HIRE-A-THON

Voices Reimagined : AI in Action

Presentation By:
Team DUCS
Department of Computer Science, University of Delhi

Problem Statement

P3 REAL TIME SPEECH-TO-SPEECH SOLUTION WITH LLM SPEAKER DIARIZATION & EMOTION DETECTION

With the rapid advancements in Natural Language Processing (NLP) and machine learning, there is an increasing need for technologies that enhance human communication through real-time speech processing.

Our challenge was to develop an open-source speech-to-speech solution that seamlessly integrates key components: accurate real-time speech recognition supporting multiple languages and dialects, speaker diarization to distinguish and tag different speakers, emotion detection to analyze and display speakers' emotional states based on tone and pitch, and conversation summarization to condense dialogues into essential points. By addressing these areas, we aim to create a comprehensive tool that improves real-time communication, making interactions more meaningful and efficient.

Deliverables:

(1) Speech Recognition : Develop an accurate speech recognition module that can convert spoken language into text in real-time.

(2) Speaker Diarization : Implement a system that can identify and differentiate between multiple speakers in a conversation. This feature should tag who is speaking at any given time and maintain an accurate record of dialogue flow.

(3) Emotion Detection : Integrate an emotion detection algorithm that analyzes the tone and pitch of speakers' voices to determine their emotional state (e.g., happiness, anger, sadness). This data should be presented alongside the transcribed text.

(4) Conversation Summarisation : Develop a summarization feature that condenses the dialogue into key points, allowing users to quickly grasp the essence of the conversation without needing to read or listen to the entire exchange.

Motivation
(Why did we choose this problem ?)

We chose this real-time speech-to-speech problem because:

🎯 Vision: To make human communication clearer and more meaningful in our digital world

🌟 Key Benefits:

- Breaks down language barriers with real-time translation
- Helps understand emotions behind words
- Keeps track of who says what automatically
- Creates smart summaries of long conversations

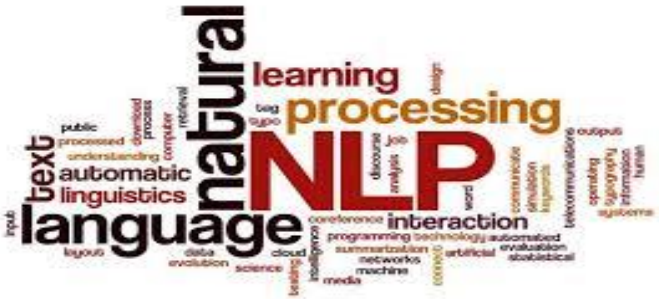
👥 Impact: Helps everyone from students and business professionals to families staying connected across borders

💡 Innovation: Combines cutting-edge speech tech with emotional intelligence to create something that doesn't just hear words - it understands people

🔑 Key Concepts Utilized

Speech Processing 🎙️

- **Real-Time Speech Recognition** 🧠
  - **Tool:** Facebook's wav2vec2-large-960h-lv60
  - **Description:** Converts spoken language into text instantly with high accuracy.
  - 🎯 **Reason:** Supports multiple languages and dialects, ensuring diverse and global communication.
- **Speaker Diarization** 🗣️
  - **Tool:** Pyannote's Pretrained Diarization Pipeline
  - **Description:** Identifies and differentiates between multiple speakers in a conversation.
  - 🎯 **Reason:** Maintains clear dialogue flow by accurately tagging each speaker in real-time.



🗣️ Natural Language Processing (NLP)

- **Conversation Summarization** ✨
  - **Tool:** Google's Gemini API
  - **Description:** Condenses lengthy dialogues into concise key points.
  - 🎯 **Reason:** Facilitates quick understanding of conversations by extracting essential information.

Emotion Analysis 🥰😠😢

- **Emotion Detection Algorithm** 🎭
  - **Tool:** SpeechBrain's emotion-recognition-wav2vec2-IEMOCAP
  - **Description:** Analyzes tone and pitch to determine speakers' emotional states.
  - 🎯 **Reason:** Provides emotional context alongside transcriptions, enhancing understanding of conversations.

Dependencies Used

TorchAudio

Streamlit

PyTorch

pyannote

SpeechBrain

Google Gemini

Transformers

USER INTERFACE

About

This application analyzes audio conversations to:

- Transcribe speech to text
- Detect speakers
- Recognize emotions
- Generate summaries
- Identify dominant emotions

Instructions

1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary

Voices Reimagined: AI in Action

Where Speech meets Emotion

Upload Audio

Drag and drop file here

Limit 200MB per file • WAV

Browse files

Voices Reimagined: AI in Action

Where Speech meets Emotion

About

This application analyzes audio conversations to:

- Transcribe speech to text
- Detect speakers
- Recognize emotions
- Generate summaries
- Identify dominant emotions

Instructions

1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary

Upload Audio

Audio file upload in progress

Drag and drop file here

Limit 200MB per file • WAV

Browse files

Neutral....

RUNNING... Stop

Voices Reimagined: AI in Action

Where Speech meets Emotion

About

This application analyzes audio conversations to:

- Transcribe speech to text
- Detect speakers
- Recognize emotions
- Generate summaries
- Identify dominant emotions

Instructions

1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary

Insert your audio file in WAV format

Upload Audio

Drag and drop file here
Limit 200MB per file • WAV

Browse files

Neutral.wav 0.8MB

Processing audio file...

Processing in progress (model loading)

Highlights the primary emotion detected in the conversation.

About

This application analyzes audio conversations to:

- Transcribe speech to text
- Detect speakers
- Recognize emotions
- Generate summaries
- Identify dominant emotions

Instructions

1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary

Analysis Results

Transcribed Dialogue with Emotions

SPEAKER_00 (['neu']): CAN YOU BREAK IT DOWN FOR US WHAT IS THIS WHAT IS SELF ATTENTION
SPEAKER_01 (['neu']): SO IMAGINE YOURE AT A NOISY PARTY AND YOU'RE TRYING TO FOLLOW LIKE THREE DIFFERENT CONVERSATIONS AT THE

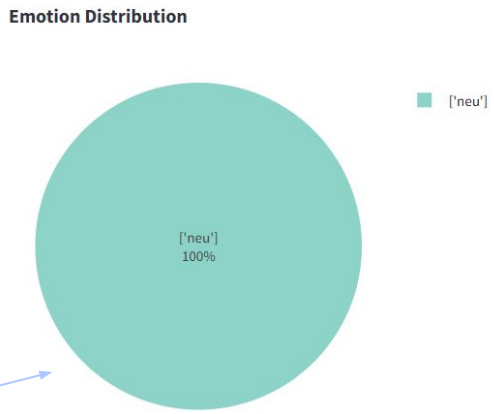
Summary

Self-attention is a technique that enables individuals to prioritize specific aspects of a situation, akin to selectively attending to conversations at a noisy party.

