A logo with a compass

AI-generated content may be incorrect.

UNDERGRADUATE FINAL YEAR PROJECT REPORT

Speech Emotion Recognition Using Deep Learning: A Foundation for Emotion-Aware Conversational Systems

TRINH HOANG ANH HONG

Bachelor of Science with Honors in Computing

REG No: 001323807

SUBMISSION DAY: 29/11/2025

Table of Contents

[1. Introduction 2](#_Toc215182563)

[1.1. Project Subject 2](#_Toc215182564)

[1.2. Project Objective 2](#_Toc215182565)

[1.3. Project Plan 2](#_Toc215182566)

[1.4. Project Outcome 4](#_Toc215182567)

[1.5. Project Evaluation 5](#_Toc215182568)

[2. Literature Review 6](#_Toc215182569)

[2.1. Domain Knowledge 6](#_Toc215182570)

[2.1.1 Speech Emotion Recognition (SER) 6](#_Toc215182571)

[2.1.2. Emotions Expression in Human Speech 6](#_Toc215182572)

[2.1.3. The Acoustic Basis of Emotion 7](#_Toc215182573)

[2.2. Literature Research in Speech Emotion Recognition 7](#_Toc215182574)

[2.2.1. Traditional Approaches 7](#_Toc215182575)

[2.2.2. Deep Learning-Based Approaches 8](#_Toc215182576)

[2.2.3. Multi-Dataset and Cross-Dataset Studies 8](#_Toc215182577)

[2.3. Feature Extraction Techniques for SER 9](#_Toc215182578)

[3. Technology and Tools 10](#_Toc215182579)

[4. Software Product Requirements 11](#_Toc215182580)

[4.1. Data Preprocessing and Organization 11](#_Toc215182581)

[4.3. Feature Extraction and Analysis 11](#_Toc215182582)

[4.4. Deep Learning Model 12](#_Toc215182583)

[4.5. Graphical User Interface (GUI) Development 13](#_Toc215182584)

[5. Review of Software Development Methodologies 13](#_Toc215182585)

[5.1. Waterfall 13](#_Toc215182586)

[5.2. Spiral 14](#_Toc215182587)

[5.3. RAD (Rapid Application Development) – Prototyping 15](#_Toc215182588)

[5.4. Agile 15](#_Toc215182589)

[5.5. Selection of Software Development Methodology 16](#_Toc215182590)

[6. Software Product Requirements 16](#_Toc215182591)

[6.1. Workflow Overview 16](#_Toc215182592)

[6.2. Data Collection 17](#_Toc215182593)

[6.3. Data Preprocessing and Organization 18](#_Toc215182594)

[6.3.1. RAVDESS Filename Convention 18](#_Toc215182595)

[6.3.2. Exploratory Data Analysis 19](#_Toc215182596)

[6.3.3. Determining Significant Frequency Ranges 21](#_Toc215182597)

[6.3.4. Preprocessing 24](#_Toc215182598)

[6.3.5. Batch preprocessing and dataset organization 26](#_Toc215182599)

[6.4. Feature Extraction and Analysis 26](#_Toc215182600)

[6.4.1. Raw waveform representation 26](#_Toc215182601)

[6.4.2. MFCC Extraction 27](#_Toc215182602)

[6.4.3. Log-Mel Spectrogram 27](#_Toc215182603)

[6.5. Machine Learning and Deep Learning Implementation and Evaluation 28](#_Toc215182604)

[6.5.1. Feature-Specific Baseline Experiments 28](#_Toc215182605)

[6.5.2. Spectrogram-Based CNN Models 30](#_Toc215182606)

[6.5.3. CNN Hybrid Model for Spectrogram 33](#_Toc215182607)

[6.5.4. Sentiment-Level Classification with CNN–RNN, CNN–LSTM and CNN–ResNet 34](#_Toc215182608)

[6.5.5. Transfer Learning and Cross-Dataset Models 39](#_Toc215182609)

[6.6. GUI Implementation 46](#_Toc215182610)

[7. Conclusion 47](#_Toc215182611)

[7.1. Knowledge Gained 47](#_Toc215182612)

[7.2. Result 48](#_Toc215182613)

[7.3. Limitation 48](#_Toc215182614)

[7.4. Further Development 48](#_Toc215182615)

[Reference 49](#_Toc215182616)

# 1. Introduction

## 1.1. Project Subject

This project focuses on Speech Emotion Recognition (SER), a subfield of affective computing that aims to identify human emotions from vocal signals. The primary goal is to develop a deep learning-based system capable of detecting and classifying emotions such as happiness, anger, sadness, calm, and neutrality from speech.

