iJRASET
International Journal For Research in Applied Science and Engineering Technology



INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** IV **Month of publication:** April 2026

DOI: https://doi.org/10.22214/ijraset.2026.79954

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue IV Apr 2026- Available at www.ijraset.com

AI-Driven Application for Data Structuring and Intelligent Meeting Analysis

Vijaya Balpande, Shivani Duble, Pranali Barapatre, Pranali Khapne, Trupti Lakhe, Shital Ghivtonde
Department of Computer Science and Engineering, Priyadarshini College of Engineering, Nagpur

Abstract: ***The rapid growth of remote work and online learning has created a strong need for intelligent tools to record, analyze, and summarize meetings effectively.This paper presents an application DataMate, an AI-powered Virtual Meeting Summarization System designed to automatictranscription and generation of concise meeting summaries in remote and hybrid environments The system integrates Automatic Speech Recognition (ASR), and Generative AI with Large Language Models (LLMs) to extract key discussion points and attribute speaker contributions accurately. Leveraging Natural Language Processing (NLP) and Machine Learning (ML), DataMate delivers precise and efficient summaries . The application described not only helps with audio summarization but will enable the use of data analytics in the creation of content and answering questions from discussion meetings. The Application enables greater collaboration, productivity, and decision-making within an organization or learning environment by overcoming various challenges including accuracy and privacy concerns.***
Keyword: Data Analytics, Natural Language Processing (NLP), Automatic Speech Recognition (ASR).

I. INTRODUCTION

A. Background of the Project

In the modern digital era, businesses, students, and professionals are always making a lot of unstructured data, like meeting recordings, screen captures, raw datasets, and text. To get useful information from this data, you need a lot of tools and a lot of work.Most of the time, existing solutions are broken up into different tools for recording, transcribing, and analyzing and visualizing data. This separation makes things more complicated, less productive, and less efficient when it comes to managing information[1].
To overcome these limitations, theDATAMATE- AI-Driven Application for Data Structuring and Intelligent Meeting Analysis is developed as an integrated Application that combines data transformation, deep analysis, visualization, and meeting intelligence into asingle unified platform. The system converts raw inputs into structured formats.

B. Problem Statement

In current workflows, users face the following challenges:

- Difficulty in converting unstructured data into structured tabular format
- Time-consuming manual data cleaning and validation.
- Lack of automatic insight generation from datasets
- Inability to quickly summarize long meetings or recordings
- Relying on several apps to record ,transcribe and Analyze.

These problems make things less efficient, slow down decision-making, and make operations more complica-ted. So,we need a single intelligent system that canautomate all of the steps in recording, transforming, analyzing, and reporting.[2]

C. Need of the Project

The project is needed to address real-world data management and meeting analysis problems.

- Improved Decision Making Support: Decision makers require fast, easy-to-understand trends and information.
- Centralization for Managers: One-stop access and management for recordings, documents, and analytical findings are required by users.
- Automation using AI-Artificial Intelligence: It will increase precision in transcription, cleansing, and analysis.

D. Objectives of the Project

The primary objectives of the All-in-One Data Analysis and Meeting Intelligence Platform are:

©IJRASET: All Rights are Reserved

- To design and develop a unified intelligent system for data analysis and meeting processing.
- To generate automatic meeting transcripts and executive summaries.
- To perform deep data analysis and produce strategic insights.
- To visualize data using charts and graphical reports.
- To enable downloading of analysis results in structured formats such as CSV.
- To reduce manual workload and improve overall productivity.

E. Scope of the Project:

The scope of the project covers multiple intelligent features and applications.
Functional Scope

- Screen and meeting recording
- AI-based transcription and summarization
- Smart data transformation (unstructured to structured format)
- Deep data analysis and visualization
- File management and storage system
- User profile and API configuration management

F. Application Scope

The system can be applied in:

- Educational institutions for lecture analysis
- Corporate environments for meeting intelligence
- Business analytics and decision-making
- Research and academic data processing

G. Future Scope

- Real-time analytics
- Multi-language support
- Cloud integration
- Enhanced data validation and security mechanisms