Dominant Emotion

['NEU']

Emotion Distribution



A pie chart visualizing the proportion of various emotions detected (e.g., neutral emotion shown at 100%).

Continued...

About

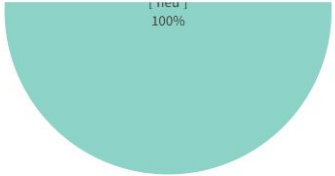
This application analyzes audio conversations to:

- Transcribe speech to text
- Detect speakers
- Recognize emotions
- Generate summaries
- Identify dominant emotions

Instructions

1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary

Self-attention is a technique that enables individuals to prioritize specific aspects of a situation, akin to selectively attending to conversations at a noisy party.



Displays the summary audio of the conversation

Summary Audio

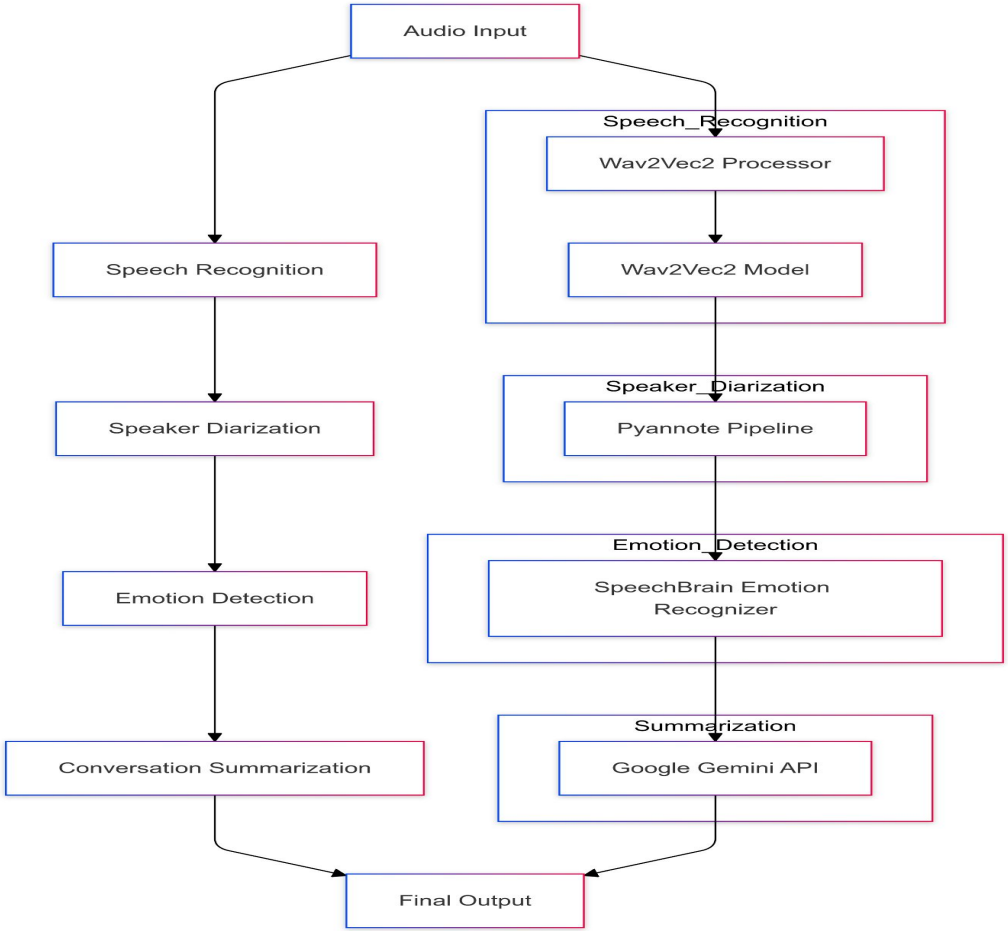
0:00 / 0:12

Download Summary Audio

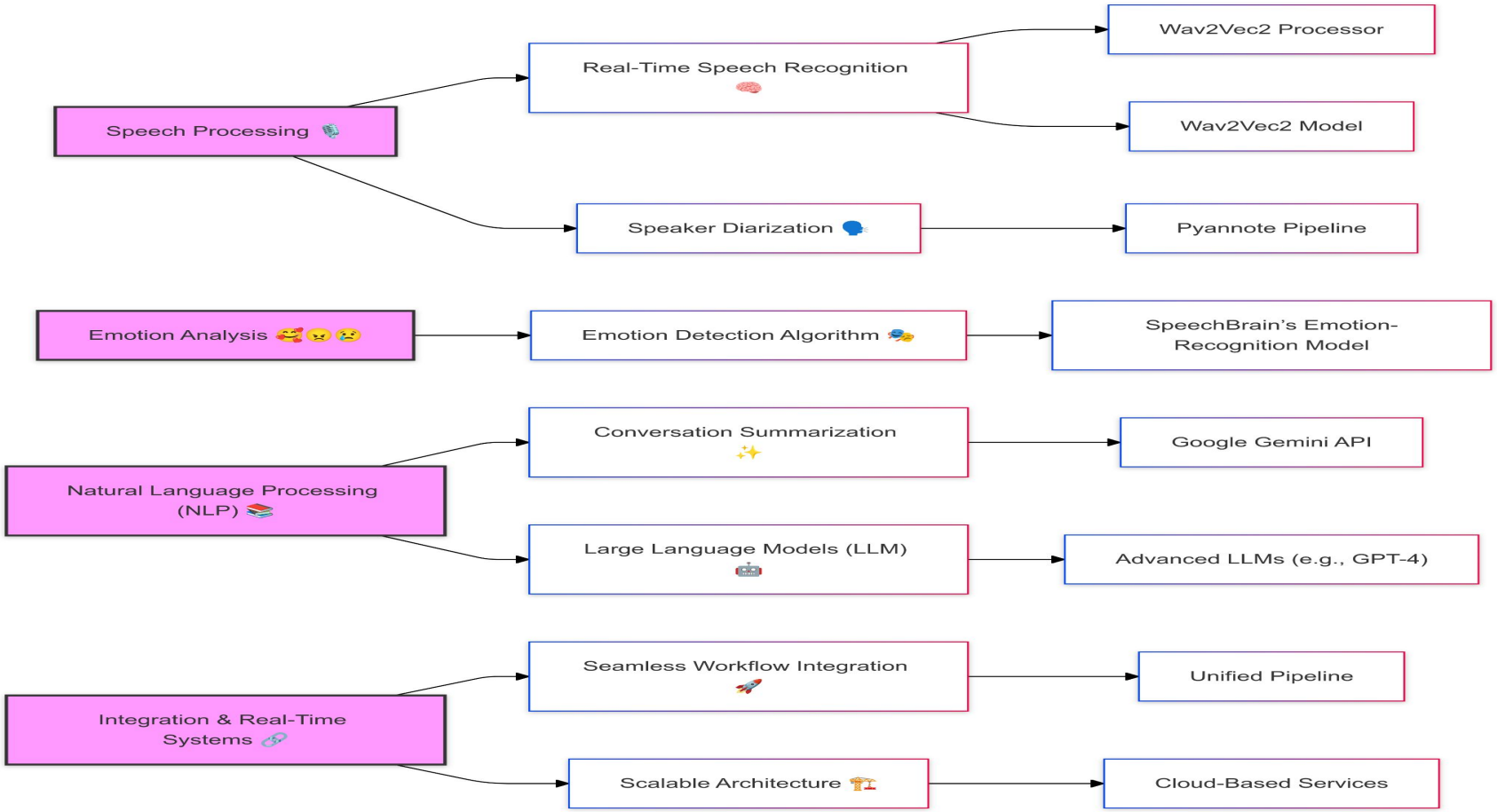
Option to download the summary audio of the conversation

Methodology and Techniques applied:

Methodology



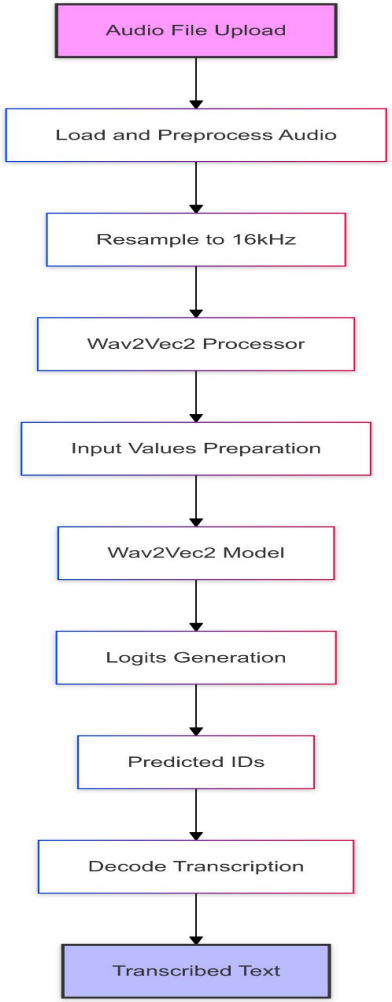
Techniques Applied



Techniques applied:

