A

Project Report

on

Automated Generation of Executive Summaries for Meetings using NLP

In partial Fulfillment of the Requirements

for the Award of the Degree of

BACHELOR OF ENGINEERING

IN

INFORMATION TECHNOLOGY

Submitted by

Banda Sujith Kumar	1601-21-737-101
Mohammed Arbaz	1601-21-737-119
Mohd Ramzan Shareef	1601-21-737-120

SUPERVISOR

Dr. B. Swathi Sowmya

Assistant Professor

DEPARTMENT OF INFORMATION TECHNOLOGY

CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY(A)

(Affiliated to Osmania University;Accredited by NBA,NAAC,ISO)

kokapet(V),GANDIPET(M),HYDERABAD-500075

Website: www.cbit.ac.in

2024-2025

CERTIFICATE

This is to certify that the project work entitled **Automated Generation of Executive Summaries for Meetings using NLP** is submitted by **Banda Sujith Kumar** & **1601-21-737-101**, **Mohd Ramzan Shareef** & **1601-21-737-120**, **Mohammed Arbaz Ahmed** & **1601-21-737-119** in partial fulfillment of the requirements for the award of the degree of **Bachelor of Engineering** in **Information Technology** to **CHAITANYA BHARATHI INSTITUTE OF TECHNOLOGY(A)** affiliated to **OSMANIA UNIVERSITY**, Hyderabad is a record of bonafide work carried out by them under my supervision and guidance. The results embodied in this report have not been submitted to any other University or Institute for the award of any other Degree or Diploma.

Dr. B. Swathi Sowmya
Assistant Professor

Dr. M Venu Gopalachari
Professor and Head, IT

Declaration

We declare that the project report entitled **"Automated Generation of Executive Summaries for Meetings using NLP"** is being submitted by us to the Department of Information Technology, Chaitanya Bharathi Institute of Technology (A), Osmania University. This is the record of bonafide work carried out by us under the guidance and supervision of **Dr. B. Swathi Sowmya**, Assistant Professor, Dept. of IT, C.B.I.T. No part of the work is copied from books, journals, or the internet, and wherever the portion is taken, the same has been duly referred to in the text. The reports are based on the project work done entirely by us and not copied from any other source.

Banda Sujith Kumar

Mohd Ramzan Shareef

Mohammed Arbaz

Acknowledgement

The satisfaction that accompanies the successful completion of the task would be put incomplete without the mention of the people who made it possible, whose constant guidance and encouragement crown all the efforts with success.

We wish to express our deep gratitude to **Dr. B. Swathi Sowmya**, Assistant Professor and Project Supervisor, Department of Information Technology, Chaitanya Bharathi Institute of Technology, for her able guidance and useful suggestions, which helped us in completing the project in time.

Additionally, we are deeply appreciative of the valuable insights and encouragement provided by **Dr. A. Sirisha**, Assistant Professor, **Dr. T. Satyanarayana Murthy**, Associate Professor and **Prof. Suresh Pabboju**, Professor whose guidance has been integral to the successful completion of this work.

We are particularly thankful to **Dr. M Venu Gopalachari**, the Head of the Department, Department of Information Technology, his guidance, intense support and encouragement, which helped us to mould our project into a successful one.

We show gratitude to our honorable Principal **Prof. C . V. Narasimhulu** , for providing all facilities and support.

We also thank all the staff members of Information Technology department for their valuable support and generous advice. Finally thanks to all our friends and family members for their continuous support and enthusiastic help.

Banda Sujith Kumar

Mohd Ramzan Shareef

Mohammed Arbaz

Abstract

The rapid shift to virtual meetings has created a pressing need for efficient, automated documentation of discussions, decisions, and action items. This project presents an NLP-driven meeting summarization tool that leverages advanced machine learning techniques to convert audio-to-text outputs into concise, actionable summaries. By combining extractive and abstractive summarization, the tool identifies and synthesizes essential points from meeting conversations, effectively capturing both factual content and contextual nuances. Additionally, the system employs Named Entity Recognition (NER) and dependency parsing to extract tasks, responsibilities, and deadlines from the summarized text, making the output directly actionable and reducing follow-up workload. The NLP models, trained on a variety of conversational datasets, ensure accurate representation of discussions while filtering out irrelevant information. This tool not only enhances productivity and information retention but also reduces cognitive load, enabling users to focus on meaningful engagement rather than manual documentation. Through the automation of meeting summaries, this system offers a scalable, intelligent solution for organizations navigating remote and hybrid work environments.

Table of Contents

List of Tables

List of Figures

Abbreviations

CBIT	Chaitanya Bharathi Institute of Technology
IT	Information Technology
LLM	Large Language Models
NLP	Natural Language Processing
ML	Machine Learning
DL	Deep Learning
NER	Named Entity Recognition
BERT	Bidirectional Encoder Representations Transformers
GPT	Generative Pretrained Transformer
T5	Text-to-Text Transfer Transformer
GDPR	General Data Protection Regulation
TF-IDF	Term Frequency - Inverse Document Frequency
GUI	Graphical User Interface
RAM	Random Access Memory
SSD	Solid State Drive

CHAPTER 1
Introduction

The rise of remote work and digital communication has transformed how businesses and organizations collaborate, with online meetings becoming a central feature of daily operations. Platforms like Zoom, Microsoft Teams, and Google Meet facilitate discussions, enable decision-making, and provide a space for critical information exchange among team members, regardless of location. However, the increased frequency of these meetings presents challenges, as participants often struggle to retain and access key points and outcomes. This information overload can hinder productivity, especially in fast-paced environments where timely and accurate documentation is essential.

Effective documentation of meetings is crucial for both individual and organizational efficiency, as it allows participants to reference important decisions, assigned tasks, and actionable items. Traditionally, note-taking has been the primary means of capturing meeting content. Yet, manual methods are prone to inconsistencies, human error, and inefficiency, particularly in lengthy or complex meetings. Reviewing full recordings or transcripts can be cumbersome and time-consuming, delaying access to crucial information and decision-making processes.

Recent advancements in Natural Language Processing (NLP) have paved the way for automated summarization tools that can address these challenges. NLP, which enables machines to understand and process human language, can efficiently analyze and distill essential information from vast amounts of meeting data. Modern techniques such as extractive and abstractive summarization models allow systems to generate summaries that highlight key topics, decisions, and responsibilities, offering stakeholders a clear and concise record of each meeting. Extractive summarization identifies the most relevant phrases, while abstractive models create coherent, contextually accurate summaries that capture the conversation's essence.

Despite these advances, existing solutions are often tailored to specific platforms or limited in their flexibility and accuracy, leaving a gap for a more adaptable and reliable summarization tool. To bridge this gap, this project proposes an automated system that generates executive summaries from online meetings by leveraging advanced NLP techniques. Designed to handle data from diverse platforms and improve over time, this system will produce summaries that are not only informative but also accessible and actionable for users.

