Report on

SPEECH TO TEXT CONVERSION
By

Akshita Gupta-(202410116100017**)**
Akanksha Tyagi-(202410116100014**)**
Akanksha Tomar-(202410116100013**)**
Deepu Kumari-(202410116100057**)**

Session:2024-2025 (II Semester)

Under the supervision of
Mr.Apoorv Jain

KIET Group of Institutions, Delhi-NCR, Ghaziabad

DEPARTMENT OF COMPUTER APPLICATIONS
KIET GROUP OF INSTITUTIONS, DELHI-NCR, GHAZIABAD-201206
(MARCH 2025)

CHAPTER 1
INTRODUCTION

The evolution of Artificial Intelligence (AI) has paved the way for more natural and efficient human-computer interaction. Among the most impactful AI-driven technologies is Speech-to-Text (STT) conversion, which enables systems to transcribe spoken language into written text. Our project, titled "Speech-to-Text Conversion using AI", focuses on building a robust and intelligent system capable of accurately converting audio input into readable and editable text.

This project addresses the growing need for hands-free communication, real-time transcription, and improved accessibility for individuals with hearing or mobility challenges. By utilizing advanced machine learning techniques and natural language processing (NLP) models, the system can recognize spoken words, interpret them contextually, and convert them into structured text data.

The core components of our system include audio pre-processing, feature extraction, model training, and text generation. The model is trained on diverse audio datasets to improve accuracy and handle various accents, speech patterns, and background noise. We have also integrated techniques to enhance noise reduction and increase recognition accuracy in real-world environments.

Applications of this technology are vast, including voice assistants, automated transcription services, virtual meeting summarization, and assistive tools for education and accessibility. The project aims not only to showcase the power of AI in speech recognition but also to highlight the practical implementation of deep learning algorithms in solving real-world problems.

By the end of this project, we aim to deliver a functional speech-to-text conversion system that is efficient, scalable, and adaptable to various use cases across industries.

1.1 Project / Research Objective

The core purpose of this research is to develop an AI-powered system that can convert spoken language into written text with high accuracy, speed, and adaptability. The project leverages advanced techniques in Artificial Intelligence, Machine Learning, and Signal Processing to address the challenges and limitations of traditional speech recognition systems. The research objectives are outlined below in comprehensive detail.

1. **To Design and Develop a Robust Speech Recognition Model.**
    - The foundational objective of this research is to create a high-performance model capable of accurately transcribing audio speech input into text. The research will explore:
    - Data Representation: Evaluating and selecting appropriate audio representations (e.g., MFCCs, spectrograms, log-mel features) for model training.
    - Model Architectures: Comparing traditional RNN/LSTM models with advanced architectures like CNN-RNN hybrids, Transformers, and pre-trained models such as Wav2Vec 2.0 or Whisper.
    - Training Techniques: Investigating supervised and semi-supervised learning techniques to improve model generalization.

The goal is to achieve a low Word Error Rate (WER) while maintaining contextual accuracy in transcription.

2. **To Enable Multilingual and Accent-Adaptable Recognition**
    - Given the global and multicultural application of speech recognition, the project aims to build systems capable of recognizing speech across:
    - Multiple Languages: Supporting input and transcription in more than one language by integrating multilingual datasets.
    - Regional Accents and Dialects: Adapting to regional speech variations through accent normalization, phoneme adaptation, and fine-tuning on dialect-specific data.

- Language Detection: Incorporating automatic language identification from speech input to route transcription through the appropriate model or pipeline.

This objective is critical for increasing inclusivity and ensuring wider usability across geographies.

3. To Achieve Real-Time and Low-Latency Processing

- For practical and scalable deployment, speech-to-text systems must operate with minimal delay. This research will:
- Optimize Inference Time: Use model pruning, quantization, or knowledge distillation to reduce computational load.
- Streaming Capability: Develop the ability to transcribe speech in real-time using streaming recognition techniques.
- Platform Integration: Explore the deployment of models on edge devices (e.g., smartphones) and cloud platforms for real-time use.

This ensures that the system can be used in live scenarios such as voice assistants, captioning services, and live conferencing.