Real-Time Speech Recognition leverages advanced models like Facebook's Wav2Vec2 to convert spoken language into written text instantly. This technique ensures high accuracy and supports multiple languages and dialects, enabling seamless and diverse communication across different regions and user groups. By processing audio in real-time, it facilitates immediate transcription, making interactions more efficient and accessible.

Architecture ->



Speech Recognition

```
processor = Wav2Vec2Processor.from_pretrained('facebook/wav2vec2-large-960h-lv60')
model = Wav2Vec2ForCTC.from_pretrained('facebook/wav2vec2-large-960h-lv60')
```

Using Facebook's Wav2Vec2 model for converting speech to text. 🎙️✨

Why ?

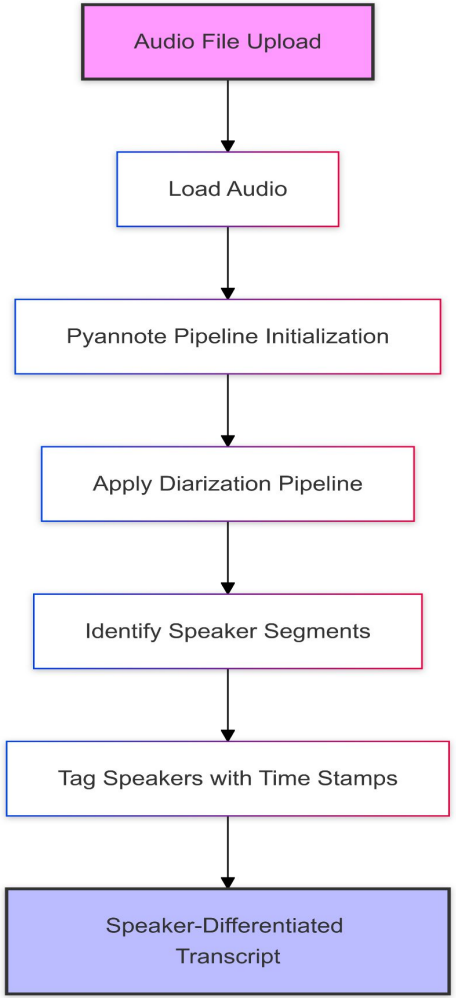
🔍 **Because :-**

- **High Accuracy 🏆:** Delivers exceptional transcription precision across diverse accents and dialects.
- **Robust Training 📚:** Trained on a massive dataset, ensuring reliable performance in various scenarios.
- **Free and Open Source 🆓:** No licensing fees, making it accessible and budget-friendly for our project.
- **State-of-the-Art Model 🧠 :** Utilizes the latest advancements in NLP and machine learning for superior speech-to-text conversion.