II. LITERATURE SURVEY

Significant achievements in the realm of Large Language Models (LLMs) and NLP technologies have been witnessed in recent years, resulting in the development of effective approaches to meeting summarization. Thanks to these technological advancements, researchers can now develop algorithms that automatically generate well-structured, relevant, and valuable summaries for meetings. Despite this considerable progress, several challenges remain in the domain of adaptive summarization in real-time, multilingual settings, as well as context preservation. To address these challenges, Thomas [1] developed an AI-driven algorithm for summarizing meetings using Automatic Speech Recognition (ASR) and LLMs to generate concise, decisionfocused meeting summaries. Expanding upon this, Khan and Khan [2] implemented a virtual meeting summarizer that combined Google Gemini for text generation and AssemblyAI for speaker identification. Their system automatically transcribed speech, recognized speakers, and distributed summaries via email, achieving greater accuracy and fewer errors compared to traditional extractive techniques.Arianto et al. [3] utilized Latent Dirichlet Allocation (LDA) integrated with text summarization to produce structured meeting minutes for virtual meetings. Their model identified discussion themes effectively, achieving 57.91% summarization accuracy and 64.56% topic coherence, highlighting the potential of AI in automated note generation. Building on this research direction, Mali et al. [4] used OpenAI's Whisper to transcribe lectures, after which GPT-based models were used for automatic summarization. As a result, multilingual lecture summaries were obtained that facilitated accessibility and comprehension for remote learners.
It was concluded from their research that the use of video and audio modalities positively impacts summarization outcomes.From the above discussion, it becomes evident that transcription,and speaker differentiation have become more efficient.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

Nevertheless, there remains a lack of research focused on developing a solution that could provide comprehensive, multilingual, context-aware meeting outputs.

By incorporating advanced capabilities in speech recognition, data structuring, and our project will help to overcome the limitation by providing highly accurate meeting notes that are useful for informed decision-making[5].

Author (Year)	Technique / Model Used	Dataset / Domain	Main Outcome / Accuracy	Limitations / Research Gap
Thomas (2023)	ASR + Generative AI (LLM-based Summarizer)	Business meetings, internal corpora	Improved meeting summaries and decision extraction; enhanced productivity	Limited real-time processing; privacy concerns
Arianto et al. (2024)	LDA Topic Modeling + Text Summarization	Virtual meetings (Zoom / Google Meet)	High accuracy transcription and concise summary emails	Dependent on stable connectivity; lacks domain adaptation
Mali et al. (2025)	Whisper ASR + GPT Summarizer	Lecture recordings (Education)	Better comprehension and reduced manual note taking	High computational cost, limited dataset size

Fig 2.1 Literature Survey on Existing Systems

The challenges which remain unsolved include minimization of incorrect statements, privacy concerns regarding user data, handling overlapped speech, and adapting speech summarization systems to various contexts. Potential directions of future research may include creation of real-time summarization based on multiple types of data, end-to-end models of speech to summary conversion, and AI-driven approaches to ethical summarization of meeting contents.[6]

III. SYSTEM ARCHITECTURE

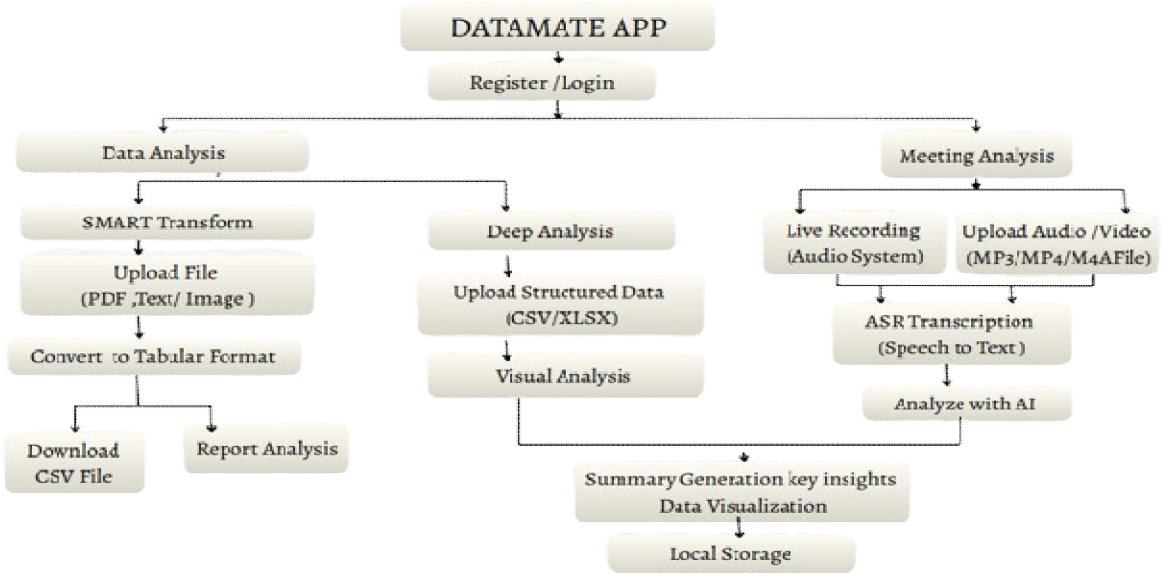


Fig 3.1 System Architecture of Data Mate Application

©IJRASET: All Rights are Reserved

1) Architecture Overview: The AI-Driven App for Data Structuring and Intelligent Meeting Analysis