By automating the summarization process, this approach can improve meeting efficiency, reduce the burden of manual note-taking, and enhance decision-making. This report outlines a comprehensive solution that leverages NLP to transform raw meeting data into executive summaries, providing a practical tool to support collaboration, accountability, and productivity in today's dynamic work environments.

1.1 Challenges in Capturing Meeting Summaries

Capturing accurate and actionable summaries of meetings is essential for maintaining productivity and accountability within organizations. However, the process is fraught with challenges, particularly as the frequency and length of virtual meetings increase. From information overload to the nuances of language, several obstacles make effective summarization both time-consuming and prone to error.

1. **Information Overload and Retention:** As virtual meetings increase, participants often face back-to-back discussions, leading to an overwhelming volume of information. This overload makes it challenging to retain all critical details, decisions, and action items, particularly in meetings that cover multiple topics or involve complex issues. The result is that participants may struggle to recall essential points from past discussions, impacting decision-making and task execution.

2. **Inconsistent Note-taking and Human Error:** Manual note-taking relies on individual attention, skill, and consistency, which can vary

significantly between participants. This variability leads to inconsistencies in the detail and accuracy of meeting notes, with key points often missed or misinterpreted, especially in fast-paced or technical discussions. Such inconsistencies can cause misunderstandings, delays, and a lack of accountability as critical decisions and tasks may go undocumented or be inaccurately recorded.

3. **Difficulty in Identifying Key Information:** Meetings tend to include both critical information and casual conversation, making it difficult to distinguish actionable items and key decisions. Important points may be scattered or buried within the discussion, especially in lengthy or informal conversations, requiring participants to sift through transcripts to find relevant details. Without a systematic approach to identifying key information, meeting records can become unfocused and less useful for future reference.

4. **Time Constraints for Post-Meeting Documentation:** In high-paced work environments, participants often lack the time to review and document meeting outcomes thoroughly. Lengthy transcripts and recordings require considerable time to process, which may lead to delayed or neglected documentation. This delay can disrupt continuity, as essential information is not readily available for follow-up meetings or related projects, ultimately impacting workflows and timely action on critical insights.

5. **Language and Contextual Complexity:** Many meetings involve specialized topics or technical jargon, making accurate documentation difficult, especially when the language is nuanced or context-dependent. Capturing the true intent and meaning requires a strong understanding of the subject, which is not always feasible with general note-taking methods. Additionally, in multilingual or culturally diverse teams, language barriers can further complicate note-taking, resulting in summaries that lack clarity or omit essential details.

1.2 Need for Automated Meeting Summarization

To address the challenges of manual meeting documentation, implementing an automated summarization solution is essential. This approach offers several key benefits:

1. **Efficient Capture of Key Discussion Points:** Automated summarization systems can process meeting transcripts and highlight essential topics, decisions, and action items with high accuracy. By eliminating the need for manual note-taking, these systems ensure that critical information is captured consistently, reducing the risk of missed or misunderstood points and improving the reliability of meeting records.

2. **Enhanced Productivity and Decision-Making:** With quick access to concise, relevant summaries, stakeholders can make informed decisions without the need to revisit lengthy recordings or transcripts. This streamlined access to key information enables faster decision-making and allows team members to focus on actionable insights, enhancing overall productivity.

3. **Real-Time Summarization and Adaptability:** Leveraging technologies such as Natural Language Processing (NLP) and Machine Learning (ML), automated systems can generate summaries in real time, allowing participants to access key points immediately after the meeting. The system's adaptability enables continuous improvements, as it learns from past meetings and user feedback to refine its accuracy and relevance.

4. **Improved Consistency and Accuracy:** Automated systems are less prone to the inconsistencies and errors inherent in manual note-taking. By standardizing the summarization process, they provide consistent, high-quality summaries that capture meeting content accurately, which is essential for accountability and follow-through on tasks.

5. **Scalability Across Platforms and Languages:** Automated meeting summarization tools can easily scale to support multiple languages, dialects, and meeting formats, making them ideal for global teams. This

capability ensures that organizations can maintain effective communication and documentation practices across diverse regions and platforms, enhancing collaboration in multilingual and multicultural environments.

6. **Reduction in Cognitive Load for Participants:** Automated summarization reduces the cognitive burden on participants by allowing them to focus fully on the discussion rather than on note-taking. This shift leads to higher engagement during meetings, as participants can actively contribute without worrying about missing details. By offloading the task of documentation, automated summaries also help reduce "meeting fatigue," enabling teams to retain productivity across multiple meetings.

In conclusion, by implementing automated summarization, organizations can streamline meeting workflows, improve access to essential information, and enhance overall efficiency in an increasingly digital work environment.

1.3 NLP for Meeting Summarization

Natural Language Processing (NLP) plays a crucial role in meeting summarization by enabling systems to analyze and distill large volumes of language data into concise, actionable summaries. Through NLP, automated summarization systems can efficiently capture key points, decisions, and tasks, making meeting content more accessible and manageable for users.

1. **Extractive and Abstractive Summarization:** NLP enables both extractive and abstractive summarization. Extractive techniques identify key sentences or phrases directly from the transcript, while abstractive methods create new sentences to summarize the meeting's main ideas. This combination ensures that summaries are both accurate and easy to read.

2. **Use of Transformer Models:** Advanced NLP models like BERT, GPT, and T5 enhance meeting summarization by understanding the context and semantics of language. These transformer models excel at identifying

core topics, decisions, and action items, generating summaries that are more natural and coherent.

3. **Named Entity Recognition (NER) and Part-of-Speech (POS) Tagging:** NER and POS tagging are essential NLP techniques that identify names, dates, actions, and roles in meeting transcripts. This helps ensure that summaries include specific, actionable details, such as assigned tasks and responsibilities.

4. **Contextual Understanding and Topic Segmentation:** NLP systems can segment transcripts into distinct topics or themes, helping to organize and prioritize information. This allows summaries to reflect the structure of the conversation, making it easier for users to find relevant information.

1.4 Justification for Using NLP and Transformer Models

Combining transformer-based models (like BERT and T5) with extractive summarization techniques provides a comprehensive approach for generating accurate, concise summaries from meeting transcripts. Each approach has unique strengths, making them ideal for effective summarization.

1. **Contextual Understanding and Sentiment Analysis with Transformer Models:** Transformer models, such as BERT and T5, are well-suited for abstractive summarization, which involves generating new sentences that capture the overall meaning of discussions. Leveraging bidirectional attention mechanisms, these models provide deep contextual understanding, capturing nuances and sentiments essential to interpreting decisions and opinions accurately in meeting conversations.

2. **Handling Complex Language Structures:** Meetings often contain technical language and dense discussions. Transformer models can simplify these structures, distilling complex language into coherent, concise summaries without losing essential meaning. This capability is especially valuable in summarizing detailed or multi-faceted topics.

3. **High Accuracy in Extracting Key Information with Extractive Summarization Models:** Extractive summarization methods, such as TF-IDF and TextRank, focus on selecting the most relevant phrases directly from the original transcript. This ensures that critical points and specific terminology are preserved, which is particularly useful in meetings that require accuracy in specific details.

