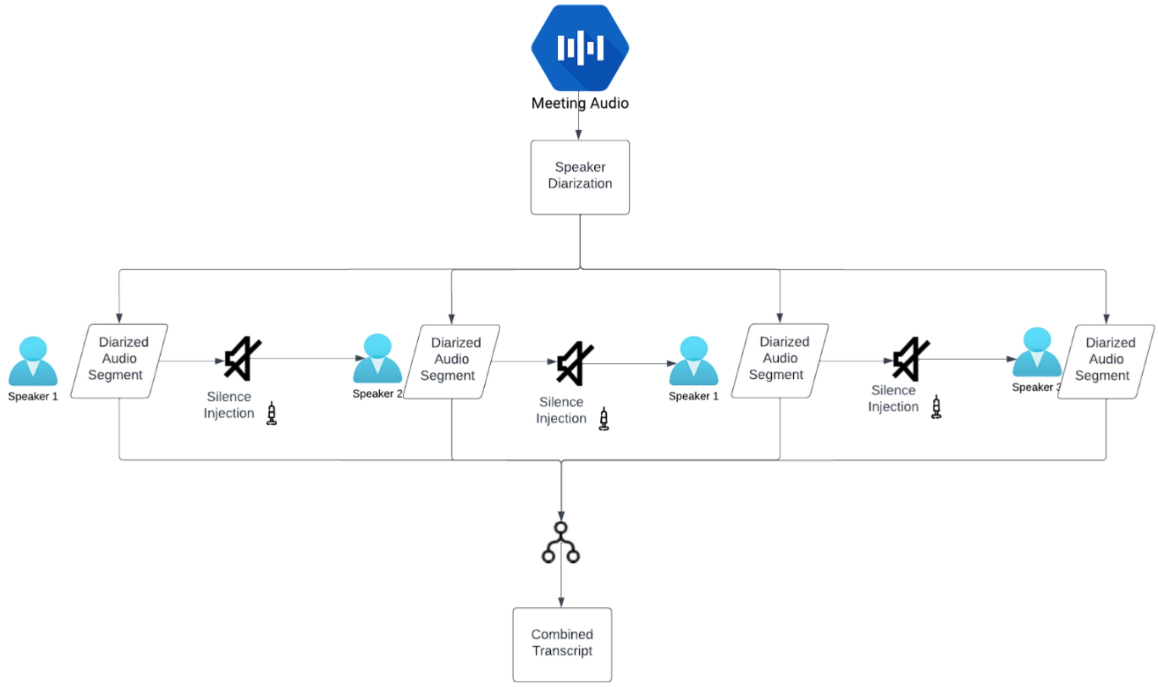


Fig. 3.3.2b Audio Segmentation

### 3.3.2c Transcript Generation

Once the audio segments have been created, the system combines them to generate the full transcript of the meeting. The system can use various speech-to-text algorithms to transcribe the audio segments into text, which can then be combined to create the full transcript. The accuracy of the transcription process can be improved by using state-of-the-art speech-to-text algorithms and by training them on meeting audio data.

### 3.3.2d Summarization

We divide meeting transcript into different transcript chunks. Using N-gram technique transcript chunks will be generated. These transcript chunks will be forwarded to Summarizer model in order to generate abstractive meeting summary. The summarization model will be build using hugging face BART base model for text summarization. In order to generate summary for dialogue-based transcript, base model will be fine-tuned using dialogue specific datasets like AMI & ICSI. Finally, we will get abstractive meeting summary for provided transcript.