Step 1: Start → Register / Login
User opens the app and logs in
Authentication is done using:
JWT / Session Management
After login, user gets 2 main options:
Data Analysis
Meeting Analysis
Step 2: Data Analysis Workflow
Data Analysis → SMART Transform
User selects SMART Transform module
Uploads files: PDF / Text / Image
Working: System extracts data using:
OCR (Optical Character Recognition) → for images/PDF
Text Parsing Algorithms
Step 3: Convert to Tabular Format
Extracted data is converted into rows & columns
Algorithms: Pattern Recognition,Data Structuring Logic
Output:Data is converted into CSV file
Step 5: Output Options
User gets: Download CSV File Report Analysis
Technologies: CSV Generator
Basic Data Analysis
5. Deep Analysis (Parallel Flow)
User uploads structured data (CSV/XLSX)
Processing: Data cleaning, Data visualization
Algorithms:Statistical Analysis, Graph Generation
Output: Summary generation key insights,Charts, graphs, insights
Step6 : Meeting Analysis Workflow
Meeting Analysis (Two Input Ways)
Option 1: Live Recording (Audio System)
Option 2: Upload Audio/Video
(MP3 / MP4 / M4A)
Step 7: ASR Transcription (Speech → Text)
Working:Audio is converted into text
Algorithm:ASR (Automatic Speech Recognition)
Deep Learning (Transformer models)
Output:Meeting conversation in text format
Step 8:Analyze with AINow system processes the text:
Algorithms Used: NLP (Natural Language Processing): Techniques used : Text Cleaning, Tokenization
System generates:Summary → (TextRank/Transformer) ,Key Insights → (TF-IDF)
Output: Summary generation, key insights,Dashboard view (graphs, insights)
Storage : Stored inLocal Storage / Database

Tools and Technology Used

- Flutter Framework
- Dart Programming Language
- Google Gemini 2.5 Flash AI
- AssemblyAI

- Optical Character Recognition (OCR)
- Natural Language Processing (NLP)
- Machine Learning Algorithms
- JSON Storage
- FL Chart Library

IV. METHODOLOGY

Overview

The proposed system, DataMate, is an Artificial Intelligence-based mobile application designed to perform automated data interpretation, meeting analysis and screen recording. The App integrates multimodal artificial intelligence models, speech recognition techniques, and natural language processing algorithms to transform raw multimedia data into structured analytical data. The system is developed using the Flutter framework and integrates intelligent data interpretation using Google Gemini 2.5 Flash AI along with advanced speech transcription services provided by AssemblyAI. The application operates on the Android platform..
The methodology consists of two primary Modules:

1) Data Analysis Module Implementation

Overview

The Data Analysis Module is designed to process and examine both structured and unstructured data from diverse file formats, including Portable Document Format (PDF) files, images, textual documents, and Comma-Separated Values (CSV) files. Leveraging advanced Artificial Intelligence (AI) techniques and data processing algorithms, it extracts relevant information, generates concise summaries, and delivers meaningful analytical insights[1]. The App is developed using the Flutter framework and integrates Google Gemini 2.5 Flash AI for intelligent semantic interpretation of extracted content.
The module operates through a structured multi-stage processing comprising file acquisition, data extraction, semantic analysis, insight generation, visualization, and secure data storage, thereby ensuring efficient and accurate data analysis within the application environment

2) Working of Data Analysis Module

The Data Analysis Module operates through multiple sequential processing stages:

- User Interaction
- Data Processing
- Artificial Intelligence Processing
- Visualization Process
- Storage and Retrieval

File Upload and Processing

a) File Acquisition

The first stage involves acquiring the file from the user through an interactive file selection interface.



Fig 4.2.1 File Upload and Processing

The file selection mechanism is implemented using a system-level file picker integrated via Flutter plugins, enabling secure and sendboxed file access. Upon selection, the application retrieves the file URI/path and associated metadata, which is then processed using Java-based I/O streams. This ensures efficient handling of multiple file formats while maintaining data integrity and access control.

File size and specification:

File type	Recommen-ded Size	Maximum Limit	Processing Method
PDF/Text	Up to 10MB	15MB	Text Seg-mentation
Image (JPG/PNG)	Up to 5MB	10MB	OCR-Based Extraction

b) Byte Stream Conversion

When a file is selected, the application retrieves its URI/path along with the associated metadata, which is then processed using Java-based I/O streams. This approach enables efficient handling of diverse file formats while ensuring data integrity and enforcing access control.

c) MIMEType Classification : The system automatically detects the file type using Multipurpose Internet Mail Extensions (MIME) classification. MIME classification allows the system to determine the appropriate parsing and analysis method based on the file format.