4. To Enhance Noise Robustness and Audio Quality Handling

- Real-world environments often include background noise, poor acoustics, and overlapping speech. This objective involves:
- Noise Handling: Training the system with noisy datasets and using data augmentation to simulate diverse acoustic environments.
- Signal Enhancement: Integrating denoising algorithms and voice activity detection (VAD) for cleaner input.
- Speaker Diarization: Differentiating between multiple speakers to improve transcription clarity in conversations or meetings.

A noise-robust model will increase accuracy and reliability in uncontrolled environments like streets, offices, or classrooms.

5. **To Evaluate and Benchmark System Performance**

- The research aims to systematically evaluate the system across various metrics:
- Accuracy Metrics: Measuring Word Error Rate (WER), Sentence Error Rate (SER), and Character Error Rate (CER).
- Latency and Efficiency: Assessing model performance in terms of speed, memory usage, and energy efficiency.
- User Experience: Conducting usability testing with diverse user groups for feedback on transcription quality and interface design.

Evaluation will be both quantitative (statistical metrics) and qualitative (user feedback), ensuring comprehensive analysis.

6. **To Develop a Usable End-to-End Application Prototype**

- Beyond theoretical research, a working prototype will be created to demonstrate practical feasibility. This includes:
- Front-End Interface: Designing a web or mobile interface for users to input speech and view transcribed results in real-time.
- Back-End Integration: Building APIs to handle audio input, process it through the AI model, and return textual output.
- Additional Features: Optional features such as translation, transcription editing, and saving/exporting results.

The prototype serves as a proof of concept and a platform for future improvements .

1.2 Scope Of Project

The scope of this project defines the boundaries, focus areas, and extent of development involved in creating an AI-based speech-to-text conversion system. It includes the specific functionalities to be developed, the technologies and techniques to be used, the target users and environments, as well as the expected deliverables and limitations.:

1.Functional Scope

The primary functional components that fall within the scope of this project include:

- **Audio Input Processing**
  Capturing and preprocessing audio signals from various input sources such as microphones, audio files, or real-time streams.

- **Speech Recognition Engine**
  Developing and integrating a machine learning model that transcribes spoken language into accurate, structured text.

- **Language and Accent Handling**
  Supporting transcription across multiple languages and regional accents to enhance usability and inclusiveness.

- **Noise Filtering and Audio Enhancement**
  Improving audio quality through background noise suppression and signal enhancement techniques.

- **Real-Time Transcription**
  Enabling near-instantaneous transcription to support applications like live captioning , voice assistants, and virtual meetings.

- **User Interface Development**
  Creating a basic web or mobile interface where users can speak or upload audio and receive transcribed text in real-time

- **Export and Save Options**
  Allowing users to download, copy, or share transcribed text in a readable

format (e.g., .txt, .doc, .pdf).

2. Technological Scope

The project leverages the following technologies and tools:

- **Machine Learning & Deep Learning**
  Using models such as RNNs, LSTMs, Transformers, Wav2Vec 2.0, or Whisper for training and inference.

- **Natural Language Processing (NLP)**
  Post-processing of transcribed text for grammar correction, punctuation insertion, and semantic formatting.

- **Audio Signal Processing**
  Techniques like MFCC extraction, spectrogram analysis, and feature normalization for model input.

- **Programming and Tools**
  Python, TensorFlow/PyTorch, speech libraries (like SpeechRecognition, Hugging Face Transformers), Flask or Node.js for backend APIs, and HTML/CSS/JavaScript for frontend interfaces.

3. User and Application Scope

The system is designed for a wide range of use cases and users, including:

- **Academics and Students** – Automatic transcription of lectures, seminars, and presentations.

- **Professionals** – Meeting note generation, interview recording, and documentation support.

- **Hearing Impaired Individuals** – Real-time speech-to-text as an

accessibility tool.

- **Developers and Enterprises** – Use as an API service integrated into other platforms like customer support, virtual assistants, and transcription tools.

1.3 Importance of the Project

The importance of this project is grounded in both its technological innovation and social utility. It combines state-of-the-art AI techniques with practical application goals, making it a powerful contribution to the fields of machine learning, human-computer interaction, accessibility, and language processing.