4. **Integration of Extractive and Abstractive Summarization:** Combining extractive and abstractive methods allows for a well-rounded summary, preserving key points (through extraction) and refining content for readability (through abstraction). This integration makes summaries both informative and coherent, presenting key details in an accessible way.

5. **Filtering Out Non-Relevant Content:** Both transformer models and extractive techniques aid in filtering out irrelevant or repetitive content. Extractive models prioritize high-value sentences, while transformers use attention mechanisms to focus on essential points. This leads to concise summaries that highlight only the core information.

6. **Scalability and Adaptability to Evolving Language:** Transformer models and extractive techniques are adaptable, allowing for fine-tuning on specific datasets to accommodate changes in language and industry terminology. This flexibility ensures that the summarization system remains relevant across various topics and meeting styles.

By leveraging the strengths of both transformer-based abstractive models and extractive methods, this approach ensures meeting summaries are clear, accurate, and contextually relevant, providing a robust solution for modern business needs.

1.5 Current Limitations and Gaps

Existing meeting summarization tools have made strides in automating documentation, but they still face several limitations that restrict their effec-

tiveness and usability across different environments.

1. **Platform-Specificity and Lack of Flexibility:** Many current summarization tools are tied to specific platforms (such as Zoom or Microsoft Teams), making them inaccessible or incompatible with other meeting services. This platform specificity limits their usability for organizations that use multiple meeting tools, requiring users to switch between summarization solutions or resort to manual note-taking for unsupported platforms.

2. **Limited Contextual Understanding:** Most existing tools rely heavily on keyword-based extraction or simple text processing, which often results in summaries that lack the depth and nuance needed for accurate representation of meeting content. Without advanced NLP models, these tools may fail to capture critical information, especially in complex discussions with context-dependent language or technical terminology. This limitation often leads to summaries that miss important insights or fail to convey the true intent of discussions.

3. **Scalability Issues for Large and Frequent Meetings:** Some tools are not designed to handle high volumes of meetings or lengthy transcripts, which can lead to processing delays and lower accuracy as the system struggles to manage large datasets. This lack of scalability is a significant drawback for organizations with frequent or extended meetings that generate substantial amounts of data needing quick, accurate summarization.

4. **The Need for a Platform-Independent, Scalable Solution:** To address these limitations, there is a clear need for a scalable, platform-independent solution, such as a Chrome extension, that can integrate seamlessly with various meeting platforms. A browser-based extension can provide flexibility by capturing and summarizing meeting content directly from the browser, regardless of the meeting platform used. This approach would allow organizations to standardize their summarization

processes across all platforms, enhancing efficiency and consistency in documentation.

By addressing these gaps, a robust, platform-independent meeting summarization tool could meet the needs of modern organizations, supporting diverse languages, capturing complex discussion context, and scaling with usage demands, ultimately providing a more versatile and reliable solution.

1.6 Ethical and Privacy Considerations

The development of an automated meeting summarization tool requires a strong commitment to data privacy and security, as meeting summaries often contain sensitive information such as business strategies, proprietary discussions, and personal details. Protecting this data involves implementing encryption protocols and secure storage practices to prevent unauthorized access or data breaches. To align with data privacy regulations like GDPR, the system should also include mechanisms to limit data retention, allowing users to delete or anonymize data as necessary. Ensuring that users feel confident in the privacy and security of their information is essential for fostering trust in the summarization tool.

Transparency and informed consent are equally crucial. Users should be made aware of how their data is processed, stored, and shared, along with any third-party access if applicable. By offering clear, accessible information about data policies and implementing an opt-in system, the tool can provide users with control over their data. Consent-based data processing not only respects user autonomy but also builds transparency into the system, ensuring that users are fully informed about how the tool impacts their privacy.

Lastly, addressing potential biases in NLP models is important to ensure that summaries are fair and representative of the full scope of a meeting. NLP models can inadvertently reflect biases present in the training data, which could lead to certain voices or topics being underrepresented or overemphasized. Regular evaluation and refinement of the summarization system, particularly with a focus on fair representation across diverse types of content, will help

to create an ethical and unbiased tool. This approach minimizes the risk of skewed summaries and ensures the system provides an accurate, balanced reflection of meeting discussions.

1.7 Problem Statement

In today's remote and hybrid work environments, online meetings have become essential for collaboration, decision-making, and information sharing. However, the high volume and frequency of these meetings often lead to information overload, making it challenging for participants to capture and retain key points, decisions, and action items. Traditional methods of manual note-taking are inconsistent, time-consuming, and prone to human error, often resulting in incomplete or inaccurate records. Reviewing lengthy meeting recordings or transcripts is also inefficient, particularly when quick decision-making is required.

Existing meeting summarization tools are limited by platform specificity, lack of flexibility, and inadequate contextual understanding, making them impractical for organizations that rely on diverse platforms and need summaries that accurately reflect nuanced discussions. Furthermore, many current solutions fail to address privacy concerns and ethical implications, especially when handling sensitive or proprietary information.

To address these issues, there is a need for an automated, platform-independent meeting summarization tool, such as a Chrome extension, that can seamlessly capture, process, and summarize discussions across various online meeting platforms. This solution should leverage advanced NLP and transformer models for accurate, context-aware summaries, while also ensuring data privacy and ethical handling of user information.

1.8 Motivation

The motivation for developing an automated meeting summarization tool stems from the growing demands and challenges faced in remote work, along with technological opportunities that can enhance efficiency and productivity

in the modern workplace. Key motivational factors include:

1. **Increasing Need for Efficient Documentation in Remote Work Environments:** As remote and hybrid work models have become mainstream, online meetings have become more frequent, involving multiple teams across various time zones. Efficient documentation of these meetings is crucial for maintaining clarity, alignment, and accountability. An automated summarization tool would provide a reliable way to capture essential information, reducing the time and effort required for manual note-taking.

2. **Addressing Information Overload and Cognitive Fatigue:** With back-to-back meetings becoming common, employees often experience information overload, making it challenging to retain key details and decisions from each discussion. An automated summarizer that distills large amounts of information into concise, actionable summaries would alleviate this cognitive load, enabling employees to stay focused and productive without feeling overwhelmed by excessive information.

3. **Time-Saving Solution for Quick Access to Meeting Insights:** Reviewing lengthy meeting recordings or transcripts is not only time-intensive but also impractical for fast-paced work environments where quick decisions are required. An automated summarization tool would enable participants to access key insights almost immediately after the meeting, allowing for faster decision-making and more agile project management.

4. **Advancements in NLP and Transformer Models for Accurate Summarization:** The development of powerful NLP and transformer models, such as BERT and T5, presents an opportunity to create an intelligent summarization system capable of understanding context, sentiment, and complex language structures. These models can generate summaries that accurately reflect the essence of discussions, capturing the nuances and key points without losing important details. Leveraging

these advancements in NLP motivates the creation of a tool that can deliver high-quality, contextually aware summaries.