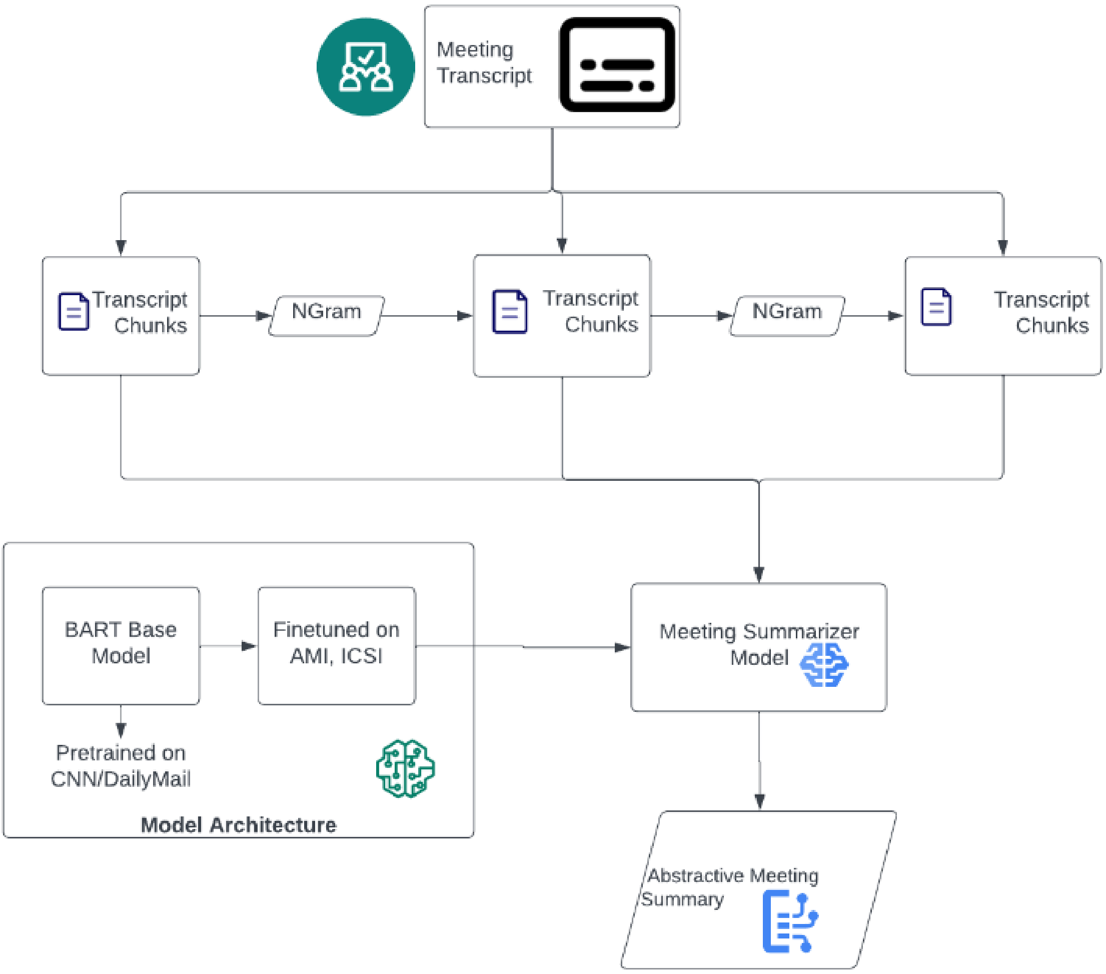


Fig. 3.3.2d. Summarization Model

### 3.3.2e Action Items

We fine-tune the DistilBERT model on the AMI and ICSI datasets, which contain meeting dialogue data with labeled action items. This fine-tuning process enables the model to learn the patterns and structures of meeting dialogues and action items, improving the accuracy of action item generation.

After fine-tuning, the DistilBERT model is used to create an action item generator. Given a meeting transcript, the generator identifies the action items discussed in the meeting and generates a list of action items