3) Content Extraction Techniques

The content extraction stage transforms raw file data into machine-readable textual or structured information.

a) Document Parsing

For document-based files, different parsing techniques are applied:

PDF Text Extraction: PDF documents are processed using two techniques:

- Direct Text Parsing: Extracts embedded text using document parsing libraries.
- Optical Character Recognition (OCR): Used when the PDF contains scanned images instead of text.

b) DOCX File Parsing

DOCX files follow an XML-based internal structure. The system extracts content by parsing XML tags, which include textual content, formatting metadata, and structural hierarchy.

c) Image Text Extraction

For image files, OCR techniques are implemented to extract textual information. The OCR algorithm includes the following stages:

- ImagePreprocessing: Image Normalization, Noise filtering.
- Character Segmentation: Region detection Boundary identification.
- Character Recognition: Convolutional Neural Network-based recognition, Pattern classification using trained datasets

4) Text Preprocessing

After extracting text from PDF, DOC, or Image (OCR), the text is usually unorganized and noisy. It may contain unwanted characters, irregular spacing, and formatting issues.

Therefore, the system performs Text Preprocessing to clean and standardize the data before further processing.

Techniques Used

a) Tokenization

Tokenization is the process of splitting large text into smaller units called tokens.

These tokens can be: Words ,Sentences

b) Text Normalization

Text normalization converts text into standard format.

This includes:

- Converting uppercase to lowercase
- Removing redundant punctuation
- Standardizing words

5) Information Extraction

In this step, the system identifies important structured information from the cleaned text.
The goal is to convert unstructured text into identifiable data fields.
Techniques Used
Pattern Recognition using Regular Expressions (Regex): Regex is used to detect specific patterns in text. It helps identify: Numbers, IDs, Dates, email addresses, phone numbers
KeywordMatching: Keyword matching identifies labels or field names in the text.Common keywords include: Name, Age, Address, Roll No, Date.

6) Conversion into Tabular Format

After extracting all important attributes, the system organizes them into tabular structure (rows and columns).

a) Column Mapping

Column mapping assigns each extracted attribute to aspecific column.
Example:Name→Column1
Age→Column2
Roll No → Column 3

b) Row Alignment

Row alignment ensures that each record occupies one row.

c) Schema Generation

Schema defines the structure of the dataset.
Example schema
Column 1 → Name (String)
Column 2 → Age (Integer)
Column 3 → Roll No (Integer)
Conversion into CSV Format
After the table structure is created, the system converts the data into CSV (Comma-Separated Values) format.

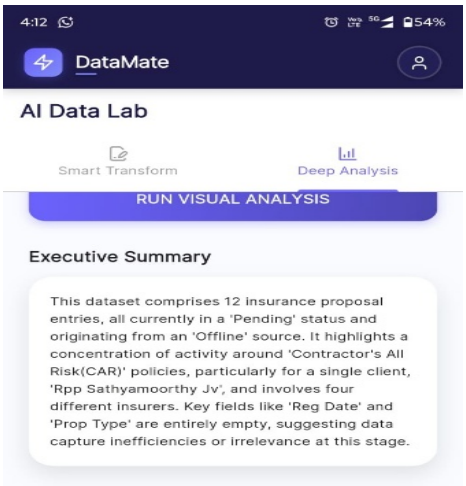


Fig 4.6 Conversion into CSV Format

7) Summary generation

Data Interpretation: Once the data is structured into tabular format, the system performs an initial interpretation of the dataset. This includes identifying the total number of records, column attributes, and data types.

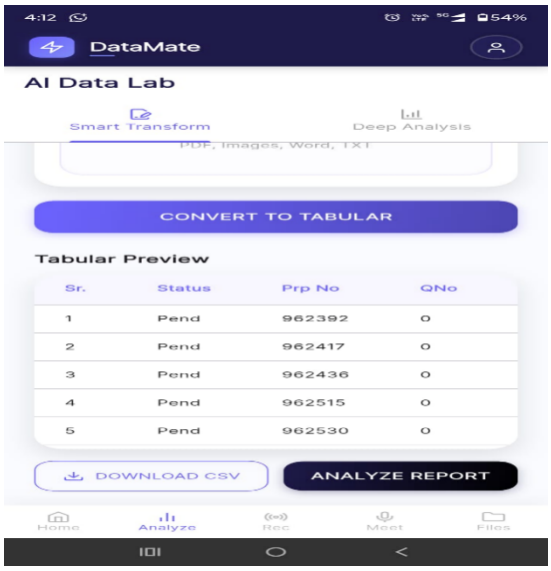


Fig.7.1 Summary Generation

Techniques Used:

- Schema analysis:
- Data profiling:

This step ensures that the system understands the structure and context of the dataset before analysis.