5. **Ethical and Privacy-Conscious Summarization:** Given the sensitivity of information shared in many meetings, there is a need for a solution that respects privacy and ethical guidelines, ensuring secure handling of meeting data. Developing a tool with privacy and transparency features (such as data encryption and informed consent) meets the growing demand for ethical technology solutions in workplaces, fostering trust and compliance with data protection standards.

This project aims to bridge the gaps in current summarization solutions, motivated by the need for a tool that is flexible, efficient, accurate, and ethical, offering users a seamless way to manage meeting information and boost productivity in dynamic work environments.

1.9 Objectives

The primary objectives of the automated meeting summarization project are:

1. **To Develop a Platform-Independent Summarization Tool:** Create a Chrome extension that can operate across various online meeting platforms, ensuring compatibility and usability for users regardless of the specific tool (e.g., Zoom, Microsoft Teams, Google Meet) used to conduct meetings. This platform independence will offer flexibility for organizations that rely on multiple communication tools.

2. **To Automate the Summarization Process Using Advanced NLP Techniques:** Leverage Natural Language Processing (NLP) and transformer models (such as BERT and T5) to generate concise and coherent summaries that accurately capture the key points, decisions, and action items from meeting transcripts. The tool should combine extractive and abstractive methods to produce both informative and readable summaries.

3. **To Provide Accurate and Context-Aware Summaries:** Ensure that the summarization system is capable of understanding complex language structures, context, and sentiment to capture the essence of discussions. The tool should identify critical information, prioritize relevant content, and exclude non-essential or off-topic information, thus providing a clear and focused summary.

4. **To Enhance Meeting Efficiency and Reduce Cognitive Load:** By automating the summarization process, the tool should reduce the burden of manual note-taking and allow participants to focus more fully on the discussion. The tool aims to provide quick access to essential meeting insights, supporting faster decision-making and streamlined project management.

5. **To Ensure Data Privacy and Ethical Handling of Information:** Incorporate privacy protection measures, including data encryption, secure storage, and options for data deletion, to ensure the confidentiality and security of meeting content. Additionally, the tool will prioritize transparency, providing clear information to users about data processing and adhering to ethical standards for data handling.

6. **To Create a Scalable Solution for Frequent and Lengthy Meetings:** Design the tool to handle large volumes of meeting data efficiently, making it suitable for organizations with frequent or extended meetings. The solution should be scalable to process long transcripts quickly and accurately, enabling consistent performance across diverse meeting scenarios.

By achieving these objectives, the project aims to create a reliable, user-friendly meeting summarization tool that improves documentation, enhances productivity, and respects ethical and privacy considerations in a modern digital workplace.

CHAPTER 2
Literature Survey

Yan and Zhou propose a hybrid summarization method combining extractive and abstractive approaches, using K-means clustering and Seq2Seq modeling to improve coherence in summaries, particularly for structured texts like news articles [1]. Bharti et al. develop an automated system for generating audio/-text summaries from webinars and meetings, enabling real-time transcription and summarization to aid comprehension across corporate and educational contexts [2]. Muppidi et al. present a model for automatic meeting minutes generation, employing BART and T5 transformers to extract key points, summarize discussions, and highlight action items in corporate and collaborative settings [3]. Asmitha et al. explore automated summarization using Hugging Face NLP models, evaluating both extractive and abstractive methods with BERT and GPT, demonstrating their efficacy across diverse text types such as news and social media [4]. Patil et al. introduce a Chrome extension for real-time speech-to-text and summarization during online meetings, leveraging Microsoft Azure's Speech SDK to enhance productivity by providing accessible, in-meeting summaries [5].

Chaurasia et al. present an enhancement in text summarization through parallelization using the TF-IDF algorithm, significantly improving processing speed and scalability for large datasets, thus aiding efficient information extraction [8]. Majeed and Kala conduct a comparative study on extractive summarization, utilizing sentence ranking and TextRank algorithms. Their findings suggest that TextRank is particularly effective in generating high-quality, concise summaries [7]. Islam et al. analyze different text summarization techniques using enhanced tokenization, applying methods like cosine similarity and TextRank specifically to Bangla text, demonstrating effectiveness across diverse datasets [6]. Kachhoria et al. propose an automated solution for gen-

erating meeting minutes from online meetings using NLP and ML techniques, incorporating models such as PEGASUS and BART, along with speaker diarization for accurate meeting documentation [11]. Ganesh et al. introduce a novel text rank algorithm focused on effective extractive summarization by measuring sentence similarity, which yields relevant and informative summaries from large text documents [10].

Mahadevan et al. introduce a hybrid text summarizer for online meetings that combines extractive and abstractive approaches using centroid-based extraction and BART for generating cohesive summaries, enhancing the accessibility of meeting content for users [24]. Vadlamudi et al. propose a meeting summarizer using NLP techniques, employing TF-IDF and PageRank algorithms to create concise summaries from transcripts, making lengthy meeting discussions more manageable [22]. Yao et al. develop the Adaptive-BERT network for advertising text generation, which tailors advertising content to fit specific layout requirements in travel magazines by leveraging an enhanced BERT model with a self-attention layer for high-quality text generation [20]. Singhal et al. focus on abstractive summarization of dialogue-based meeting conversations using transformer models, showcasing how abstractive methods can produce more human-like summaries for dialogue-heavy content [19]. Zade et al. present a comprehensive analysis of NLP-based automated summarization and translation, creating a GUI application that integrates extractive summarization techniques and image processing for secure document management [12]. Bhat et al. propose a hybrid approach for summarizing formal online meetings, using a combination of extractive and abstractive methods, including BERT and TF-IDF, to deliver concise, human-readable summaries [23].

Reference	Technique	Observation
[7]	Comparative Study on Extractive Summarization	TextRank algorithm proves most effective for generating concise summaries, comparing multiple extractive summarization methods and their impact on summary quality.
[2]	Automated Summarization for Webinars	Real-time transcription and summarization aids comprehension in corporate settings, focusing on automatic summarization during live webinars and meetings for better accessibility.
[3]	Meeting Minutes Generation	Automates extraction of key points, action items from meeting discussions, reducing the need for manual minutes taking in collaborative corporate environments.
[9]	Hybrid Summarizer for Online Meetings	Combines extractive and abstractive methods for cohesive meeting summaries, enhancing accessibility and user-friendliness by combining different summarization techniques.
[5]	Real-time Speech-to-Text Summarization	Provides in-meeting summaries using speech-to-text for better productivity, with a focus on real-time summarization during virtual meetings using advanced speech recognition tools.
[6]	TF-IDF Parallelization for Summarization	Improves processing speed and scalability for large datasets, applying parallelization techniques to speed up the extraction of relevant content from large corpora.

Table 2.1: Summary of Prominent Works in Text Summarization

CHAPTER 3
System Requirements

3.1 Hardware Requirements

1. **Computing Device:** A computing device capable of running machine learning models is necessary. This can include:

   a) Desktop computer or workstation

   b) Laptop computer

   c) Cloud-based virtual machines (e.g., AWS, Google Cloud Platform, Microsoft Azure)

2. **Processor (CPU/GPU):** The spam detection model can be trained and executed on both CPUs and GPUs. For faster training times and inference speeds, a GPU with CUDA support is recommended. However, training on a CPU is also feasible, albeit potentially slower.