Techniques applied: cont.

Speaker Diarization 🗣️ utilizes Pyannote's pretrained diarization pipeline to identify and differentiate between multiple speakers in a conversation. This technique tags each speaker accurately, maintaining a clear and organized record of who is speaking at any given time. By preserving the natural flow of dialogue, it enhances the understanding of interactions, especially in scenarios involving multiple participants like meetings or interviews.

Architecture ->



Speaker Diarization

Hugging Face

```
diarization_pipeline = Pipeline.from_pretrained(
    "pyannote/speaker-diarization-3.1",
    use_auth_token="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"
)
```

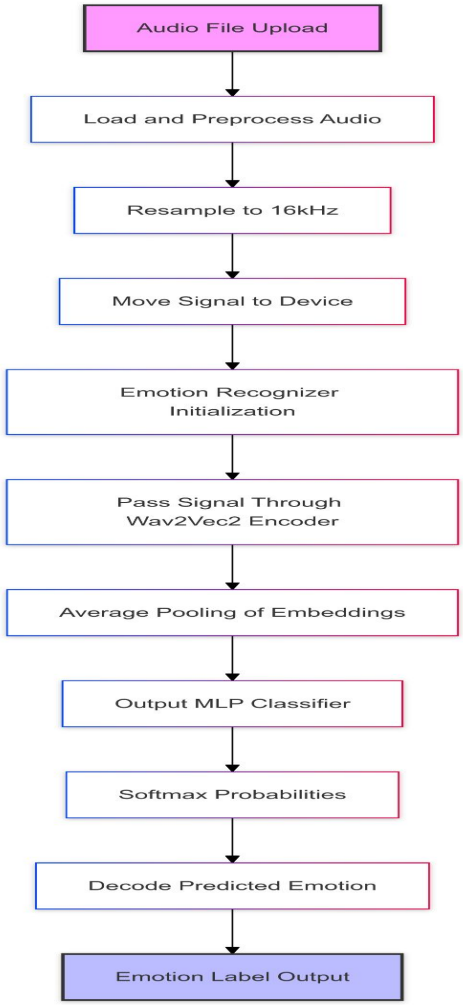
```
diarization = diarization_pipeline({"waveform": signal, "sample_rate": 16000})
dialogue_entries = []

for segment, _, speaker in diarization.itertracks(yield_label=True):
    start_time = segment.start
    end_time = segment.end
    start_sample = int(start_time * 16000)
    end_sample = int(end_time * 16000)
    segment_audio = signal[:, start_sample:end_sample]
```

OUTPUT – `Speaker SPEAKER_00 from 1.4s to 13.0s`

Techniques applied: cont.

Emotion Detection🎭 integrates SpeechBrain's emotion-recognition models to analyze the tone and pitch of speakers' voices, determining their emotional states such as happiness, anger, or sadness. This technique adds an emotional layer to the transcribed text, providing deeper insights into the conversation dynamics. By understanding the emotions behind the words, it enhances the overall communication experience and enables more empathetic and responsive interactions.

Architecture ->



Emotion Detection

```
with torch.no_grad():
    segment_signal = segment_audio.to(emotion_recognizer.device)
    if segment_signal.ndim == 1:
        segment_signal = segment_signal.unsqueeze(0)
    embeddings = emotion_recognizer.mods.wav2vec2(segment_signal)
    embeddings = emotion_recognizer.mods.avg_pool(embeddings)
    logits_emotion = emotion_recognizer.mods.output_mlp(embeddings)
    probabilities = torch.softmax(logits_emotion, dim=-1)
    predicted_index = torch.argmax(probabilities, dim=-1)
    emotion_label = emotion_recognizer.hparams.label_encoder.decode_torch(predicted_index)[0]
```

Using **SpeechBrain's** emotion recognition model to detect emotions in speech segments

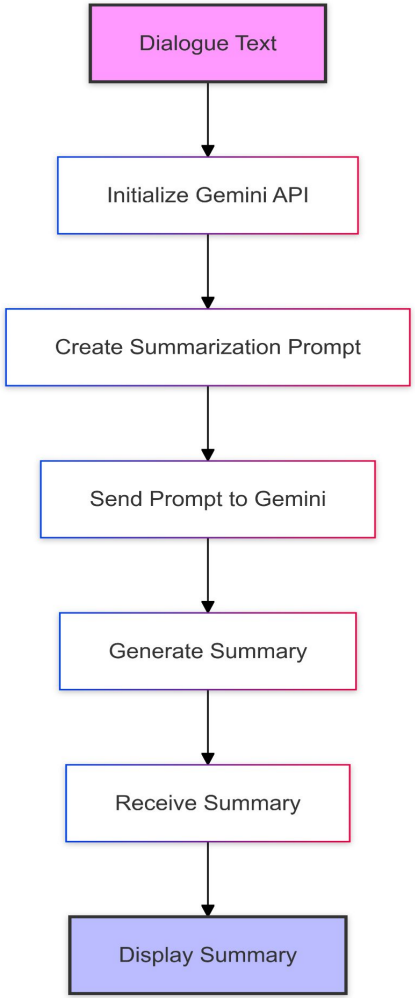
Why ?

- 🏆 **High Accuracy** : achieves an accuracy of **78.7%**,
- 🛠️ **Easy Integration:** Seamlessly integrates with our existing pipeline, allowing smooth incorporation
- 🆓 **Free to Use:** As an open-source tool, SpeechBrain reduces costs.

Techniques applied: cont.

Conversation Summarization ✨ employs Google's Gemini API to condense lengthy dialogues into concise summaries. This technique extracts key points and essential information from the transcribed conversations, allowing users to quickly grasp the main ideas without needing to review the entire exchange. It is particularly useful for generating meeting minutes, customer service logs, and providing quick overviews of lengthy discussions.

Architecture ->



Conversation Summarisation

```python
def summarize_with_gemini(text):
    try:
        model = genai.GenerativeModel('gemini-pro')
        prompt = f"You are expert in conversational summarization. Please provide a brief, factual summary of this conversation in the form of paragraph in not more than 30 word
        response = model.generate_content(prompt)
        return response.text if response.text else "[Summary generation failed]"
    except Exception as e:
        st.error(f"Summarization error: {e}")
        return "[Summary generation failed]"
```

Using **Google's Gemini-Pro** model to generate concise summaries of the conversation.