The motivation behind this work is to enable machines to interpret emotional cues embedded in speech, which is essential for creating more natural and empathetic human–machine interactions. By analysing acoustic features and leveraging advanced neural architectures, this project establishes a robust foundation for emotion-aware technologies.

Although the long-term vision includes integrating this capability into conversational agents and social robots, the current implementation is limited to speech-based emotion detection using spectrogram representations and CNN-based models.

## 1.2. Project Objective

**Primary Goal:**

Develop an intelligent system capable of detecting and classifying human emotions through speech, forming the foundation for emotion-aware conversational and robotic systems.

**Secondary Goal:**

* Design and implement a speech emotion recognition (SER) model that classifies emotional states such as happiness, sadness, anger, calm, neutrality.
* Extract and analyze features from speech to train and evaluate deep learning models for emotion detection.
* Evaluate the model’s performance through emotion recognition accuracy and confusion matrix analysis.
* Discuss probable future developments, including integration with a physical robot and potential applications in areas such as mental health support, customer service, and social robotics.

## 1.3. Project Plan

|  |  |  |
| --- | --- | --- |
| **Weeks** | **Tasks** | **Description** |
| Weeks 1 | Problem Definition and Planning | Defined the research problem: Speech Emotion Recognition (SER) using audio signals. Established the project goal: classify emotions from speech using deep learning methods on different features. Reviewed literature on emotion recognition and related work. |
| Weeks 2 | Dataset Preparation | Collected four open-source datasets (RAVDESS, TESS, SAVEE, CREMA-D) and merged them into a unified dataset. Standardized filename conventions and metadata verified sample consistency and audio formats. Final dataset contained approximately 13174 samples after merging. |
| Weeks 3-4 | Data Exploration and Analysis | Inspected dataset structures and class distribution (identified emotional imbalance). Analyzed distribution of the dataset. Exploring the features such as waveform, MFCC, frequency, time frequency. |
| Weeks 5-8 | Feature Exploration and Preprocessing | Experimented with multiple feature extraction techniques. Tested statistical features (mean, std) and frequency range analyses. Evaluated log-Mel spectrograms with and without sliding windows (no overlap / overlap). Finalized preprocessing: trimming, denoising, normalization, and padding. Followed by Extracted log-Mel spectrogram features with parameters. applied data augmentation with 10% overlapping windows to increase dataset diversity and saved processed spectrograms for model training. |
| Weeks 11-12 | Model Training and Architecture Experiments | Implemented and compared multiple architectures: CNN, RNN–CNN, and LSTM–CNN hybrids. Addressed class imbalance by grouping emotions into three sentiment categories: Positive, Moderate, and Negative. Trained and evaluated models using validation accuracy and confusion matrices. |
| Weeks 13-14 | Advanced Experiments with FFT and Deep CNN Models | Tested FFT-based spectrogram representations and trained advanced CNN backbone (ResNet) and other variants to improve emotion classification accuracy. Tuned hyperparameters and optimized training pipeline. |
| Weeks 15 | Evaluation and Result Analysis | Compared model performance using metrics including accuracy, F1-score, and loss curves. Generated visualizations such as ROC curves and confusion matrices. |
| Weeks 16 | Pre-trained model integration | Experimented with Wav2Vec2 using raw audio from the four datasets (RAVDESS, TESS, SAVEE, CREMA-D). Evaluated generalization by training on three datasets and testing on the remaining one. |
| Weeks 17-18 | Sentiment Grouping and Feature Enhancement | Added a sentiment grouping layer to categorize model outputs into three sentiment classes: Negative, Moderate, and Positive. Conducted additional experiments by applying low-pass and high-pass filters to extract emotion-relevant features from low and high frequency bands. Trained CNN models on filtered spectrograms for comparative analysis. |
| Weeks 19 | GUI Prototype Development | Developed a lightweight GUI prototype for real-time emotion detection. |
| Weeks 20 | Final Report and Documentation | Compiled project findings, visual results, and comparative analyses. Created diagrams and summaries, finalized report, and prepared presentation slides for submission. |

