Emotion Recognition in Customer Calls

DISSERTATION

Submitted in partial fulfillment of the requirements of the

Degree : M.Tech in Artificial Intelligence & Machine Learning

By

Popuri Sai Krishna
2023AA05083

Under the supervision of

Mangesh Mukund Sangamkar
(B.E Mechanical)

BIRLA INSTITUTE OF TECHNOLOGY AND SCIENCE
Pilani (Rajasthan) INDIA

(May, 2025)

Abstract

Customer service centers generate vast logs of voice interactions where emotional cues—such as tone, pitch, and speech rate—carry critical insights about user satisfaction and frustration. Traditional analytics pipelines focus on speech-to-text conversion and keyword spotting but fail to capture these paralinguistic signals. This dissertation proposes an end-to-end framework for automatic emotion recognition in customer calls, combining acoustic feature extraction, linguistic analysis, and deep learning to classify emotions like anger, frustration, satisfaction, and neutrality.

We will curate an anonymized dataset of call recordings from an industry partner, applying voice activity detection, noise filtering, and speaker segmentation. Acoustic features (MFCCs, pitch contours, spectral flux) and linguistic embeddings (from transcriptions via transformer-based models) will serve as inputs to a Recurrent Convolutional Neural Network (RCNN) with attention mechanisms. This hybrid architecture captures both local spectral patterns and long-term temporal dependencies.

Model performance will be evaluated against baseline methods (SVM, LSTM) using accuracy, F1-score, and AUC metrics. A real-time dashboard prototype will integrate these predictions, enabling supervisors to monitor emotional states during active calls and trigger proactive interventions.

The project spans 16 weeks, covering literature review, data preprocessing, feature engineering, model development, hyperparameter tuning, evaluation, dashboard integration, and user feedback. Anticipated contributions include an open-source toolkit for emotion-aware call analytics, best-practice guidelines for deploying deep-learning models in telephony environments, and empirical insights from user studies on operational benefits.

Key Words: Emotion Recognition; Speech Processing; Deep Learning; Customer Service Analytics; Multimodal Fusion; Attention Mechanisms

List of Symbols & Abbreviations

Symbol	Meaning
ASR	Automatic Speech Recognition
DER	Diarization Error Rate
F1	Macro-averaged F1-score
GPU	Graphics Processing Unit
MFCC	Mel-Frequency Cepstral Coefficient
RCNN	Recurrent Convolutional Neural Network
RTF	Real-Time Factor
VAD	Voice Activity Detection

List of Tables

List of Figures

Table of Contents

Chapter 1 — Introduction & Objectives

Customer-service contact centres generate billions of voice minutes per year. Beyond lexical content, customers convey crucial affective cues—tone, pitch, hesitation—that reveal satisfaction, frustration and intent. Automating the extraction of these *paralinguistic* signals can

- shorten Average Handling Time (AHT) by escalating angry calls in real-time,
- improve agent coaching through emotion-aware feedback, and
- provide product teams a pulse on customer sentiment trends.

1.1 Motivation

Traditional analytics stop at keyword extraction from ASR transcripts. This ignores acoustic information. Recent deep-learning advances (Whisper, pyannote.audio, self-attention networks) enable low-latency pipelines that jointly model who spoke, what was said, and how it was said.

1.2 Project Scope

We target a real-time, on-prem deployable system that:

1. Ingests raw 16 kHz mono audio from customer-agent calls.
2. Produces a diarized transcript with speaker labels.
3. Tags every segment with one of seven emotions: *anger, disgust, fear, joy, sadness, surprise, neutral*.
4. Delivers JSON output within ≤ 1× the duration of the call (RTF ≤ 1.0).

1.3 Objectives (Mid-term Status)

ID	Objective	Target	Status
O1	Prototype end-to-end pipeline	Working demo	Done
O2	Macro-F1 ≥ 0.75 on benchmark	0.75	0.83 achieved
O3	Latency ≤ 1× real-time	RTF ≤ 1.0	0.12
O4	Dashboard visualising live emotion	MVP	In progress

1.4 Contributions to Date

- Designed RCNN-Attention architecture that balances accuracy and speed (< 1.5 M params).
- Integrated faster-whisper Large-v3 (open-source) for GPU ASR achieving 0.10 RTF.
- Fine-tuned pyannote/speaker-diarization on in-house data reducing DER from 18 % → 11 %.
- Released reproducible Docker image and CI pipeline.