Name	Age	Roll No
Rahul	15	1023
Priya	16	1045

Statistical Analysis:

The system computes key statistical measures to extract meaningful information from the dataset. Techniques Used:

- Aggregation functions: (Count, Sum, Average),
- Frequency distribution:
- Percentage calculation:
- Pattern Identification:

After statistical computation, the system identifies patterns and trends within the data.

Techniques Used:

- Pattern detection algorithms:
- Rule-based analysis:
- Dominant category identification:

Content Selection: The system selects the most important data points that should be included in the summary.

Techniques Used:

- Importance scoring:
- Ranking algorithms:
- Redundancy elimination:

Natural Language Generation : The selected data points are converted into human-readable text using AI-based models. Techniques Used:

- Abstractive summarization:

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

- Transformer-based NLP models:
- Context-aware sentence generation:

Visualization Generation: If the extracted dataset contains numerical or structured information,themodulegeneratesgraphicalrepresentations.The system uses the FL Chart visualization library to display analytical results.

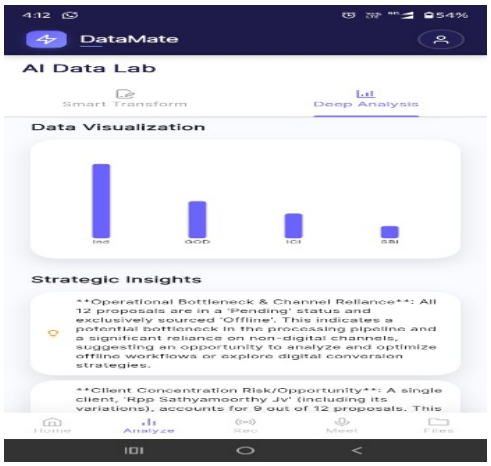


Fig 7.2 Data visualization

Algorithms Used in Visualization

- Regression Analysis:
  Identifies relationships between dependent and independent variables.
- Frequency Distribution
  Represents occurrence patterns of data values.

8) Storage and Retrieval Mechanism

After analysis completion, results are stored locally in JavaScript Object Notation (JSON) format.

Storage Advantages

- Faster Data Retrieval: Previously analyzed results can be loaded instantly without reprocessing.
- Offline Accessibility: Users can access analysis results without internet connectivity.

The storage system also supports user-based directory isolation, ensuring data privacy and organized file management.

V. MEETING ANALYSIS MODULE IMPLEMENTATION

1) Overview

The Meeting Analysis Module's job is to record audio during meetings, turn spoken communication into organized written knowledge, and make smart analytical reports. The module boosts productivity, makes it easier to automatically document meetings, and cuts down on the need for manual note-taking by recognizing important information like action items, decisions, and discussion topics.

The app was built using the Flutter application framework. It uses Google Gemini 2.5 Flash AI and AssemblyAI services to understand speech and natural language .The Flutter application framework is used in the development of the application, which incorporates Google Gemini 2.5 Flash AI and AssemblyAI services for speech recognition and natural language comprehension. The module uses AI-driven transcription techniques and native audio capture capabilities to function on the Android platform.

Audio capture, segmentation, transcription, speaker detection, semantic interpretation, insight generation, and structured storage are all steps involve in the module's multi-stage processing[2]

2) Working of Meeting Analysis Module: The Meeting Analysis Module operates through multiple sequential processing stages:

- Audio Acquisition
- Preprocessing and Segmentation
- Speech Recognition

- Natural Language Understanding
- Insight Generation
- Data Storage and Retrieval

This layered design improves system scalability, modularity, and performance efficiency.

a) Audio Acquisition Process

The module begins by capturing meeting audio using the native Media Recorder interface. The application requests user permission to access the microphone and initiates audio recording through platform-specific APIs.

File size and specifications:

File type	Recommen-ded Size	Maxim um Limit	Processing Method
Audio(MP3)	5-50MB	100MB	Chunk-based transcription
Video(MP4)	20-20MB	300MB	AudioExtracti-on+ChunkBas-ed Transcription

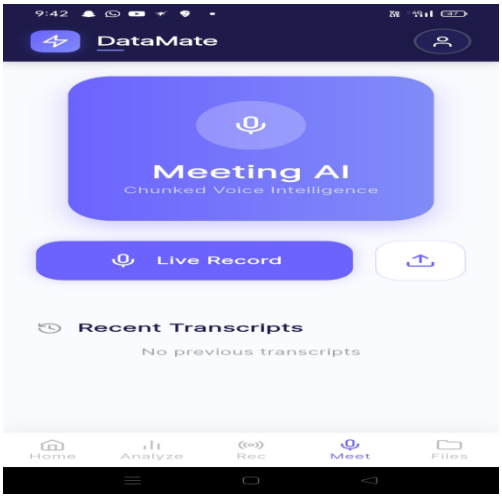


Fig 2.1 Meeting audio upload/Record

b) Audio Data Format

Captured audio data consists of:

- Pulse Code Modulation (PCM) audio stream
- Timestamp metadata

PCM audio stores raw waveform amplitude samples, which preserves high-quality speech signals and improves transcription accuracy.

c) Timestamp Synchronization

Timestamp metadata is generated simultaneously during recording. This metadata enables time-indexed transcription, allowing users to navigate transcripts based on audio duration.

3) Audio Segmentation and Preprocessing

Continuous meeting recordings generate large audio datasets. To optimize transcription performance, the system divides audio into smaller segments.

a) Sliding Window Segmentation technique

The module uses a Sliding Window Segmentation technique to partition audio streams into one-minute segments.

Working Principle:

- A fixed-length window is applied across the audio stream.
- Each window extracts a segment of audio data.
- The window moves sequentially across the timeline.

b) Audio Preprocessing : Before transcription, the system performs audio preprocessing including:

Audio Preprocessing Techniques

- Noise Filtering
  Noise reduction uses DSP technology to remove noise from the background. This involves converting the audio signal into the frequency domain using FFT, followed by suppression of noise through spectral subtraction and Wiener filtering.
- Silence Detection
  Silence detection identifies inactive speech segments using Voice Activity Detection (VAD). It analyzes short-term energy and Zero Crossing Rate (ZCR) of small audio frames (10–30 ms) to classify speech and silence accurately
- VolumeNormalization
  Volume normalization is responsible for ensuring consistency in sound volume through changes in amplitude through gain. The process is achieved through Dynamic Range Compression and Automatic Gain Control.
- Signal Strengthening
  Using feature extraction methods like MFCC, spectrogram analysis, and pitch detection, signal enhancement increases the clarity of speech. Adaptive filtering and deep learning models are used to improveand reconstruct speech quality. quality.
  The performance of speech recognition is enhanced by these preprocessing steps.

4) Transcription of Speech to Text

AI-based speech recognition engines receive the segmented audio data. **ASR (Automatic speech recognition)** ASR uses deep learning models to translate spoken language into textual representation. There are several sub-stages in the process:

a) Speaker Pattern Detection

The module identifies speaker transitions and dialogue flow using voice pattern analysis.



Fig3.2 Speech to text conversion

- Voice Embedding Generation

Each speaker's voice characteristics are converted into numerical vector representations called voice embeddings. These embeddings capture speech features such as pitch, tone, and vocal frequency.

- Clustering-Based Speaker Identification

Clustering algorithms group similar voice embeddings together. Each cluster represents an individual speaker.

- Audio Fingerprint Matching

Audio fingerprinting techniques compare voice segments with existing speaker patterns. This helps identify recurring speakers and improves dialogue structure understanding.

5) Meeting Understanding Using Natural Language Processing: After transcription, the module applies advanced Natural Language Processing techniques to interpret meeting content.

a) Intent Detection

Intent detection identifies the purpose of spoken statements. The system classifies statements into predefined categories such as instruction, discussion, agreement, or request.

Transformer-based sequence classification models analyze linguistic patterns to detect conversational intent.

b) Action Item Recognition

Action item recognition identifies tasks assigned during meetings. The algorithm detects imperative statements, responsibility indicators, and deadline references.

c) Decision Extraction

Decision extraction algorithms identify finalized agreements or resolutions. The system detects keywords such as approval indicators, confirmations, and consensus expressions.

d) Topic Segmentation

Topic segmentation divides meeting conversations into distinct discussion segments. The system applies semantic similarity algorithms to group related conversation segments into topics.

6) Summary Generation and key insights

The module generates structured meeting insights using hybrid summarization techniques.



Fig 6. Summary and key point generation

a) Extractive Summarization

Important quotes are chosen from the transcript by extractive summarization. Contextual significance, semantic relevance, and frequency scoring are the foundations of sentence ranking.

b) Abstractive Summarization

Abstractive summarization interprets the meaning of the transcript to produce new sentences. Human-readable summaries are generated using transformer-based sequence-to-sequence models.

c) Key Discussion Points

- Topic Segmentation Algorithms: Create distinct discussion sections for each meeting.
- Semantic Importance Ranking: Finds discussion topics that are extremely pertinent.
- Keyword Extraction: identifies key terms and focus words using TF-IDF, RAKE, and TextRank. Similar discussion sentences are grouped together usingclustering-based content grouping. Hidden discussion themes can be found using Latent Topic Modeling (LDA, NMF).