3. **Memory (RAM):** Sufficient RAM is required to accommodate the dataset, model parameters, and intermediate computations during training and inference. A minimum of 16 GB of RAM is recommended, with 32 GB or more preferred for larger datasets and models.

4. **Storage:** Adequate storage space is necessary for storing the dataset, trained model weights, and any intermediate results or logs generated during the training process. A minimum of 256 GB SSD is recommended for faster read/write speeds, along with additional storage for datasets, which may exceed several GBs.

5. **Network Requirements:** Reliable internet access is essential for cloud computing, data retrieval, and model updates. A high-speed internet connection is preferable to support efficient data transfer and real-time processing.

3.2 Software Requirements

1. **Operating System:** The spam detection model can be implemented on various operating systems, including:

   a) Windows

   b) macOS

   c) Linux distributions (e.g., Ubuntu, CentOS)

2. **Python:** The implementation of the spam detection model is typically done using the Python programming language. Ensure that Python is installed on the system, preferably with a package manager like Anaconda, which simplifies package management and environment setup.

3. **Python Libraries:** Several Python libraries are required for implementing and training the spam detection model, including:

   a) **Pandas**: For data manipulation and preprocessing

   b) **NumPy**: For numerical computations and array operations

   c) **Scikit-learn**: For machine learning algorithms and evaluation metrics

   d) **TensorFlow or PyTorch**: For building and training deep learning models (LSTM)

   e) **Keras**: High-level deep learning API that can use TensorFlow or PyTorch as a backend

   f) **Matplotlib or Seaborn:** For data visualization and plotting

   g) **NLTK or SpaCy**: For natural language processing tasks, such as tokenization and text analysis

4. **Development Environment:** An integrated development environment (IDE) or text editor is needed for writing and executing Python code. Popular options include:

   a) **Jupyter Notebook or JupyterLab**: Interactive notebooks for data exploration and experimentation

b) **Visual Studio Code (VS Code)**: Lightweight and versatile IDE with Python support

c) **PyCharm**: Full-featured IDE specifically designed for Python development

5. **JavaScript Libraries and Frameworks:** Several libraries and frameworks are necessary for building and managing both the frontend and backend:

   a) **React.js**: For building the user interface of the Chrome extension, providing a responsive and dynamic user experience.

   b) **Express.js**: A minimal Node.js framework for setting up the backend server, handling API routes, and facilitating communication between the extension and the backend.

   c) **MongoDB and Mongoose**: MongoDB, typically hosted via MongoDB Atlas, will be used as the primary database for storing meeting data and summaries. Mongoose serves as an Object Data Modeling (ODM) library for seamless integration with MongoDB.

6. **Optional Software:** Depending on the specific requirements of the project, additional software may be necessary. This can include:

   a) Virtualization software (e.g., Docker) for creating isolated development environments.

   b) Version control systems (e.g., Git) for collaborative development and code management.

   c) Communication tools (e.g., Slack, Microsoft Teams) for team collaboration and project updates

3.2.1 Google Colab

Google Colab, short for Google Colaboratory, is a cloud-based development environment provided by Google. It allows users to write and execute Python code in a browser-based interactive notebook interface, similar to Jupyter

Notebook. Google Colab offers several advantages for developing and running machine learning models, including:

1. **Free Cloud Computing:** Google Colab provides free access to high-performance computing resources, including CPU, GPU, and TPU instances. Users can execute computationally intensive tasks, such as training deep learning models, without the need for expensive hardware or infrastructure.

2. **Pre-installed Libraries:** Google Colab comes pre-installed with popular Python libraries, including NumPy, Pandas, Matplotlib, TensorFlow, and PyTorch. Users can easily import and utilize these libraries within their notebooks without the need for manual installation.

3. **Collaboration Features:** Google Colab supports real-time collaboration, allowing multiple users to work on the same notebook simultaneously. Users can share their notebooks with collaborators via a shareable link and collaborate on code development, data analysis, and model training in real-time.

4. **Integration with Google Drive:** Google Colab seamlessly integrates with Google Drive, enabling users to save and access their notebooks and datasets directly from their Google Drive accounts. This integration simplifies data management and facilitates sharing and collaboration on projects.

5. **Hardware Acceleration:** Google Colab provides access to GPU and TPU accelerators, which can significantly speed up model training and inference for deep learning tasks. Users can select the desired hardware accelerator type from the runtime settings menu and leverage the computing power of Google's infrastructure.

6. **Interactive Data Visualization:** Google Colab supports interactive data visualization using Matplotlib, Plotly, and other plotting libraries. Users can create interactive charts, graphs, and dashboards directly within

their notebooks to visualize model performance, explore datasets, and communicate results effectively.

Overall, Google Colab offers a convenient and cost-effective environment for developing, prototyping, and deploying machine learning models. Its integration with Google Drive, collaborative features, and access to powerful computing resources make it an attractive choice for data scientists, researchers, and machine learning practitioners looking to leverage cloud-based infrastructure for their projects.

By ensuring compatibility with the hardware and software requirements outlined above, you can effectively implement and deploy the spam call detection model, enhancing user protection against unsolicited calls and contributing to a safer communication environment.

CHAPTER 4
Methodology

4.1 Architecture

The architecture of the automated meeting summarization system is designed to capture, process, and distill key information from meeting conversations efficiently. The system integrates multiple NLP techniques and models to deliver both extractive and abstractive summaries, along with identified action items.