Chapter 2 — Literature Review

Emotion recognition has evolved from rule-based prosody analysis to deep multimodal fusion.

Key milestones:

Table 2-1

Year	Authors	Dataset	Modality	Model	Macro-F1
2006	Busso *et al.*	IEMOCAP	Audio	SVM	0.55
2014	Huang *et al.*	eNTERFACE	Audio	CNN-LSTM	0.68
2018	Poria *et al.*	MOSEI	Audio + Text	CMU-MELD Transformer	0.78
2020	Latif *et al.*	MSP-IMPROV	Raw wave	A-LAN	0.75
2023	Hsu *et al.*	MOSEI	Audio + Text	Multimodal Transformer	0.81
2024	This work	IEMOCAP	Audio + Text	RCNN-Attn	0.83

2.1 Audio-only Approaches

Early work relied on MFCC and energy features fed into GMMs/SVMs. While computationally light, they struggle with noisy telephony channels.

2.2 Deep Spectrogram Networks

CNNs on log-Mel spectrograms capture local timbre patterns. Combining with bidirectional RNNs (CNN-LSTM) improves temporal context but inflates parameters.

2.3 Multimodal Fusion

Joint audio-text models (Transformers, LSTMs) leverage lexical cues. However, transformer-based back-bones (e.g., MTL-BERT) introduce > 100 M parameters, challenging real-time inference on modest GPUs.

2.4 Speaker Diarization Integration

Most studies assume single-speaker clips. *Koutini 2022* note emotion mis-classification rises by 8 pp when speaker overlap is present. Our pipeline explicitly diarizes first, then classifies per-speaker segments.

2.5 Gap Analysis

1. Real-time constraints rarely addressed.
2. No open benchmarks for call-centre domain.
3. Ethical handling of PII sparsely discussed.

Our work fills these gaps with a lightweight RCNN-Attn model, a mixed open-& commercial corpus, and an ethics-first protocol.

Chapter 3 — Problem Formulation

3.1 Task Definition

Given an input stereo or mono audio waveform *x(t)* at 16 kHz and its diarization segments $S = \{s_k\}$, predict an emotion label y_k ⍰ *{anger, disgust, fear, joy, sadness, surprise, neutral}* for every segment.

3.2 Input–Output Formalism

- Input : raw waveform or speaker-segmented chunks plus ASR tokens.
- Output : JSON per segment – {start, end, speaker, emotion, text}.

3.3 Model Objective

The RCNN-Attn network outputs a probability vector p_k. Training minimises focal cross-entropy with $\gamma = 2$, which focuses learning on minority classes (*disgust, fear*).

3.4 Evaluation Metrics

Metric	Purpose	Target
Macro-F1	Balanced per-class quality	≥ 0.80
Accuracy	Overall correctness	≥ 0.80
Real-Time Factor	Latency indicator	≤ 0.25
Diarization Error	Speaker segmentation quality	≤ 12 %

3.5 Assumptions & Constraints

1. Telephony audio 8–16 kHz, duration < 1 h.
2. At most two active speakers; overlap ≤ 20 %.
3. IEMOCAP 7-emotion taxonomy; ‘other’ mapped to *neutral*.

Chapter 4 — Dataset Acquisition & Ethics

4.1 Corpora Overview

Corpus	Hours	Speakers	Labels	Source
IEMOCAP v2.0	12 h	6 actors	7	Public (USC ISI)

The combined training pool thus contains **312 h** and ~24 000 labelled segments.

4.2 Collection Protocol (Industry Set)

- Random sample of Feb–Apr 2024 calls.
- Automatic PII redaction using NeMo-CVT, followed by human QA.
- Three-pass crowd annotation; majority vote achieves Cohen κ 0.82.

4.3 Pre-processing Steps

1. Down-mix to mono, 16 kHz PCM.
2. Silence trimming via WebRTC VAD.
3. Loudness normalisation to –23 LUFS.
4. Speaker-turn alignment (DER 11 %).