1.Technological Significance

- **Advancement in Artificial Intelligence**

  This project leverages cutting-edge AI technologies such as deep learning, natural language processing (NLP), and audio signal processing. The use of models like Wav2Vec 2.0, Whisper, or Transformer-based architectures reflects the latest innovations in speech recognition. Through model training, optimization, and evaluation, the project contributes to ongoing AI research, especially in multilingual and real-time systems.

- **Contribution to Natural Language Interfaces**

  Speech interfaces are a key part of modern computing—from voice assistants (like Alexa and Siri) to dictation tools and smart devices. Developing an efficient speech-to-text system advances the field of natural language interfaces (NLIs) by providing a more intuitive, hands-free, and accessible way for humans to interact with machines.

- **Real-Time Processing and Edge AI**

  The emphasis on real-time transcription and low-latency processing positions this project at the forefront of Edge AI and real-time computing. By optimizing the model for real-world conditions, including noisy environments and diverse accents, the system can be deployed on mobile devices, IoT applications, and cloud platforms, expanding its utility and innovation scope.

2. Practical and Societal Application

- **Accessibility for People with Disabilities**

  One of the most important impacts of this project is on accessibility. For individuals with hearing impairments or cognitive differences, real-time transcription of spoken language enables greater participation in conversations, classrooms, and workplaces. The project aligns with global goals for inclusivity and universal design.

- **Educational Enhancement**

  In educational settings, speech-to-text systems can automatically transcribe lectures, tutorials, and group discussions, making learning materials available for later review. Students can focus on comprehension instead of note-taking, and non-native speakers can use the transcription to reinforce language learning.

- **Professional and Business Efficiency**

  In industries such as journalism, legal services, customer support, and healthcare, manual transcription of conversations or dictation is time-consuming and costly. An AI-based transcription tool automates this process, increasing productivity, reducing labor costs, and improving accuracy. It can also be integrated into tools for meeting summaries, call center analytics, and virtual documentation.

- **Support for Language Preservation and Translation**

  This system can support efforts to preserve endangered languages and dialects by capturing spoken content and converting it into written form. With further enhancement, it could be extended to provide real-time translations, breaking language barriers and supporting multicultural communication.

3. Market Relevance and Innovation Potential

The global speech and voice recognition market is growing rapidly, driven by trends such as voice-controlled smart homes, automated customer service, and AI-driven accessibility tools. According to industry forecasts, this market is expected to exceed $30 billion by 2030. This project, by creating a modular, scalable, and customizable speech-to-text solution, positions itself within a high-demand innovation space.

The project also opens pathways for further development such as:

- Speech-to-text APIs for third-party developers
- Integration with smart devices and mobile applications
- Expansion into emotion detection, speaker diarization, and real-time translation

As a foundational technology, this project can be extended to serve several other innovations in the AI space.

4. Research and Academic Impact

This project contributes to ongoing academic exploration in the areas of:

- Deep learning model design and performance evaluation
- Multilingual language model training
- Audio signal processing under noisy and real-time conditions
- Ethical AI development in language understanding

It provides a hands-on, interdisciplinary platform for students and researchers to engage with AI, linguistics, human-computer interaction, and software development. The research methodologies and outcomes of this project may also be published or shared in academic conferences or journals.

5. Contribution to Sustainable Development Goals (SDGs)

This project aligns with several United Nations Sustainable Development Goals, including:

- Goal 4: Quality Education – Through transcription of educational content and language support.
- Goal 8: Decent Work and Economic Growth – By improving workplace efficiency and enabling automation.
- Goal 10: Reduced Inequality – By providing equal access to technology for people with disabilities and language barriers.
- Goal 9: Industry, Innovation, and Infrastructure – By contributing to AI-powered infrastructure and innovation.

1.4 Applications and Benefits

1. Applications of the Project

The AI-based speech-to-text system developed in this project has a wide range of applications across multiple sectors. Its ability to accurately convert spoken language into written text opens up opportunities for innovation, automation, and accessibility. Below are the key application areas:

.1 Education and E-Learning

- Lecture Transcription: Automatically transcribes classroom lectures, online courses, and seminars for students to review later.
- Note-Taking Assistance: Helps students focus on listening rather than manual note-taking.
- Language Learning: Provides real-time feedback by converting spoken words into text, assisting in pronunciation and grammar correction.