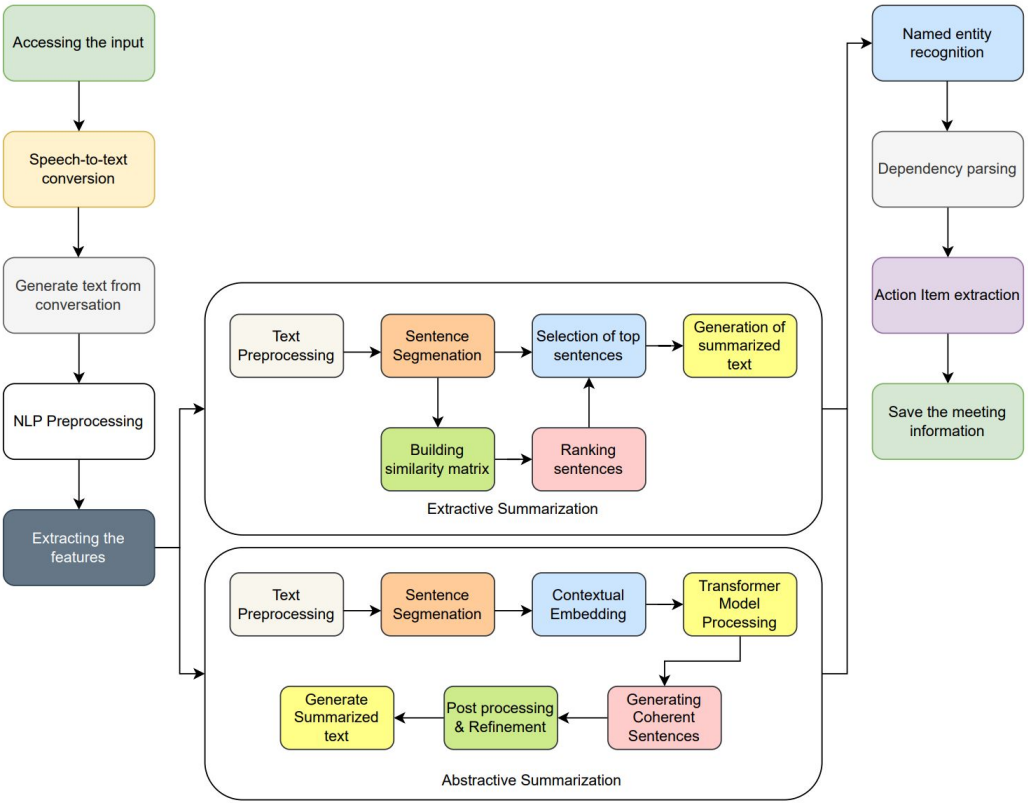

Figure 4.1: Flow of work

1. **Input Acquisition and Speech-to-Text Conversion:** The process starts by accessing the meeting input (audio or video), which is con-

verted into text through *speech-to-text conversion*. This step leverages Automatic Speech Recognition (ASR) models to ensure accurate transcription, forming the basis for text-based processing.

2. **NLP Preprocessing and Feature Extraction:** The transcribed text undergoes *NLP preprocessing*, including tokenization, stopword removal, and normalization, to clean and prepare it for analysis. The *feature extraction* step then identifies key linguistic features (e.g., named entities, parts of speech) that aid in later summarization and action item identification.

3. **Summarization Module:** The system uses two parallel paths for summarization: *Extractive Summarization* and *Abstractive Summarization*.

   (a) *Extractive Summarization:* This method selects important sentences from the text. After preprocessing and sentence segmentation, a *similarity matrix* is created to compare and rank sentences based on relevance. The top-ranked sentences are then used to form a concise extractive summary.

   (b) *Abstractive Summarization:* Unlike extractive summarization, this approach generates new sentences that represent the core ideas. Using *contextual embeddings*, the text is processed by a *transformer model* (e.g., BERT, T5), which creates a coherent paraphrased summary. A *post-processing* step refines the text for readability and accuracy.

4. **Action Item Identification:** This module focuses on extracting actionable items, such as tasks and assignments. *Named Entity Recognition (NER)* is used to detect key entities, and *dependency parsing* captures relationships between entities and actions, allowing for the generation of a list of tasks with responsible parties and deadlines.

5. **Data Storage and Output:** The summarized text and action items are presented to the user, with an integrated *feedback loop* for continuous

improvement. All meeting data, summaries, and action items are stored in a database, allowing users to access past meeting records as needed.

4.2 Dataset

The dataset used in this system consists of meeting transcripts and related metadata, which are processed and summarized to extract key insights and action items. The data plays a critical role in training the Natural Language Processing (NLP) models to generate accurate and contextually relevant summaries. The primary fields of the dataset include the meeting transcript, call durations, string lengths, and unique identifiers for each entry. These data points are used by the system to extract the essential elements of each meeting and generate summaries.

summary string · lengths 75 1.1k	uid string · lengths 27 37	id int64 0 5.17k	transcript string · lengths 541 376k
AS AMENDED a bill for an ordinance amending the Denver Zoning Code to revise parking exemptions for…	DenverCityCouncil_02272017_17-0161	2	I Please close the voting. Announce the results. Nine Ice nine Ice Council Bill 153 has been ordered…
Recommendation to respectfully request City Council approval of the 2017 Federal Legislative Agenda as…	LongBeachCC_03072017_17-0161	3	Motion passes. Hey, thank you very much. Now we're moving on to item 22. Communication from Vice Mayor…
AS AMENDED a bill for an ordinance amending the Denver Zoning Code to revise parking exemptions for…	DenverCityCouncil_03262017_17-0161	4	All right. Pursuant to Council Bill 3.7 consideration of the Council, Bill 285 will be…
AS AMENDED a bill for an ordinance amending the Denver Zoning Code to revise parking exemptions for…	DenverCityCouncil_02212017_17-0161	5	11 Eyes, Resolution 109 has been adopted as amended. All right, Madam Secretary King, please pull up 161…
AS AMENDED a bill for an ordinance amending the Denver Zoning Code to revise parking exemptions for…	DenverCityCouncil_04032017_17-0161	6	Council is reconvene. We have one final vote to take on a previously previously amended council bill 161…
Report of the City Clerk on the Certificate of Sufficiency for Initiative No. 134, concerning…	SeattleCityCouncil_06282022_CF 314495	7	Councilmember Peterson has items four, five and six, and Councilmember Nelson has item number seven in…
A bill for an ordinance vacating a portion of the right-of-way adjacent to 2099 Chestnut Place bounde…	DenverCityCouncil_10282019_19-0776	8	All right. Seeing done council members Monday, November 4th is the last night to offer amendments.…

Figure 4.2: Dataset

The dataset contains the following key attributes:

1. **Summary:** This field contains the full text of the meeting transcript, which is processed and used for summarization and action item extraction. It is the primary input for the NLP processing module, which applies various algorithms to summarize the content.

2. **Stringlengths:** This metric represents the length of the summary text. It is used to assess the verbosity of the meeting content, and it can help detect unusually long or short summaries, which may be indicative of irregular meeting discussions.

3. **UID:** This is a unique identifier assigned to each entry in the dataset. It helps track individual records and enables the system to reference specific meetings easily for further analysis or retrieval.

4. **ID:** This numerical identifier is used to index and organize the dataset. It provides an additional layer of organization, ensuring that each record can be distinguished from others in a structured manner.

5. **Transcript:** This field holds the detailed transcription of the meeting's audio, which is used for NLP-based summarization. The transcription includes all the dialogue and discussions during the meeting and is essential for generating accurate and contextually relevant summaries.

This dataset is used to train and fine-tune the NLP models, enabling them to generate high-quality summaries and action items from the meeting transcripts.

The integration of these dataset attributes into the system's design helps the backend process the data efficiently, store it in MongoDB, and display the summarized output to users in an easily accessible format via the Chrome extension and React.js frontend.