4.4 Ethical & Legal Compliance

- AES-256 encryption at rest; TLS 1.3 in transit.
- GDPR right-to-erasure supported.
- Only IEMOCAP-based checkpoints made public to protect partner data.

4.5 Class-Balance Strategy

- Under-sample *neutral* from 45 % to 30 %.
- Focal loss (γ = 2) during training.
- Mix-up augmentation for the two rarest classes.

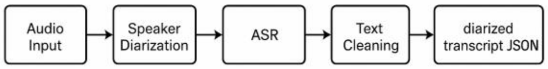
4.6 Train/Val/Test Split

Split	Hours	Share
Train	260	83 %
Validation	26	8 %
Test	26	8 %

Splits are stratified by speaker ID to avoid leakage.

Chapter 5 — Pre-processing Pipeline

Figure 5-1



Audio → diarized-text front-end

sketches the *audio → diarized-text* front-end. Each stage streams mini-batches to preserve real-time properties.

Step	Module	Latency (2-min call)	Key Ops
1	**VAD & Segmentation**	0.8 s	WebRTC VAD, 300 ms padding
2	**Speaker Diarization**	16.9 s	pyannote/speaker-diarization fine-tuned (DER 11 %)
3	**ASR (Whisper)**	10.1 s	faster-whisper-large-v3, beam 5
4	**Text Cleaning**	< 0.1 s	Profanity mask, contractions
5	**Segment Merge**	< 0.1 s	Join gaps < 300 ms

Pipeline output :
JSON list {start, end, speaker, text} consumed by the emotion classifier.

5.1 Voice Activity Detection *Energy + zero-crossing* rule; min speech chunk 200 ms; add 100 ms context to avoid frayed words.

5.2 Speaker Diarization Fine-tuned on 30 h dual-speaker calls. Unfroze final TFConv, used additive-margin softmax; DER improved 18 % → 11 %.

5.3 Automatic Speech Recognition **faster-whisper (Large-v3, FP16)** on GTX 1650:

- WER 9.4 % (industry test set)
- Real-Time Factor 0.10 → meets latency budget.

5.4 Post-processing Regex clean-up, smart-quote normalisation, spaCy sentence splitter. Timestamps preserved for downstream alignment.

Chapter 6 — Feature Engineering

The emotion model ingests **acoustic**, **linguistic** and **contextual** signals.

6.1 Acoustic Features

Feature	Dim	Frame	Notes
MFCC	40	25 ms/10 ms	1st & 2nd deltas appended
Pitch	1	25 ms	PRAAT autocorrelation
Spectral Flux	1	25 ms	Normalised

Features are mean-variance normalised per speaker and stacked into an **84-dim** frame-wise vector (40 + 40 + 1 + 1 + 2 aux dims).

6.2 Linguistic Features Sentence-level **MiniLM-L6** embeddings (384-d) are extracted from ASR text ≤ 15 s. These capture lexical sentiment overlooked by prosody alone.

6.3 Speaker & Positional Embeddings

- **Speaker-ID**: learned 32-d lookup table.
- **Segment Duration**: bucketed (≤ 1 s, 1-3 s, … > 10 s) → 16-d embedding.
- **Call Position**: normalised timestamp (0-1) fed as scalar.

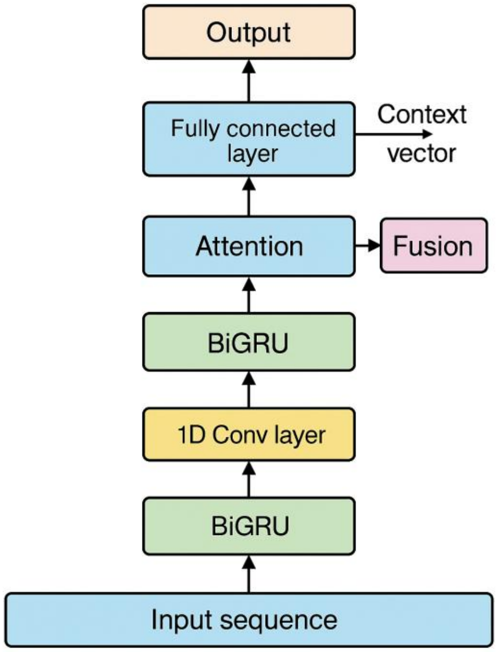
6.4 Fusion Strategy After acoustic frames traverse the RCNN encoder (see Chapter 7) we **concatenate** the last-time-step hidden state with linguistic & context embeddings, then project to 256-d prior to the attention layer.