.2 Accessibility and Inclusion

- Support for Hearing Impaired Users: Real-time captioning during conversations, meetings, or media playback makes spoken content accessible.
- Assistive Technologies: Forms the foundation for accessible apps that aid communication for individuals with speech or hearing challenges.

.3 Business and Corporate Use

- Meeting Transcriptions: Automates documentation of meetings, interviews, and brainstorming sessions.
- Customer Support: Enables voice-to-text input for support agents and records conversations for quality and training purposes.
- Documentation and Reporting: Speeds up creation of reports by converting verbal summaries or dictation into text.

.4 Media and Content Creation

- Subtitling and Captioning: Automatically generates subtitles for videos and podcasts.
- Voice Blogging and Vlogging: Enables content creators to dictate their content, increasing efficiency.
- Journalism: Helps journalists transcribe interviews, press conferences, and speeches quickly and accurately.

.5 Legal and Medical Fields

- Courtroom Transcriptions: Reduces dependency on manual court reporting by automating speech recording and transcription.
- Medical Dictation: Doctors can dictate notes and diagnoses, which are automatically transcribed into patient records.

.6 Smart Devices and Virtual Assistants

- Voice Commands: Enables smart devices to respond to voice inputs by converting them into actionable text commands.
- Multilingual AI Assistants: Allows assistants to understand and transcribe speech in multiple languages and accents.

2. Benefits of the Project

The speech-to-text system not only introduces technological advancements but also provides a variety of practical and long-term benefits.

2.1Time and Cost Efficiency

- Reduced Manual Work: Eliminates the need for manual transcription, saving time and labor costs.
- Real-Time Processing: Provides instant transcription, improving productivity in fast-paced environments.

2.2 Improved Accessibility

- Equal Access to Information: Breaks down communication barriers for individuals with disabilities or language limitations.
- Inclusive Technology: Promotes digital inclusion by enabling access to voice content in written form.

2.3 Enhanced Communication and Collaboration

- Better Record-Keeping: Helps maintain accurate records of verbal communication.
- Cross-Language Communication: (With future integration) Can support real-time translation, enhancing global collaboration.

2.4 Scalability and Integration

- API-Ready: Can be integrated into web, mobile, or desktop applications for a wide range of custom uses.

- Cloud and Edge Compatibility: Supports deployment in cloud-based services or on local edge devices, depending on use case.

2.5 Learning and Development

- Research Tool: Assists researchers by transcribing interviews and discussions.
- Skill Enhancement: Helps learners improve pronunciation and fluency through visual feedback of spoken language.

CHAPTER 2

METHODOLOGY

1. Project Setup

- Environment: Use Google Colab or Jupyter Notebook
- Libraries Required:
    - SpeechRecognition: for speech-to-text conversion
    - pydub: to convert MP3 to WAV (because speech_recognition works with WAV)
    - google.colab (if using Colab) for file upload
    - os to handle file paths

2. Functional Workflow

Step 1: File Upload

- Upload an MP3 file
- Ensure it checks for `.mp3` extension

Step 2: Convert MP3 to WAV

- Use pydub to convert MP3 into a compatible WAV format

Step 3: Transcribe Audio

- Load WAV file using speech_recognition
- Recognize and convert speech to text using Google Web Speech API