4.3 Design

The system is designed as a Chrome extension integrated with a backend and NLP processing module to automatically generate meeting summaries and action items from online meetings. The design follows a modular architecture, allowing seamless communication between the frontend Chrome extension, backend server, and database. This architecture enables real-time processing of meeting audio and flexible data storage for user access. The primary components of the system design are as follows:

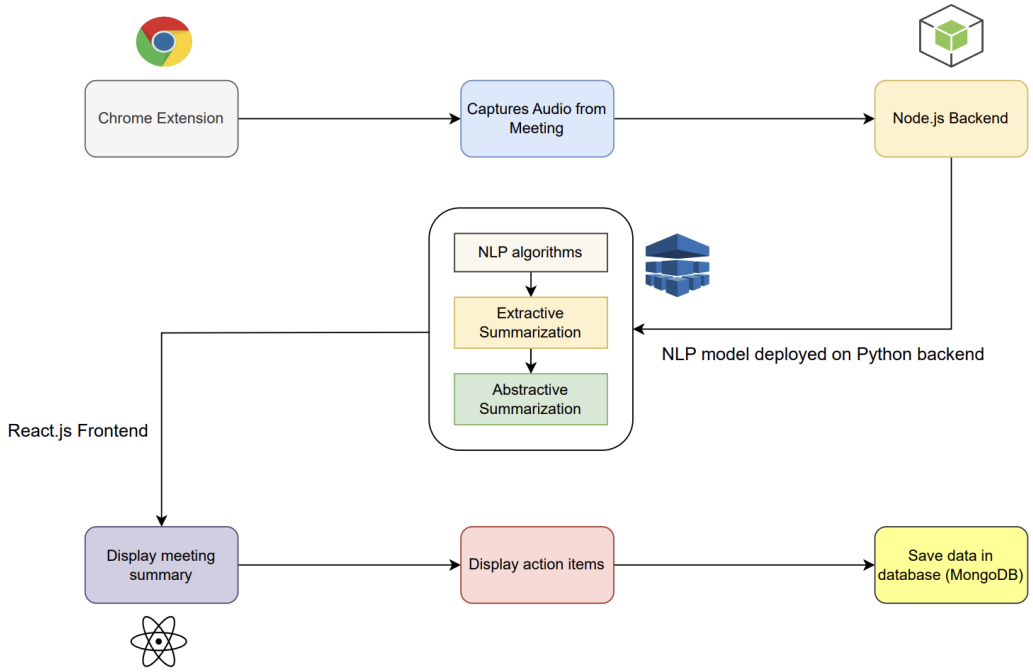

Figure 4.3: Prototype

1. **Chrome Extension:** The system starts with a Chrome extension that users can activate during online meetings. This extension captures audio directly from the browser, ensuring it records the conversation without requiring additional software. The extension provides an easy-to-use interface, making it accessible for users to initiate the summarization process with minimal setup.

2. **Audio Capture and Speech Processing:** Once activated, the Chrome extension captures audio from the ongoing meeting and sends it to the backend for processing. This captured audio undergoes *speech-to-text conversion* using ASR (Automatic Speech Recognition) models, transforming spoken language into text format. This text output forms the basis for the subsequent NLP-based summarization and action item extraction.

3. **Node.js Backend:** The backend is built using Node.js, which serves as the central hub for managing data flow between the Chrome extension, NLP model, and frontend display. Node.js handles incoming data from the

extension, forwards it to the NLP processing module, and communicates with the MongoDB database to store meeting summaries and action items. The backend also manages user requests and responses, ensuring smooth and efficient processing of data in real-time.

4. **NLP Processing Module:** The text generated from the meeting audio is sent to a Python-based NLP module deployed on the backend. This module applies *NLP algorithms* to process and summarize the text. The summarization module consists of two components:

   (a) *Extractive Summarization:* Identifies and selects the most relevant sentences from the text, providing a concise summary based on the original phrasing.

   (b) *Abstractive Summarization:* Generates new, coherent sentences that capture the main ideas of the conversation, creating a more fluid and readable summary.

5. **Database (MongoDB):** MongoDB serves as the storage solution for meeting transcripts, summaries, and action items. The backend stores processed summaries and action items in MongoDB, allowing users to retrieve previous meeting records for reference. MongoDB's flexible and scalable document storage model is well-suited for handling diverse meeting data, enabling efficient storage and retrieval of user data.

6. **React.js Frontend for Summary Display:** The processed summary and extracted action items are displayed to the user through a React.js frontend. The frontend provides an organized view, showing the summarized meeting content and specific action items derived from the conversation. Users can review key points and assigned tasks directly from the extension interface, making it easy to access essential information quickly.

CHAPTER 5
Conclusion

This project showcases the potential of Natural Language Processing (NLP) to revolutionize meeting documentation through automated summarization. With the rise of virtual meetings, there is an increasing need to efficiently capture and distill key information from extensive conversations. Leveraging NLP techniques, the system developed in this project offers a robust solution that combines extractive and abstractive summarization to deliver accurate and concise summaries. By processing audio-to-text conversion outputs, the tool utilizes NLP models to identify and retain essential information, reducing the manual effort required for note-taking and improving the accessibility of meeting insights.

The system architecture, centered on NLP algorithms, integrates both extractive and abstractive summarization methods to produce summaries that capture the essence and context of discussions. The extractive approach prioritizes key sentences, while the abstractive model generates new sentences, offering a cohesive and natural summary. Additionally, the action item identification module uses Named Entity Recognition (NER) and dependency parsing to extract tasks and responsibilities, making the summarization tool not only informative but actionable for users.

Despite advancements, NLP-based summarization has inherent challenges, such as handling domain-specific language and ensuring accuracy in complex conversations. However, this project's modular design allows for iterative improvements, with potential future developments including enhanced language models, multilingual support, and refined action item extraction techniques.

In conclusion, this NLP-driven meeting summarization tool addresses a critical need in modern workplaces by automating the extraction of valuable insights from meetings. By enabling efficient documentation, this tool empowers users to focus on meaningful participation.

References

[1] J. Yan and S. Zhou, "A Text Structure-based Extractive And Abstractive Summarization Method," in *Proc. 7th Int. Conf. on Intelligent Computing and Signal Processing (ICSP)*, 2022, pp. 678–684.

[2] N. Bharti, S. N. Hashmi, and V. M. Manikandan, "An Approach for Audio/Text Summary Generation from Webinars/Online Meetings," in *Proc. 13th Int. Conf. on Computational Intelligence and Communication Networks (CICN)*, 2021, pp. 6–12.

[3] S. Muppidi, J. Kandi, B. S. Kondaka, C. Kethireddy, and S. E. Kandregula, "Automatic Meeting Minutes Generation Using Natural Language Processing," in *Proc. Int. Conf. on Evolutionary Algorithms and Soft Computing Techniques (EASCT)*, 2023, pp. 124–130.

[4] A. M., A. Danda, H. Bysani, R. P. Singh, and S. Kanchan, "Automation of Text Summarization Using Hugging Face NLP," in *Proc. 5th Int. Conf. for Emerging Technology (INCET)*, 2024, pp. 78–85.

[5] G. Patil, B. Sapariya, K. Saravanan, and P. Gotarne, "Chrome Extension for Speech-to-Text Conversion and Text Summarization Using NLP," in *Proc. IEEE Int. Conf. on Information Technology, Electronics and Intelligent Communication Systems (ICITEICS)*, 2024, pp. 1–6.