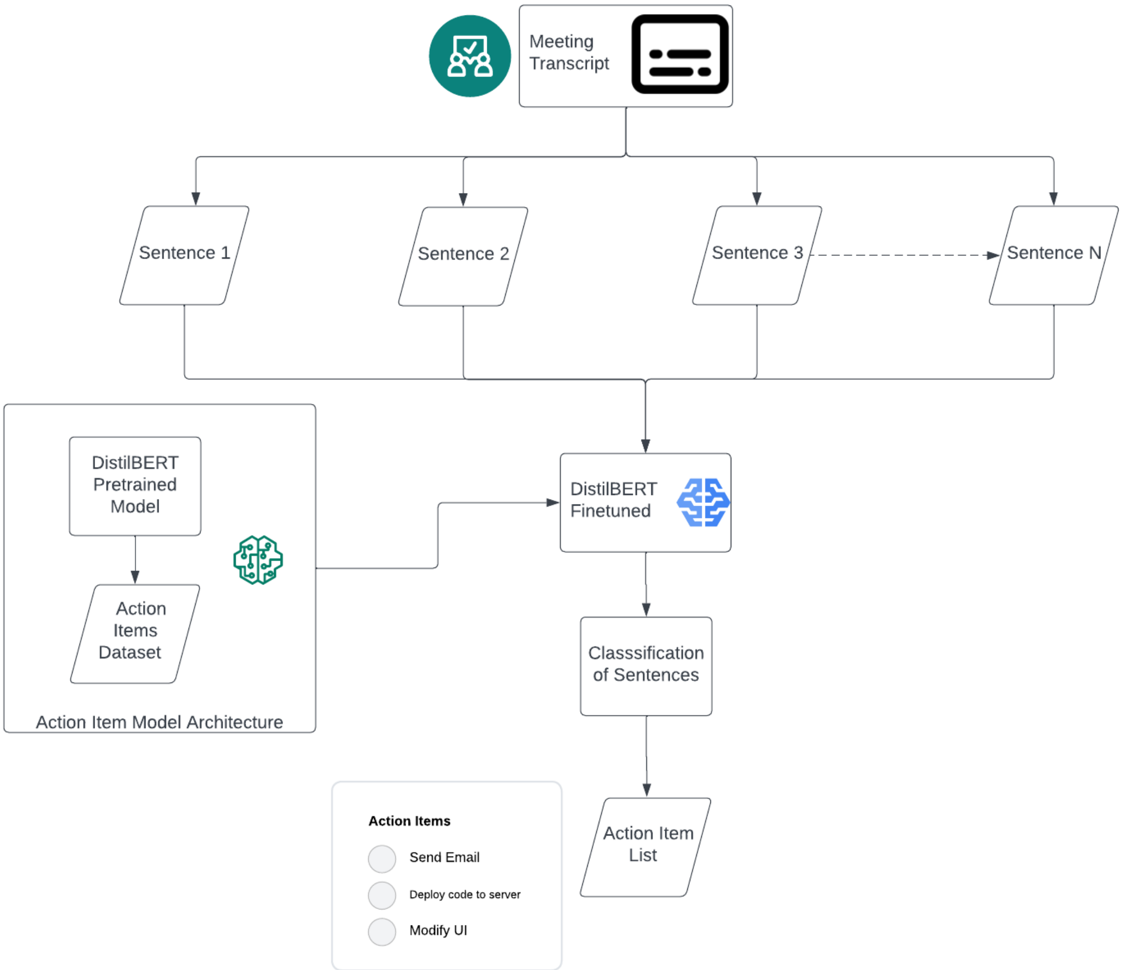


Fig. 3.3.2e Action Items Extraction Model

### 3.4 Sample Dataset

AMI Corpus, ICSI Corpus and SAMSUM datasets combined and shuffled for fine-tuning purpose.

| | transcripts | summaries |
|---|---|---|
| **0** | #Person B#: this is the kick-off meeting for o... | The project manager introduced the upcoming pr... |
| **1** | #Person B#: is that alright now? #Person B#: e... | The project manager briefed the team on some n... |
| **2** | #Person A#: 's to do now is to decide how to f... | The project manager recapped the decisions mad... |
| **3** | #Person B#: we all all set? #Person B#: this i... | The project manager recapped the decisions mad... |
| **4** | #Person B#: we're ready to begin. #Person B#: ... | The team members introduced themselves to each... |
| **5** | #Person C#: just put it on the deskt desktop. ... | The project manager recapped the events of the... |
| **6** | #Person B#: so we'll start off with quick over... | The project manager recapped the decisions mad... |
| **7** | #Person B#: that should hopefully do the trick... | The project manager recapped the decisions mad... |
| **8** | #Person B#: are we we're not allowed to dim th... | The Project Manager gave an introduction to th... |
| **9** | #Person B#: it's up there? #Person B#: that sc... | The Industrial Designer gave his presentation ... |