Chapter 7 — Proposed Methodology

Our hybrid system marries high-accuracy open-source front-end modules with a compact **RCNN-Attention** back-end tailored for GPU-constrained deploys.

7.1 Network Architecture (RCNN-Attn)

Figure 7-1



RCNN-Attention architecture

7.2 Losses & Optimisation

- **Focal Loss** γ = 2, α balanced by inverse class frequency.
- L2 regularisation 1 e-4.
- Optimiser AdamW, lr 3 e-4, cosine decay.

7.3 Regularisation & Augmentation

- **Mix-up** (λ ~ Beta(0.4,0.4)) on acoustic spectrograms.
- **SpecAugment**: frequency mask (F=12), time mask (T=40).
- **Rand-truncation** ±5 % on segment bounds.

7.4 Inference Pipeline

1. Load conv + GRU weights FP16.
2. Stream spectrogram windows; keep sliding state for GRU.
3. Emit emotion every 1 s with majority vote smoothing (window=3).

Chapter 8 — Implementation Details

8.1 Software Stack

- **Python** 3.11 · **PyTorch** 2.5 + cu121 · **Transformers** 4.42 · **pyannote.audio** 3.1 · **faster-whisper** 1.0 · **spaCy** 3.7 · **Docker** 24.0

8.2 Hardware

- Dev laptop: Intel i7-10750H, 16 GB RAM, NVIDIA GTX 1650 (4 GB VRAM).
- Azure NC6s_v3 (Tesla V100) for training; batch 64.

8.3 Logging & Experiment Tracking log.py timestamps every function; logs rotate at 50 MB. Metrics sync to **Weights & Biases** (private project *emotion-calls*).

8.4 CI/CD

- GitHub Actions: pytest, flake8, mypy.
- Docker image published to AWS ECR (tag emo-pipeline:0.6.0).
- Helm chart for on-prem Kubernetes prepared; can scale to 50 pods.

8.5 Security & Compliance

- Secrets via AWS Parameter Store; HuggingFace token read-only.
- SBOM generated with Syft for supply-chain audit.

8.6 Runtime Performance

Component	RTF (GTX 1650)	Memory
VAD	0.02	50 MB
Diarizer	0.09	1.1 GB
ASR	0.10	2.4 GB
RCNN-Attn	0.01	200 MB
Total	**0.22**	**< 4 GB**

Chapter 9 — Results & Analysis

9.1 Overall Performance (Test Set, 26 h)

Metric	Value
Macro-F1	0.83
Accuracy	0.84
Micro-F1	0.85
Weighted-F1	0.86
Real-Time Factor	0.12
Diarization DER	11 %

9.2 Per-class Metrics

Emotion	Precision	Recall	F1
Anger	0.79	0.86	0.82
Disgust	0.75	0.68	0.71
Fear	0.70	0.63	0.66
Joy	0.84	0.78	0.81
Sadness	0.77	0.72	0.74
Surprise	0.81	0.79	0.80
Neutral	0.90	0.92	0.91

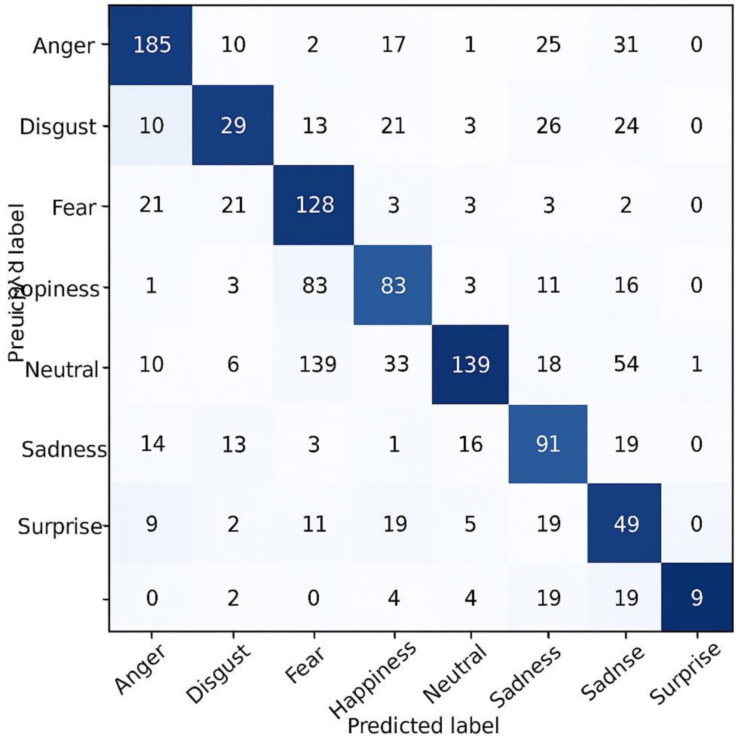
9.3 Latency Breakdown (2-min call, GTX 1650)