Recommendations for better performance

Future Aspects and Further enhancements

Integrating Advanced Summarization Models 📄🔗

Experiment with fine-tuning state-of-the-art LLMs like OpenAI's GPT or Google's Gemini for richer, more personalized summaries.

Automatic Speech Recognition Advancements 🗣️📈

Integration of Whisper or other next-gen ASR models for higher accuracy and multilingual support.

Dynamic Summarization ✨📑

Let users choose between concise summaries, detailed summaries, or emotion-focused summaries. Offer customization in tone—formal, conversational, or persuasive.

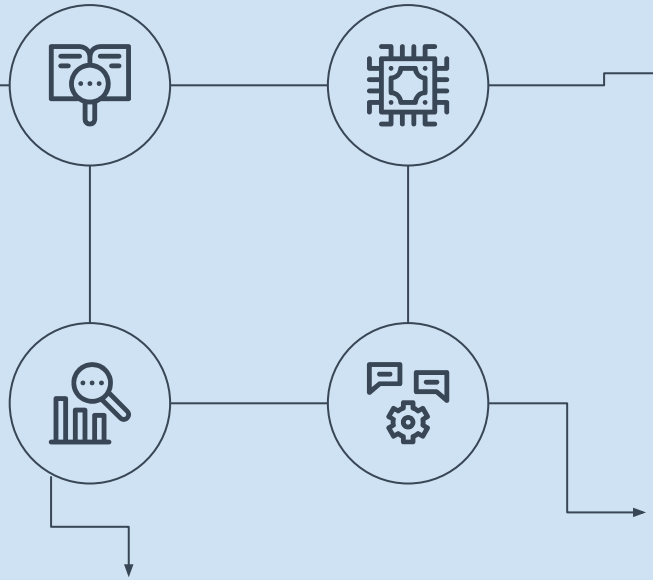
Cloud-Native Deployments ☁️🚀

Transition to scalable, cloud-native platforms (e.g., AWS, Azure) for global accessibility.

Future Aspects and Further enhancements

Voice Cloning and Sentiment Generation 👥🔄❤️

Use voice synthesis for generating audio summaries in a voice similar to the dominant speaker in the audio.
Allow the summary to reflect the emotional tone of the conversation.



GANs for Speech Enhancement🤖🔧

Train Generative Adversarial Networks (GANs) to enhance audio quality before transcription and emotion analysis, especially for noisy inputs.

Domain-Specific Vocabulary 📚🔍

Train the model on domain-specific audio data (e.g., healthcare, legal, customer service) to improve recognition of specialized terms or jargon.

Deeper Emotional Insights🧠💡😌

Provide detailed emotional trajectories over time.
Add more nuanced emotional categories and subcategories, such as mixed emotions or mood shifts.

Behavioral Analytics

Use the system for psychological studies, employee training, or customer behavior analysis by correlating emotion patterns with actions.

Leveraging Transfer Learning

Leverage pre-trained models to apply existing knowledge to new tasks, reducing the need for large datasets and speeding up development.

Implement Continuous Learning and Feedback Loops

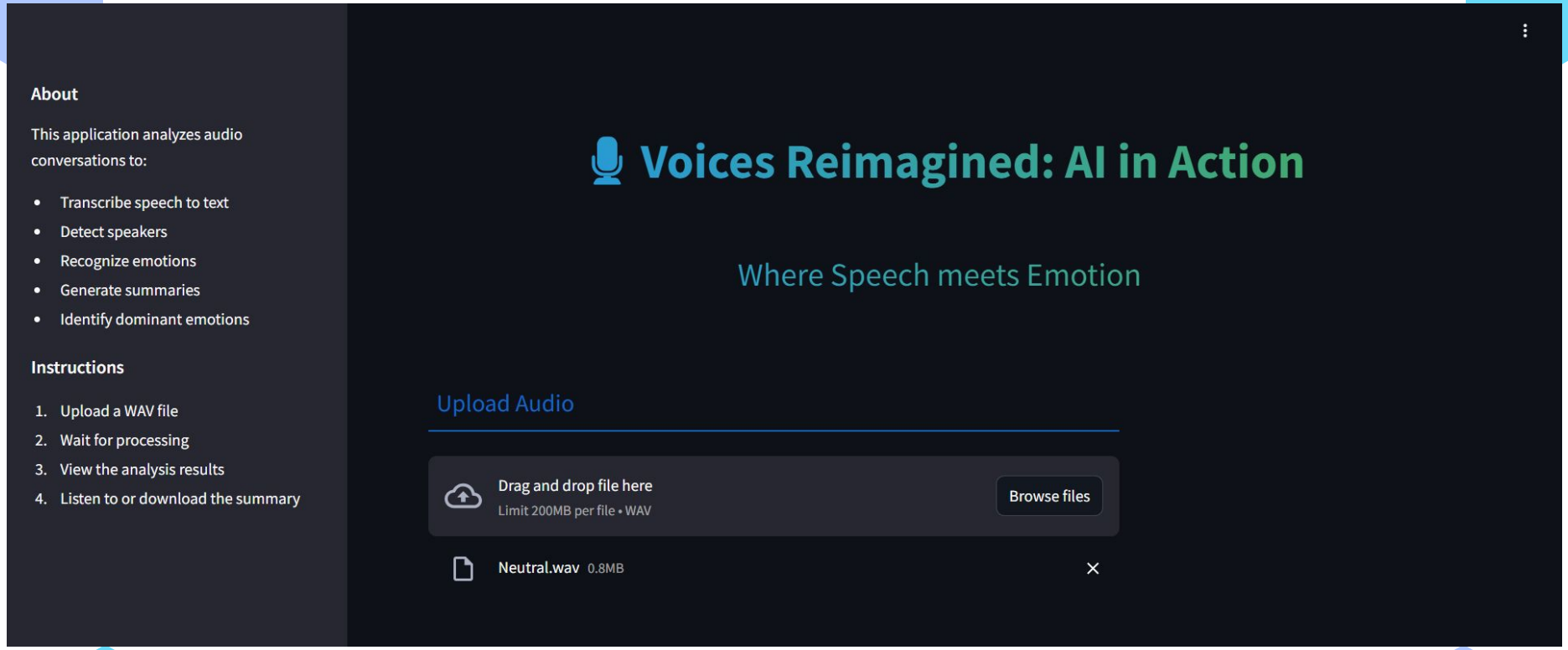
Establish mechanisms for ongoing model updates based on new data and user feedback to maintain and improve performance over time.