Table 3.4 Sample Dataset

```
DatasetDict({
    train: Dataset({
        features: ['summaries', 'transcripts'],
        num_rows: 13247
    })
    valid: Dataset({
        features: ['summaries', 'transcripts'],
        num_rows: 1656
    })
    test: Dataset({
        features: ['summaries', 'transcripts'],
        num_rows: 1656
    })
})
```

Fig. 3.4 Features of Dataset Dictionary

### 3.5 Hardware and Software Specifications

#### 3.5.1 Software Specifications

| | |
|---|---|
| Platforms | Azure or AWS (API Gateway), Kubernetes, Bot SDKs |
| Databases | MongoDB, S3 Buckets |
| Languages | Python, JavaScript |
| Frameworks | PyTorch, Tensorflow, ReactJS, NodeJS, Flask |

Table 3.5.1 Software Specifications

#### 3.5.2 Hardware Specification

Client Side:

| | |
|---|---|
| Operating System | Windows/ Mac OS/ Linux |
| RAM | Minimum 4 GB |
| Browser | Google Chrome, Mozilla Firefox, Safari |

Table 1.2.3 Client Hardware Specifications

Server Side:

| | |
|---|---|
| Operating System | Linux |
| GPU | At Least 4 GB VRAM |
| RAM | Minimum 16 GB |

Table 1.2.3 Server Hardware Specifications

# Chapter 4

# Applications

The project outlined above has several potential applications in various industries and organizations. Here are some examples:

### 4.1 Training and Education:

Abstractive summarization can be used in virtual training sessions to generate summaries of the material covered, which can be helpful for learners to quickly review the key concepts covered. Similarly, automated minutes of meeting can be used to generate records of virtual educational sessions for future reference.

### 4.2 Compliance:

In some industries, such as healthcare and finance, it is important to maintain detailed records of meetings to ensure compliance with regulations and industry standards. Automated minutes of meeting can help ensure that all relevant information is captured and documented.

### 4.3 Performance Management:

Abstractive summarization can be used to summarize performance review meetings, making it easier for managers and employees to understand feedback and track progress over time.

### 4.4 Knowledge Management:

Automated minutes of meeting can be used to create a repository of information and insights generated during virtual meetings, which can be helpful for knowledge sharing and retention.

### 4.5 Customer Service:

Abstractive summarization and automated minutes of meeting can be used to generate records of customer service interactions, which can be useful for tracking customer issues and improving service quality.

### 4.6 Research and Development:

Abstractive summarization and automated minutes of meeting can be used to capture and summarize research meetings, helping researchers keep track of ideas, insights, and progress over time.

### 4.7 Project Management:

Automated minutes of meeting can be used to generate project documentation, such as meeting minutes, action items, and progress reports. Abstractive summarization can help to quickly review key points and identify areas where additional attention is needed.

### 4.8 Sales and Marketing:

Abstractive summarization and automated minutes of meeting can be used to capture and summarize sales meetings and marketing discussions, helping teams to quickly understand customer needs, preferences, and feedback.

### 4.9 Human Resources:

Abstractive summarization and automated minutes of meeting can be used to capture and summarize HR meetings, such as performance reviews, employee onboarding, and training sessions. This can help to streamline HR processes and ensure that all relevant information is captured and documented.

### 4.10 Legal and Compliance:

Abstractive summarization and automated minutes of meeting can be used to capture and summarize legal and compliance meetings, such as board meetings, regulatory compliance reviews, and internal investigations. This can help to ensure that all legal and regulatory requirements are met, and can also serve as a record of decision-making processes.