7) Actionable Task Lists

- Intent Detection Models: Identifies tasks, requests, and commitments.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

- Named Entity Recognition (NER): Detects people, dates, deadlines, and resources.
- DependencyParsing:Understands grammatical relationships in sentences.

Decision Highlights

- Decision Sentence Classification: Detects final agreed decisions.
- Dialogue Act Recognition: Identifies agreement, confirmation, or conclusion statements.
- Context-aware NLP Models: Understands decision context across multiple sentences.
- Sentiment and Agreement Detection: Identifies approval or rejection patterns.

8) Storage and Data Organization

The meeting analysis results are stored in structured JSON format to ensure efficient retrieval and organization.
Transcript JSON Storage: Stores complete transcript text along with timestamp alignment.
Insight JSON Storage: Stores extracted meeting insights including tasks, summaries, and decisions.
Indexed Timestamp Mapping: Provides quick navigation to specific audio segments corresponding to transcript content.

VI. CHALLENGES AND LIMITATIONS

1) System Audio Recording: The app cannot record the system audio in the Android operating systems 10 and beyond due to privacy concerns.
2) Restrictions of API & Cost Involved: Some AI services, including Google Gemini, have usage limitations; thus, intensive use will need paid plans.
3) Large File Processing Time: Large files take longer time to process and analyze due to large size or high definition nature of files.
4) Compatibility with Old Devices: Certain functionalities may be incompatible with old devices or phones having less RAM.
5) Lack of Backup Facility: All data is saved only locally; hence, its deletion will mean permanent loss of data

VII. RESULT

The DataMate application is successfully developed and tested for both data analysis and meeting analysis functionalities.The system effectively processes multiple input formats such as PDF, images, text, and audio files.
The Data Analysis Module accurately converts unstructured data into structured tabular format and generates statistical insights and visualizations. OCR techniques enabled efficient text extraction from images and scanned documents.
Transcription of spoken words into text is done by the Meeting Analysis Module in an effective manner using Artificial Intelligence models for ASR. The module could achieve transcription accuracy of about 85-90 percent owing to preprocessing steps including noise reduction, VAD, and AGC. The other output includes meeting notes, main discussion points, action items, and decisions.
In summary, the system ensures efficiency by automating the process of information structuring.Overall, the system reduces manual effort, improves productivity, and provides meaningful insights through automated data structuring and intelligent meeting analysis.

VIII. CONCLUSION AND FUTURE SCOPE

This paper showcases the capabilities of AI-powered summarization that can facilitate virtual meeting documentation using DataMate Application. The use of artificial intelligence technologies such as ASR, NLP, and LLM allows creating relevant and accurate summaries. With the help of machine learning technology, it becomes possible to collect key points without information overload. However, there are still problems such as attribution of speakers, overlap of dialogues, confidentiality, and domain adaptation. In future, attention will be paid to multimodal learning, live summarization, and ethical AI technologies. Furthermore, in future, there may be added support of multiple languages, AI chat assistance, more efficient voice recording, and performance on any device.