Utilize Hardware Acceleration

Employ high-performance GPUs and optimized hardware to accelerate model training and inference, ensuring efficient processing.

Prediction and Output

INPUT


About
This application analyzes audio conversations to:
Transcribe speech to text
Detect speakers
Recognize emotions
Generate summaries
Identify dominant emotions
Instructions
1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary
Voices Reimagined: AI in Action
Where Speech meets Emotion
Upload Audio
Drag and drop file here
Limit 200MB per file • WAV
Browse files
Neutral.wav 0.8MB

OUTPUT

About

This application analyzes audio conversations to:

- Transcribe speech to text
- Detect speakers
- Recognize emotions
- Generate summaries
- Identify dominant emotions

Instructions

1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary

Analysis Results

📝 Transcribed Dialogue with Emotions

SPEAKER_00 (['neu']): CAN YOU BREAK IT DOWN FOR US WHAT IS THIS WHAT IS SELF ATTENTION
SPEAKER_01 (['neu']): SO IMAGINE YOURE AT A NOISY PARTY AND YOU'RE TRYING TO FOLLOW LIKE THREE DIFFERENT CONVERSATIONS AT THE

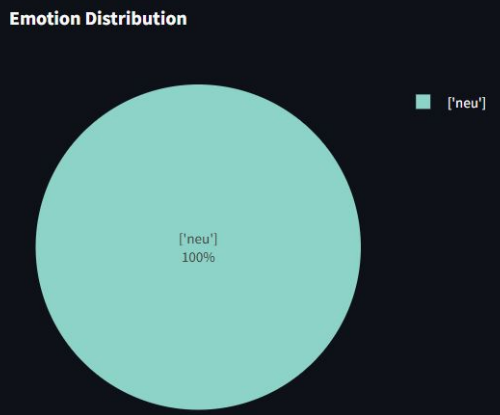
📋 Summary

Self-attention is a technique used in neural networks to enable them to focus on specific parts of their input, allowing them to better process and comprehend complex information.

🎭 Dominant Emotion

['NEU']