# Chapter 5

# Summary

The project aims to automate the process of producing meeting minutes and meeting summary for professional virtual meetings using AI and deep learning techniques. The proposed system will convert audio from meeting recordings to text using a unique approach for speech-to-text that involves detecting the accent of the speaker and transcribing their audio using a model trained for their specific accent, improving the accuracy of the transcription. The transcript summarization task will be done by BART pre-trained model for abstractive summarization, fine-tuned using conversational text datasets like AMI and ICSI corpus. Sentiment analysis will be used to determine action items, tone of speaker and overall tone of the meeting. Live statistics about attendees and meeting chat will be sent by the meeting bot/extension. The summary, action items, and minutes of meeting document will be made available on the dashboard of the host. The objective of easing and automating the management of virtual meetings will be achieved with this project, making it easier and more efficient to manage virtual meetings and enabling better decisions in organizations that rely on virtual meetings.

# References


[1] M. Zhong, Y. Liu, Y. Xu, C. Zhu, and M. Zeng, “DialogLM: Pre-trained Model for Long Dialogue Understanding and Summarization,” arXiv:2109.02492 [cs], Jan. 2022, Accessed: Mar. 22, 2023. [Online]. Available: https://arxiv.org/abs/2109.02492 .

[2] Y. Zhang *et al.*, “Summ^N: A Multi-Stage Summarization Framework for Long Input Dialogues and Documents,” *arXiv:2110.10150 [cs]*, Apr. 2022, Accessed: Mar. 22, 2023. [Online]. Available: https://arxiv.org/abs/2110.10150

[3] Y. Zhang et al., “An Exploratory Study on Long Dialogue Summarization: What Works and What’s Next,” ACLWeb, Nov. 01, 2021. https://aclanthology.org/2021.findings-emnlp.377/ (accessed Mar. 22, 2023).

[4] L. Mehnaz et al., “GupShup: Summarizing Open-Domain Code-Switched Conversations,” ACLWeb, Nov. 01, 2021. https://aclanthology.org/2021.emnlp-main.499/ (accessed Mar. 22, 2023).

[5] Singhal, K. Khatter, T. A and J. R, "Abstractive Summarization of Meeting Conversations," 2020 IEEE International Conference for Innovation in Technology (INOCON), Bangluru, India, 2020, pp. 1-4, doi: 10.1109/INOCON50539.2020.9298305.

[6] J. Maynez, S. Narayan, B. Bohnet, and R. McDonald, “On Faithfulness and Factuality in Abstractive Summarization,” Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, 2020, doi: https://doi.org/10.18653/v1/2020.acl-main.173

[7] M. Li, L. Zhang, H. Ji, and R. J. Radke, “Keep Meeting Summaries on Topic: Abstractive Multi-Modal Meeting Summarization,” Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics, 2019, doi: https://doi.org/10.18653/v1/p19-1210.

[8] S. Mukherjee, S. Mukherjee, M. Hasegawa, A. Hassan Awadallah, and R. White, “Smart To-Do: Automatic Generation of To-Do Items from Emails,” ACLWeb, Jul. 01, 2020. https://aclanthology.org/2020.acl-main.767/ (accessed Mar. 22, 2023).

[9] K. Sachdeva, J. Maynez and O. Siohan, "Action Item Detection in Meetings Using Pretrained Transformers," 2021 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), Cartagena, Colombia, 2021, pp. 861-868, doi: 10.1109/ASRU51503.2021.9688167.

[10] Cohen, A. Kantor, S. Hilleli, and E. Kolman, “Automatic Rephrasing of Transcripts-based Action Items,” ACLWeb, Aug. 01, 2021. https://aclanthology.org/2021.findings-acl.253/ (accessed Mar. 22, 2023).

[11] T. Tran, F. Bonin, L. Deleris, D. Ganguly, and K. Levacher, “Preparing a Dataset for Extracting Decision Elements from a Meeting Transcript Corpus,” www.semanticscholar.org, 2018. https://www.semanticscholar.org/paper/Preparing-a-Dataset-for-Extracting-Decision-from-a-Tran-Bonin/ae2384bb9d9c965a31f3a4107f25c14c3bb3406a (accessed Mar. 22, 2023).