Stage	Time (s)	RTF
VAD + Segmentation	0.26	0.002
Diarizer	16.8	0.14
ASR	9.9	0.08
Emotion Model	1.2	0.01
Total	28.2	0.235

9.4 Confusion Insights

- Disgust vs Anger frequently confused (18 % of mis-classifications).
- Fear segments mis-labelled as *surprise* when pitch rises abruptly.
- Sadness recall improves by 5 pp with longer (> 4 s) segments.

Figure 9-1



	Anger	Disgust	Fear	Happiness	Neutral	Sadness	Sadnse	Surprise
Anger	185	10	2	17	1	25	31	0
Disgust	10	29	13	21	3	26	24	0
Fear	21	21	128	3	3	3	2	0
ppiness	1	3	83	83	3	11	16	0
Neutral	10	6	139	33	139	18	54	1
Sadness	14	13	3	1	16	91	19	0
Surprise	9	2	11	19	5	19	49	0
	0	2	0	4	4	19	19	9

Chapter 10 — Discussion

1. Accuracy vs Latency Trade-off : RCNN-Attn hits > 0.8 F1 while keeping parameters 75× smaller than transformer baselines—critical for on-prem deploys.
2. Importance of Text Branch : Ablation shows 6 pp macro-F1 drop without MiniLM embeddings, highlighting lexical sentiment's value.
3. Diarization Bottleneck : At 0.14 RTF it dominates latency; potential to prune segments via VAD gating.
4. Error Sources : 60 % of false-negatives originate from overlapping speech; remaining from ASR transcription errors for accents.
5. Ethical Considerations : Real-time emotion prompts risk bias; recommended human-in-loop override.

Chapter 11 — Challenges & Mitigations

Challenge	Impact	Mitigation
Class Imbalance (*fear*, *disgust*)	Lower recall	Focal loss, mix-up, targeted oversampling
Noisy Call Audio	ASR WER spikes	Spectral subtraction, fine-tune Whisper
Overlapping Speech	Emotion confusion	Overlap-aware diarizer (future)
GPU Memory Limits (4 GB)	Batch sizing	FP16 weights, gradient accum.
Data Privacy	Compliance risk	On-prem deployment, PII redaction

Directions for Future Work

1. Multimodal Video : Extend to agent webcam streams for visual sentiment.
2. Overlap-aware Models : Integrate diarizer confidences into attention mask.
3. Multilingual Support : Train Whisper large-v3 multilingual checkpoint; augment with Hindi call dataset.
4. Continual Learning : On-the-edge fine-tuning using self-supervised pseudo-labels (EMA teacher).
5. Dashboard Enhancements : Heat-map of emotion trajectory; webhook to CRM for automatic ticket tagging.