📊 Emotion Distribution

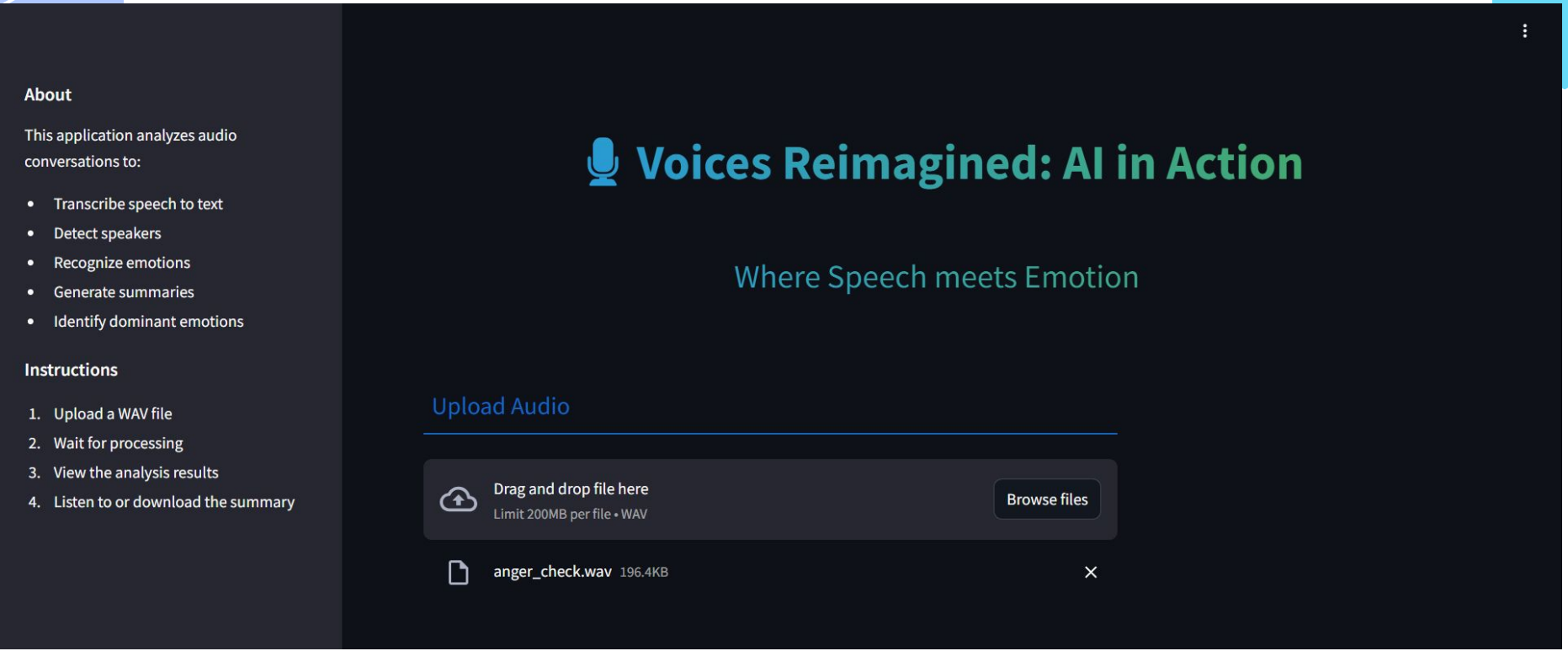


Summary Audio

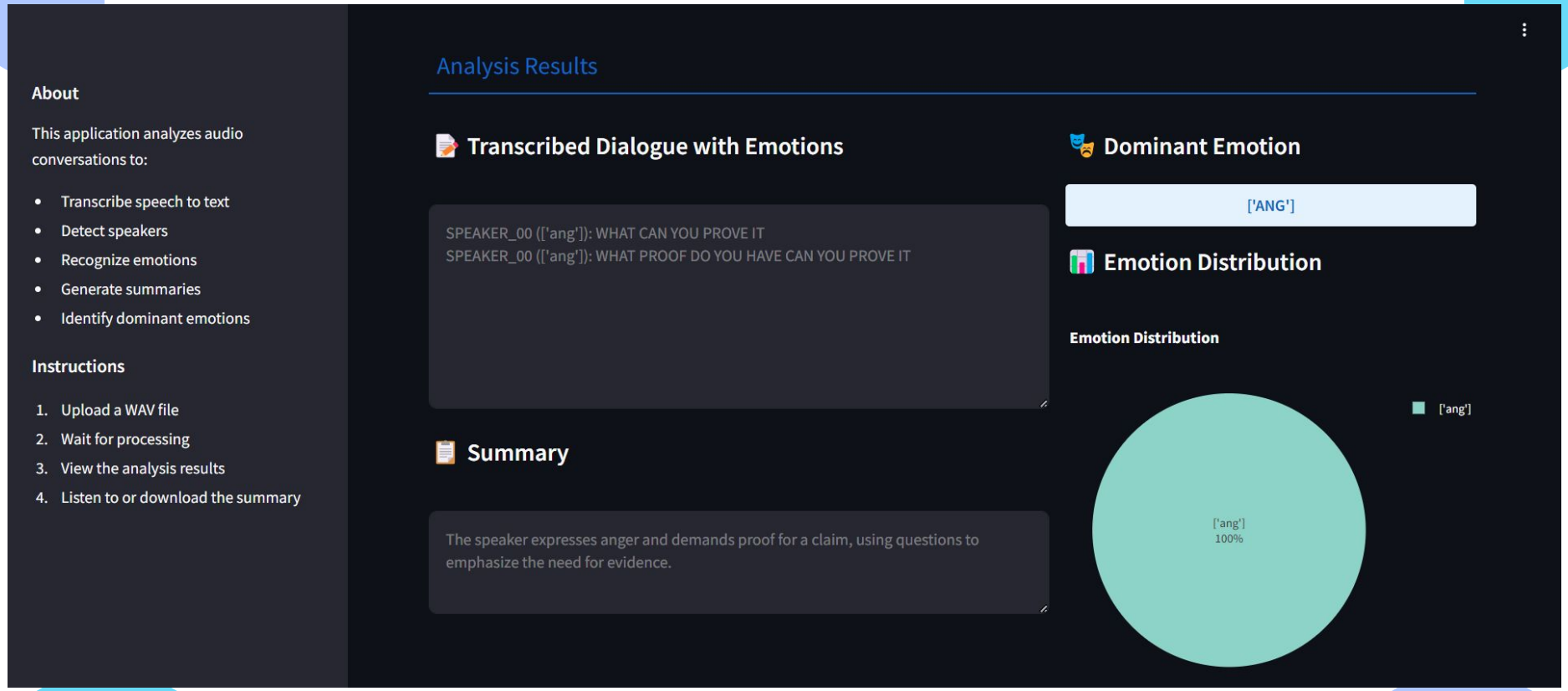
0:00 / 0:12

Download Summary Audio

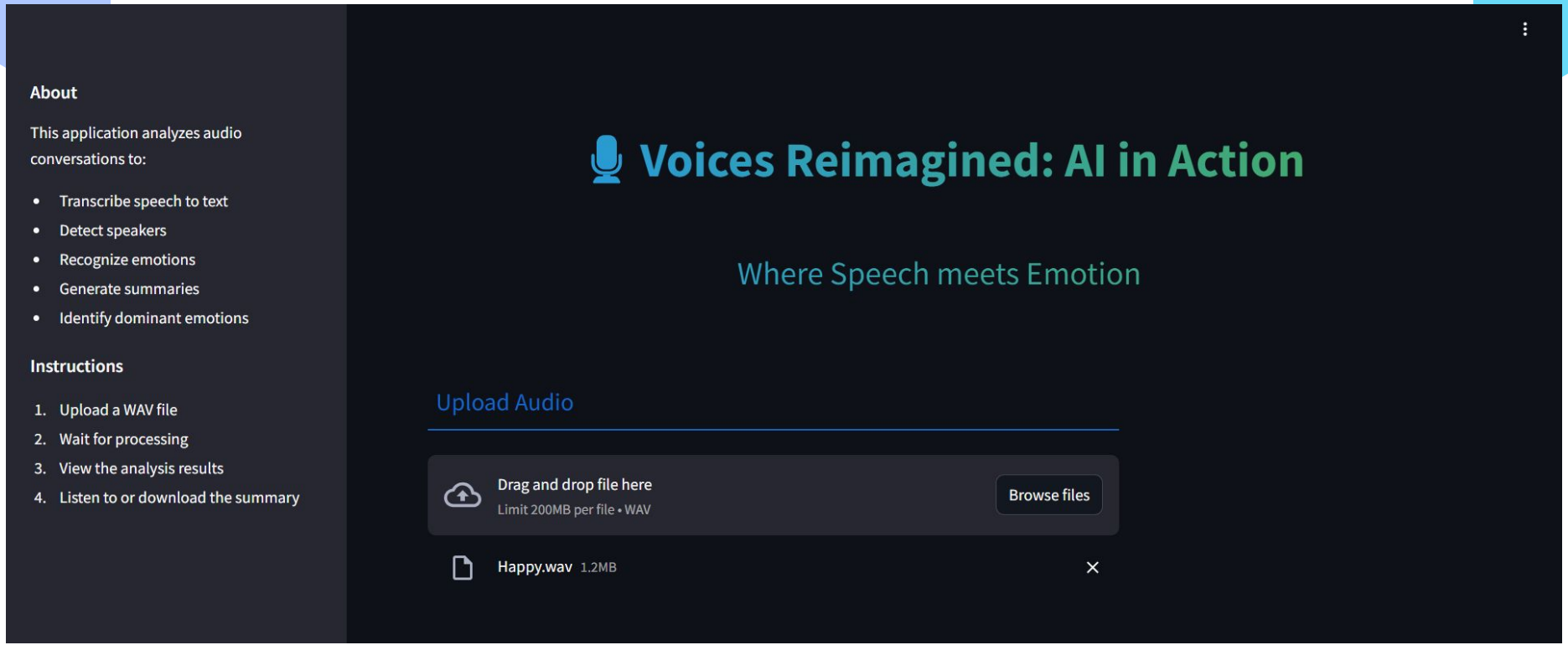
INPUT


About
This application analyzes audio conversations to:
Transcribe speech to text
Detect speakers
Recognize emotions
Generate summaries
Identify dominant emotions
Instructions
1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary
Voices Reimagined: AI in Action
Where Speech meets Emotion
Upload Audio
Drag and drop file here
Limit 200MB per file • WAV
Browse files
anger_check.wav 196.4KB