[12] YouRadford, J. W. Kim, T. Xu, G. Brockman, C. McLeavey, and I. Sutskever, “Robust Speech Recognition via Large-Scale Weak Supervision,” arXiv:2212.04356 [cs, eess], Dec. 2022, Available: https://arxiv.org/abs/2212.04356

[13] Q. Shao et al., “Linguistic-Acoustic Similarity Based Accent Shift for Accent Recognition,” arXiv:2204.03398 [cs, eess], Jul. 2022, Accessed: Mar. 22, 2023. [Online]. Available: https://arxiv.org/abs/2204.03398

[14] H. Bredin et al., “pyannote.audio: neural building blocks for speaker diarization,” arXiv:1911.01255 [cs, eess], Nov. 2019, Accessed: Mar. 22, 2023. [Online]. Available: https://arxiv.org/abs/1911.01255

[15] T. Atmaja and A. Sasou, “Sentiment Analysis and Emotion Recognition from Speech Using Universal Speech Representations,” Sensors, vol. 22, no. 17, p. 6369, Aug. 2022, doi: https://doi.org/10.3390/s22176369.

[16] Solbiati, K. Heffernan, G. Damaskinos, S. Poddar, S. Modi, and J. Cali, “Unsupervised Topic Segmentation of Meetings with BERT Embeddings,” arXiv:2106.12978 [cs], Jun. 2021, Accessed: Mar. 22, 2023. [Online]. Available: https://arxiv.org/abs/2106.12978

[17] X. Feng, X. Feng, and B. Qin, “A Survey on Dialogue Summarization: Recent Advances and New Frontiers,” arXiv:2107.03175 [cs], Apr. 2022, Available: https://arxiv.org/abs/2107.03175

[18] Zhu, R. Xu, M. Zeng, and X. Huang, “A Hierarchical Network for Abstractive Meeting Summarization with Cross-Domain Pretraining,” Findings of the Association for Computational Linguistics: EMNLP 2020, 2020, doi: https://doi.org/10.18653/v1/2020.findings-emnlp.19.

[19] M. Lewis et al., “BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension,” arXiv:1910.13461 [cs, stat], Oct. 2019, Available: https://arxiv.org/abs/1910.13461

[20] V. D, L. Vig, G. Shroff, and P. Agarwal, “MEETING BOT: Reinforcement Learning for Dialogue Based Meeting Scheduling,” arXiv:1812.11158 [cs], Dec. 2018, Accessed: Mar. 22, 2023. [Online]. Available: https://arxiv.org/abs/1812.11158

[21] T. Tran, F. Bonin, L. Deleris, D. Ganguly, and K. Levacher, “Preparing a Dataset for Extracting Decision Elements from a Meeting Transcript Corpus,” www.semanticscholar.org, 2018. https://www.semanticscholar.org/paper/Preparing-a-Dataset-for-Extracting-Decision-from-a-Tran-Bonin/ae2384bb9d9c965a31f3a4107f25c14c3bb3406a (accessed Mar. 22, 2023).

[22] S. Wang, X. Zhao, B. Li, B. Ge and D. Tang, "Integrating Extractive and Abstractive Models for Long Text Summarization," 2017 IEEE International Congress on Big Data (BigData Congress), Honolulu, HI, USA, 2017, pp. 305-312, doi: 10.1109/BigDataCongress.2017.46.

# Acknowledgement

We would like to express our special thanks to **Prof. K.S. Charumathi** and **Prof. Shubhangi Chavan,** our major project guides who guided us through the project and who helped us in applying the knowledge that we have acquired during the semester and learning new concepts.

We would like to express our special thanks to **Dr. Sharvari Govilkar,** the Head of the Department for Computer Engineering who gave us the opportunity to do this project because of which we learned new concepts and their applications.

Finally, we would like to express our special thanks of gratitude to Principal **Dr. Sandeep Joshi,** who gave us the opportunity and facilities to conduct this project.

Alashi Harsh Kishor

Sankpal Shreyas Shashikant

Sase Omkar Ramkrishna

Zade Chaitanya Pramod