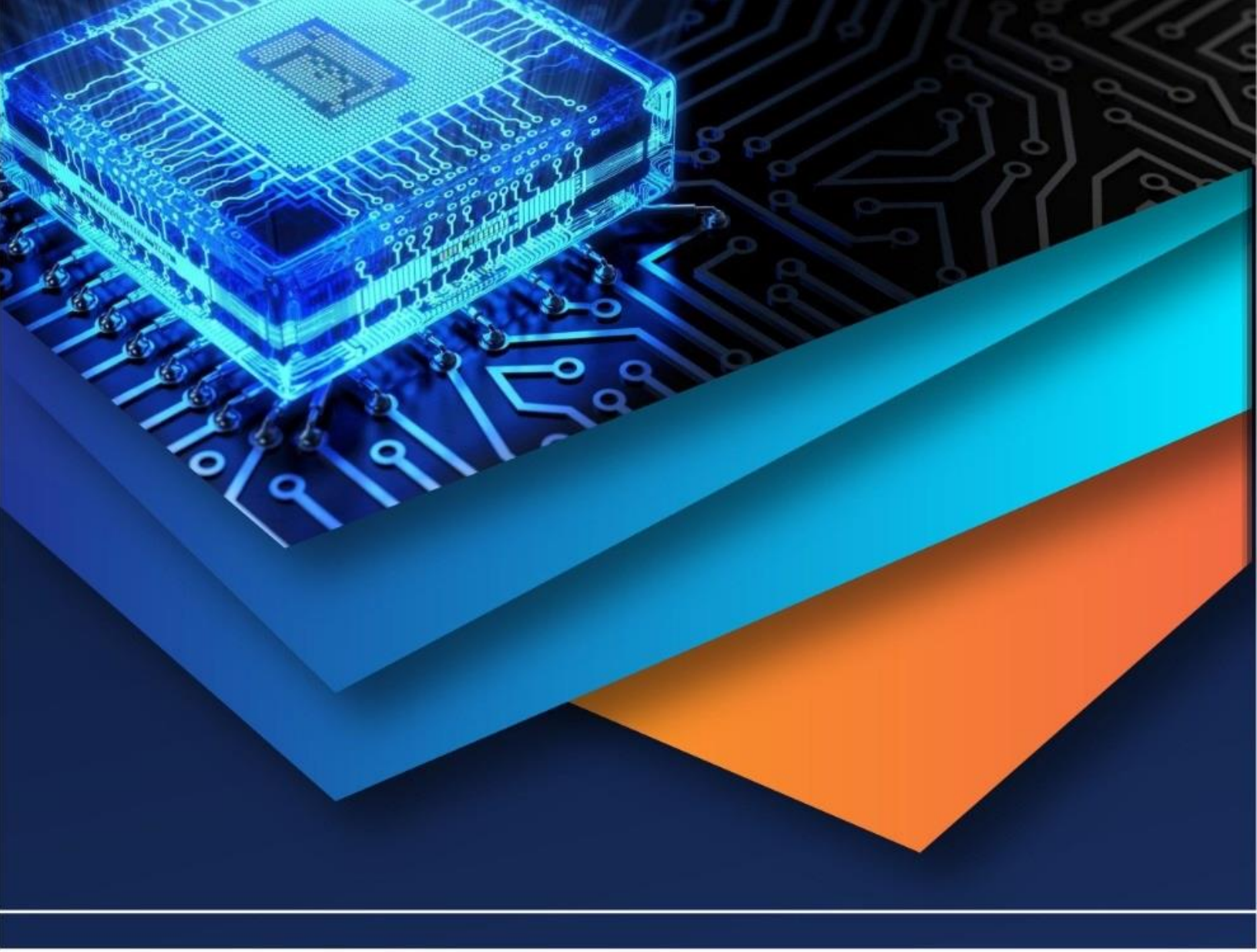
REFERENCES

[1] Swapnali Lakhan Mali, Balaji Chougule, et al. “A Review on AI based lecture summarizer” Vol. 06, Issue 04, pp.9817-9821, April, 2025 Available online athttp://www.scienceijsar.com
[2] Rakhmat Arianto , Rosa Andrie Asmara , Usman Nurhasan ,et.al “Automatic notes based on video records of online meetings using the latent Dirichlet allocation method” Vol. 14, No. 4, August 2024, pp. 4147~4153 doi: 10.11591/ijece.v14i4.pp4147-4153

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |

[3] Sibin-Thomas, et al. “Revolutionizing Meeting Analysis: An AI-Powered Approach to Automatic TranscriptionandSummarization” Vol: 1 & Iss: 2023, doi.org/10.51219/JAIMLD/sibin-thomas/346.
[4] N. Khodake, S. Kondewar, S. Bhandare, et al. “Automatic generation of meeting minutes using NLP,” international Journal for Research in Applied Science and Engineering Technology, vol. 11, no. 5, pp.7015-7019,May 2023 doi: 10.22214/ijraset.2023
[5] J. Shah and N. Jain, “Advances in automatic meeting minute generation: A survey,” International Journal of Advanced Research in Science, Communication and Technology, pp. 503–510, Feb. 2023, doi: 10.48175/ijarsct-8328.
[6] U. Sharma, M. Singh, and H. Singh,et.al “Team the turing TESTament @ AutoMin 2021: A pipeline based approach to generate meeting minutes using TOPSIS,” in First Shared Task on Automatic Minuting at Interspeech 2021, Sep. 2022, pp. 71–77, doi: 10.21437/automin.2021-9.
[7] S. H. Sheikhi, E. Iosif, O. Basov, and I. Dionysiou, et al. “SmartMinutes—A blockchain-based framework for automated, reliable, and transparent meeting minutes management,” Big Data and Cognitive Computing, vol. 6, no. 4, p. 133, Nov. 2022, doi: 10.3390/bdcc6040133.
[8] X. Chen et al., “Yanji: An automated mobile meeting minutes system,” in Proceedings - 2021 2nd International Symposium on Computer Engineering and Intelligent Communications, ISCEIC 2021, Aug. 2021, pp. 425–428, doi: 10.1109/ISCEIC53685.2021.00095.

©IJRASET: All Rights are Reserved | SJ Impact Factor 7.538 | ISRA Journal Impact Factor 7.894 |



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Call : 08813907089 (24*7 Support on Whatsapp)