[6] T. Islam, M. Hossain, and M. F. Arefin, "Comparative Analysis of Different Text Summarization Techniques Using Enhanced Tokenization," in *Proc. 3rd Int. Conf. on Sustainable Technologies for Industry 4.0 (STI)*, 2021, pp. 56–63.

[7] M. Majeed and K. M. T., "Comparative Study on Extractive Summarization Using Sentence Ranking Algorithm and Text Ranking Algorithm," in *Proc. Int. Conf. on Power, Instrumentation, Control and Computing (PICC)*, 2023, pp. 39–45.

[8] D. Chaurasia and V. D. K., "Enhancing Text Summarization through Parallelization: A TF-IDF Algorithm Approach," in *Proc. 2nd Int. Conf. on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*, 2024, pp. 1503–1509.

[9] D. M. Alsekait, W. Almanaseer, I. A. Gomaa, M. A. Zeid, S. Baligh, H. Alaa, J. Hegazy, J. Magdy, A. T. Eldmrat, and D. S. AbdElminaam, "Harnessing Deep Learning for Effective Extractive Text Summarization: A Comparative Study," in *Proc. IEEE Conf. on Intelligent Methods, Systems, and Applications (IMSA)*, 2024, pp. 581–590.

[10] D. Ganesh, M. Kiran Kumar, J. Varsha, K. J. Naik, K. Pranusha, and J. Mallika, "Implementation of Novel Test Rank Algorithm for Effective Text Summarization," in *Proc. Int. Conf. on Advances in Computing, Communication and Applied Informatics (ACCAI)*, 2023, pp. 32–40.

[11] R. Kachhoria, N. Daga, H. Ramteke, Y. Akotkar, and S. Ghule, "Minutes of Meeting Generation for Online Meetings Using NLP & ML Techniques," in *Proc. 2024 Int. Conf. on Emerging Smart Computing and Informatics (ESCI)*, 2024, pp. 1–6.

[12] N. Zade, G. Mate, K. Kishor, N. Rane, and M. Jete, "NLP Based Automated Text Summarization and Translation: A Comprehensive Analysis," in *Proc. 2nd Int. Conf. on Sustainable Computing and Smart Systems (ICSCSS)*, 2024, pp. 517–528.

[13] R. R. Kiruba, J. Kavitha, S. Sowmyayani, R. Preethi, S. Anitha, and C. Saranya Jothi, "Text Summarization based on Feature Extraction using GloVe and B-GRU," in *Proc. 2nd Int. Conf. on Sustainable Computing and Smart Systems (ICSCSS)*, 2024, pp. 485–491.

[14] P. Gupta, S. Nigam, and R. Singh, "A Ranking based Language Model for Automatic Extractive Text Summarization," in *Proc. 1st Int. Conf. on Artificial Intelligence Trends and Pattern Recognition (ICAITPR)*, 2022, pp. 136–145.

[15] I. P. Tummala, "Text Summarization Based Named Entity Recognition for Certain Application Using BERT," in *Proc. 2nd Int. Conf. on Intelligent Cyber Physical Systems and Internet of Things (ICoICI)*, 2024, pp. 1136–1142.

[16] Q. Zhang, C. Deng, J. Liu, H. Yu, Q. Chen, W. Wang, Z. Yan, J. Liu, Y. Ren, and Z. Zhao, "MUG: A General Meeting Understanding and Generation Benchmark," in *Proc. IEEE Int. Conf. on Acoustics, Speech, and Signal Processing (ICASSP)*, 2023, pp. 4276–4282.

[17] X. He and J. Ling, "A Video Summarization Method Based on Key Frames Extracted by TMOF," in *Proc. IEEE Conf. on Electronics and Information*, 2012, pp. 92–98.

[18] J. Zhang and H. Yuan, "A Comparative Study on Extractive Speech Summarization of Broadcast News and Parliamentary Meeting Speech," in *Proc. Int. Conf. on Asian Language Processing (IALP)*, 2014, pp. 111–118.

[19] D. Singhal, K. Khatter, T. A, and J. R, "Abstractive Summarization of Meeting Conversations," in *Proc. IEEE Int. Conf. for Innovation in Technology (INOCON)*, 2020, pp. 1–5.

[20] S. Yao, X. Pan, H. Dong, C. Kong, and X. Wu, "Adaptive-Bert Network for Advertising Text Generation," in *Proc. IEEE 6th Int. Conf. on Electronic Information and Communication Technology (ICEICT)*, 2023, pp. 192–198.

[21] Z. Deng, S. Yoon, T. Bui, F. Dernoncourt, Q. H. Tran, S. Liu, W. Zhao, T. Zhang, Y. Wang, and P. S. Yu, "Aspect-based Meeting Transcript Summarization: A Two-Stage Approach with Weak Supervision on Sentence Classification," in *Proc. IEEE Int. Conf. on Big Data (BigData)*, 2023, pp. 636–643.

[22] G. Vadlamudi, N. Vemuru, S. Vangapalli, R. K. Surapaneni, and S. Nimmagadda, "Meeting Summarizer Using Natural Language Processing," in *Proc. 6th Int. Conf. on Trends in Electronics and Informatics (ICOEI)*, 2022, pp. 1610–1616.

[23] S. S. Bhat, U. A. Nawaz, S. M, N. Tantri, and V. Vasudevan, "Jotter: An Approach to Summarize the Formal Online Meeting," in *Proc. Int. Conf. on Ambient Intelligence, Knowledge Informatics, and Industrial Electronics (AIKIIE)*, 2023, pp. 1–5.

[24] A. Mahadevan, A. Pillai, and J. Lamba, "Minutes: Hybrid Text Summarizer for Online Meetings," in *Proc. 14th Int. Conf. on Computing Communication and Networking Technologies (ICCCNT)*, 2023, pp. 32–38.

[25] D. Alqahtani and M. Al-Yahya, "Exploring Arabic Pre-Trained Language Models for Arabic Abstractive Text Summarization," in *Proc. 10th Int. Conf. on Social Networks Analysis, Management and Security (SNAMS)*, 2023, pp. 215–223.

[26] J. Jiang, H. Zhang, C. Dai, Q. Zhao, H. Feng, Z. Ji, and I. Ganchev, "Enhancements of Attention-Based Bidirectional LSTM for Hybrid Automatic Text Summarization," *IEEE Access*, vol. 9, pp. 123660–123675, Sep. 2021.

[27] X. Chen, S. Gao, M. Li, Q. Zhu, X. Gao, and X. Zhang, "Write Summary Step-by-Step: A Pilot Study of Stepwise Summarization," *IEEE/ACM Trans. Audio, Speech, and Lang. Process.*, vol. 32, pp. 1406–1417, 2024.

[28] K. Kulkarni and R. Padaki, "Video Based Transcript Summarizer for Online Courses Using Natural Language Processing," in *Proc. 2021 IEEE Int. Conf. on Computation System and Information Technology for Sustainable Solutions (CSITSS)*, 2021, pp. 45–52.