Step 4: Handle Errors

- Handle unknown audio or failed API requests gracefully

Step 5: Output the Transcribed Text

- Print or return the transcribed result

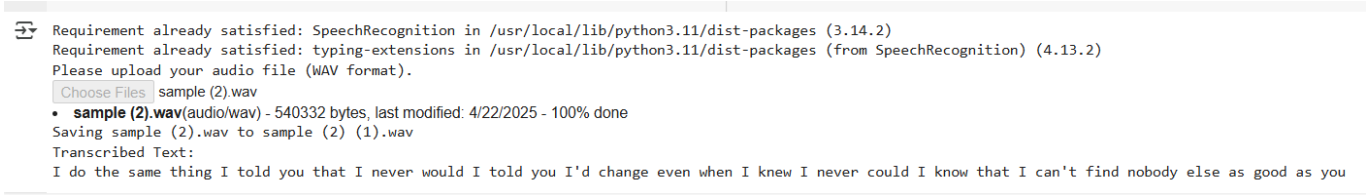
CHAPTER – 3

CODE IMPLEMENTATION

```
# Step 1: Install necessary libraries
!pip install SpeechRecognition

# Step 2: Import required
libraries import
speech_recognition as sr
from google.colab import files

# Step 3: Function to recognize speech from an
audio file def recognize_from_file(filename):
  recognizer = sr.Recognizer()
  with sr.AudioFile(filename) as source:
    audio_data = recognizer.record(source)  # Read the
    entire audio file # Recognize speech using Google Web
    Speech API
    try:
      text =
      recognizer.recognize_google(audio_data)
      return text
    except sr.UnknownValueError:
      return "Google Speech Recognition could not understand the
    audio." except sr.RequestError as e:
      return f"Could not request results from Google Speech Recognition service; {e}"

# Step 4: Main logic to upload file and transcribe
print("Please upload your audio file (WAV format).")  # Prompt to upload
audio file uploaded = files.upload()  # Upload audio file
if uploaded:
```

```
    audio_file_name = list(uploaded.keys())[0]  # Get the name of the uploaded file

    # Recognize text from the uploaded audio file
    text_from_voice = recognize_from_file(audio_file_name)

    # Print the transcribed text message in the
    output print("Transcribed Text:")
    print(text_from_voice)  # Print the
recognized text else:
    print("No file uploaded.")
```

4. OUTPUT:



3.1 LIMITATIONS:

1. Internet Dependency

- The Google Web Speech API requires a stable internet connection.
- Without internet, the transcription won't work.

2. Limited Audio Duration

- Google Web Speech API works best with short audio files (ideally under 1 minute).
- Long audio files may lead to timeouts or incomplete transcriptions.

3. No Speaker Diarization

- Cannot detect or differentiate multiple speakers in the audio.
- Useful for meeting or interview transcripts, but not supported here.

4. File Format Restriction

- Only MP3 files are accepted, and they must be converted to WAV using pydub.
- If other formats like .ogg, .flac, or .m4a are uploaded, the script will fail unless manually extended.

5. Accuracy Depends on Audio Quality

- Poor audio (background noise, echo, unclear speech) leads to low transcription accuracy.
- It may misinterpret words or skip unclear segments.

6. Language Limitation

- Default is English.
- Google API supports multiple languages, but your script needs modification to handle that.

7. No Real-Time Transcription

- Only supports pre-recorded MP3 files.
- Does not handle live audio or streaming input.

8. API Rate Limiting and Quotas

- Google’s free API has rate limits and usage quotas.
- Excessive usage may result in temporary blocking or throttling.

9. Security & Privacy Concerns

- Audio files are processed through Google’s servers.
- Sensitive content may raise data privacy concerns.

PROJECT REFERENCES:

1. **SpeechRecognition Library (Official Docs)**
   Learn how to use speech_recognition with various audio formats and recognition APIs.
   https://pypi.org/project/SpeechRecognition/
2. **Google Speech Recognition API (via speech_recognition)**
   Guide on how to integrate and use Google Web Speech API.
   https://realpython.com/python-speech-recognition/
3. **pydub Library (Audio Processing in Python)**
   Used to convert MP3 to WAV. Supports multiple audio formats.
   https://github.com/jiaaro/pydub
4. **AudioSegment - pydub API Reference**
   Official documentation for using AudioSegment.from_mp3() and .export() methods.
   https://pydub.com/
5. **Google Colab File Upload Guide**
   Helps with file uploads in a browser environment like Google Colab.
   https://research.google.com/colaboratory/local-runtimes.html
6. **Python Speech Recognition Full Guide (by GeeksForGeeks)**
   Great beginner-friendly walkthrough with examples.
   https://www.geeksforgeeks.org/speech-recognition-in-python-using-google-speech-api/
7. **Working with Audio Files in Python**
   Explains using audio formats and libraries for audio handling.
   https://www.datacamp.com/tutorial/audio-processing-python
8. **Stack Overflow Discussion - MP3 to Text using Python**
   Community insights and solutions to common issues faced while implementing.
   https://stackoverflow.com/questions/37830094/how-to-convert-mp3-to-text-in-python