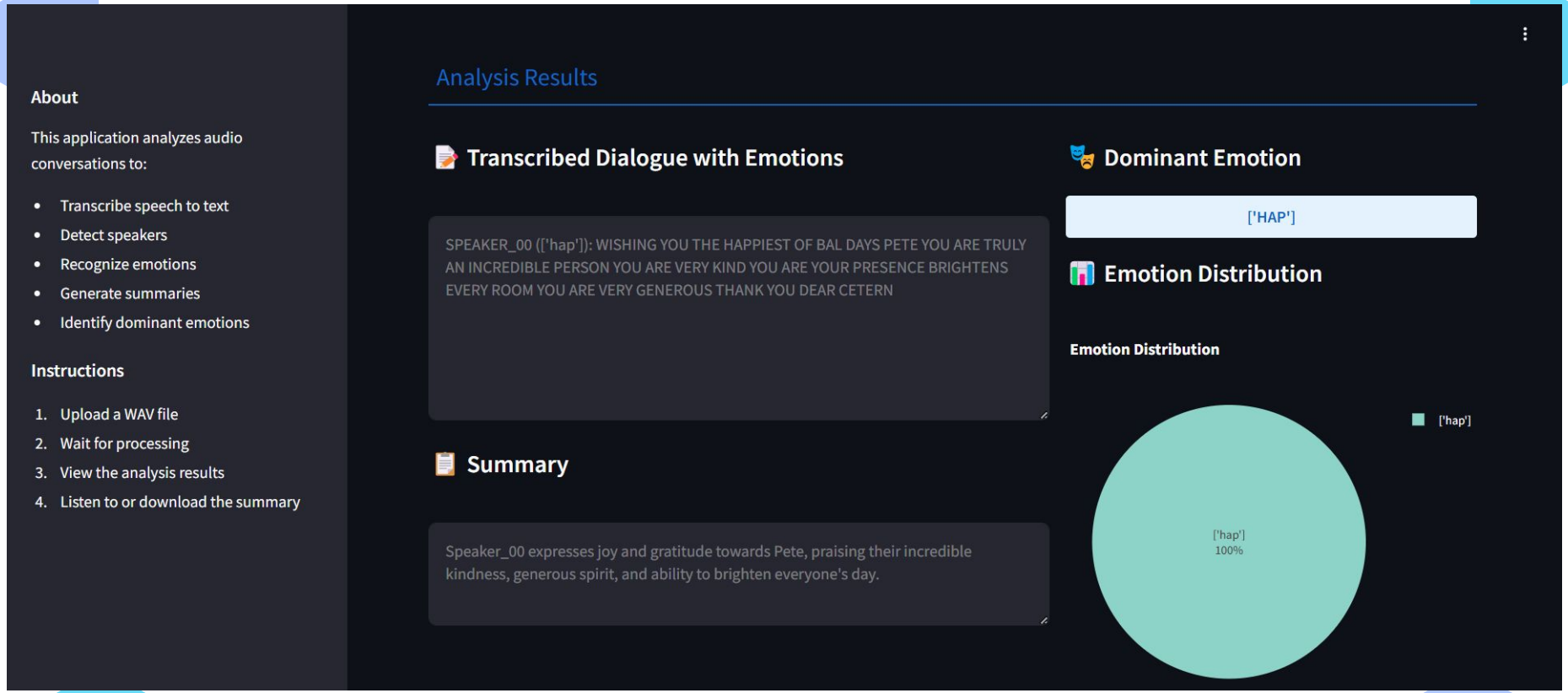
OUTPUT


About
This application analyzes audio conversations to:
Transcribe speech to text
Detect speakers
Recognize emotions
Generate summaries
Identify dominant emotions
Instructions
1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary
Analysis Results
Transcribed Dialogue with Emotions
SPEAKER_00 (['ang']): WHAT CAN YOU PROVE IT
SPEAKER_00 (['ang']): WHAT PROOF DO YOU HAVE CAN YOU PROVE IT
Summary
The speaker expresses anger and demands proof for a claim, using questions to emphasize the need for evidence.
Dominant Emotion
['ANG']
Emotion Distribution
Emotion Distribution
['ang']
['ang']
100%

INPUT


About
This application analyzes audio conversations to:
Transcribe speech to text
Detect speakers
Recognize emotions
Generate summaries
Identify dominant emotions
Instructions
1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary
Voices Reimagined: AI in Action
Where Speech meets Emotion
Upload Audio
Drag and drop file here
Limit 200MB per file • WAV
Browse files
Happy.wav 1.2MB

OUTPUT


About
This application analyzes audio conversations to:
Transcribe speech to text
Detect speakers
Recognize emotions
Generate summaries
Identify dominant emotions
Instructions
1. Upload a WAV file
2. Wait for processing
3. View the analysis results
4. Listen to or download the summary
Analysis Results
Transcribed Dialogue with Emotions
SPEAKER_00 (['hap']): WISHING YOU THE HAPPIEST OF BAL DAYS PETE YOU ARE TRULY AN INCREDIBLE PERSON YOU ARE VERY KIND YOU ARE YOUR PRESENCE BRIGHTENS EVERY ROOM YOU ARE VERY GENEROUS THANK YOU DEAR CETERN
Summary
Speaker_00 expresses joy and gratitude towards Pete, praising their incredible kindness, generous spirit, and ability to brighten everyone's day.
Dominant Emotion
['HAP']
Emotion Distribution
Emotion Distribution
['hap']
['hap']
100%

Thank You!