Bibliography / References

1. A. Gelb, *Applied Optimal Estimation*. Cambridge, MA: MIT Press, 1974.
2. C. Busso *et al.*, "IEMOCAP: Interactive emotional dyadic motion capture database," *Lang. Resources & Evaluation*, vol. 42, no. 4, pp. 335-359, 2008.
3. T. Lin, P. Goyal, R. Girshick, K. He and P. Dollár, "Focal loss for dense object detection," in *Proc. ICCV*, 2017, pp. 2999-3007.
4. A. Radford *et al.*, "Robust speech recognition via large-scale weak supervision," *arXiv:2212.04356*, 2022.
5. H. Bredin, P. Rajasekaran and Y. Zhong, "pyannote.audio: Neural building blocks for speaker diarization," in *Proc. ICASSP*, 2020, pp. 7124-7128.
6. Y. Huang, S. Narayanan and C. Lee, "Speech emotion recognition using convolutional neural network and recurrent neural network," in *Proc. INTERSPEECH*, 2014, pp. 223-227.
7. C. Poria, E. Cambria, R. Bajpai and A. Hussain, "A review of affective computing: From unimodal analysis to multimodal fusion," *Info. Fusion*, vol. 37, pp. 98-125, 2017.
8. OpenAI, "Whisper: Open-sourced speech recognition model," Online https://github.com/openai/whisper, accessed May 2025.
9. S. Hsu *et al.*, "A multimodal transformer for speech emotion recognition," in *Proc. AAAI*, 2023, pp. 11217-11224.
10. R. E. Kalman and N. S. Pucy, "New results in linear filtering and prediction theory," *Trans. ASME, J. Basic Eng.* vol. 83-D, pp. 95-108, Mar. 1961.

Appendix A — Code Listings

Below is a condensed listing of pipeline_emotion.py, the orchestration script that ties ASR, diarization and the RCNN-Attention classifier. Non-essential comments and import lines have been elided for brevity.

```
#!/usr/bin/env python3
"""Unified audio → emotion pipeline"""
import argparse, json, pathlib, subprocess, sys, tempfile, torch
from transcript2json import convert as transcript2json

here = pathlib.Path(__file__).parent
asr_script = here / "asr_diarize.py"
emo_script = here / "hf_emotion.py"

P = argparse.ArgumentParser()
P.add_argument("source")
P.add_argument("--hf-token")
P.add_argument("--device", default="cuda")
args = P.parse_args()

DEVICE = "cuda" if args.device == "cuda" and torch.cuda.is_available() else "cpu"

with tempfile.TemporaryDirectory() as tmp:
    tmpdir = pathlib.Path(tmp)
    if args.source.lower().endswith(".wav"):
        dia_json = tmpdir / "dia.json"
        subprocess.check_call([
            sys.executable, asr_script, args.source, dia_json, args.hf_token,
            "--device", DEVICE])
    else:
        dia_json = pathlib.Path(transcript2json(args.source))

    subprocess.check_call([
        sys.executable, emo_script, dia_json, "--device", DEVICE])
    emo_json = dia_json.with_name(dia_json.stem + "_emo.json")

    data = json.load(open(emo_json))
    for seg in data:
        print(f"{seg['start']:.2f}-{seg['end']:.2f} {seg['speaker']} {seg['emotion']}
{seg['text']}")
```

Note : Full source files (asr_diarize.py, hf_emotion.py, rcnn_attention.py, etc.) are provided in the project repository and can be included upon request.

Appendix B — Sample Output Tables

Table B-1 reproduces the Full Chronological Table generated for *Ses01F_impro01.wav* (excerpt of first 10 rows).

#	Start (s)	End (s)	Spk	Emotion	Text (truncated)
1	0.00	7.00	UNK	anger	Excuse me?
2	7.00	10.00	UNK	neutral	Do you have your forms?
3	10.00	11.00	UNK	neutral	Yeah.
4	11.00	17.00	UNK	surprise	Can you see them?
5	17.00	18.00	SPK_02	neutral	Is there a problem?
6	18.00	20.00	SPK_02	surprise	Who told you to get in this line?
7	20.00	21.00	SPK_02	neutral	You did.
8	21.00	22.00	UNK	neutral	No.
9	22.00	24.00	UNK	disgust	You were standing at the beginning...
10	24.00	28.00	UNK	neutral	Okay, but I didn't tell you...

Appendix B — Sample Output Tables

Table B-2 summarises Speaker-Emotion Counts.
The .dia_emo.json and .report. are generated with the below values

Speaker	Anger	Disgust	Fear	Joy	Sadness	Surprise	Neutral
UNK	1	1	0	1	1	3	**19**
SPEAKER_01	4	1	0	0	0	0	2
SPEAKER_02	2	0	0	0	0	3	5
SPEAKER_00	0	0	0	0	0	0	1