## 1.4. Project Outcome

* A functional SER system capable of classifying emotions from human speech signals using CNN-based architecture.
* A unified and preprocessed dataset created from four open-source sources (RAVDESS, TESS, SAVEE and CREMA-D), standardized by file format, labeling scheme, and sampling rate.
* A complete data preprocessing pipeline, including silence trimming, denoising, normalization, and log-Mel/FFT spectrogram feature extraction.
* Implementation and comparation of multiple deep learning architectures (CNN, RNN-CNN, LSTM-CNN and ResNet) to evaluate classification performance and robustness.
* Fine tuning experiments conducted with the pre-trained Wav2Vec2 model for transfer learning and performance benchmarking.
* Visualizations and analytical reports, including accuracy curves, confusion matrices, and ROC curves, summarizing model performance.
* A reproducible workflow for emotion recognition research, providing a foundation for future development of context-aware chatbot systems and emotion-aware robots.

## 1.5. Project Evaluation

**Feasibility:**

* The project is built on reliable frameworks such as TensorFlow, Pytorch, and Librosa, ensuring reproducibility and compatibility with modern machine learning environments.
* The modular design architecture separates emotion recognition and response generation components, supporting incremental testing and improvement of each module.
* The availability of high-quality, open-source datasets enables experimentation with multiple emotion classes and model configurations.
* Log-Mel spectrogram representations provide efficient and information-rich features suitable for both real-time and offline emotion detection.
* The integration of a pre-trained model (Wav2Vec2) with the audio dataset enhances performance and demonstrates transfer learning feasibility.

**Validity:**

* Quantitative validation is conducted using metrics such as accuracy, precision, recall, and F1-score, measure across all emotion categories.
* Cross-validation and test splits are used to ensure model generalization and to reduce overfitting bias
* Comparative experiments across different features and models ensure reliability and replicability of results.

**Technical Evaluation:**

Quantitative Metrics:

* Emotion Classification Accuracy: Measures how accurately the system identifies emotions from speech inputs.
* Precision, Recall, and F1-score: Evaluate the balance between true positive and false negatives for each emotion class.
* Confusion Matrix and ROC Curves: Provide detailed insight into model strengths and misclassification patterns.
* Inference Time: Assesses system efficiency and suitability for potential real-time emotion detection and future chatbot integration.

Qualitative Assessment:

* Feature Visualization: Analyze log-Mel spectrograms, as well as CNN feature maps, to confirm emotional signal regions and feature learning behavior.
* Robustness Testing: Evaluates model stability under different conditions such as background noise, speaker variation, and mixed emotions.

# 2. Literature Review

## 2.1. Domain Knowledge

### 2.1.1 Speech Emotion Recognition (SER)

Speech Emotion Recognition (SER) is a subfield of affective computing concerned with identifying a speaker’s emotional state from vocal expression (Alluhaidan et al., 2023). Unlike sentiment analysis, which focuses on linguistic content, SER emphasizes paralinguistic characteristics such as changes in voice quality, prosody, timing, and articulation that naturally arise from emotional processes. Emotional cues are embedded in how speech is produced rather than the literal meaning of the words, which makes acoustic analysis a central component of SER.

Modern SER systems typically transform raw audio into structured representations that capture temporal and spectral variations. These representations serve as inputs for machine-learning or deep learning models designed to detect distinctive emotional patterns. With the rapid development of artificial intelligence, SER has become foundational to emotionally adaptive technologies such as intelligent tutoring systems, telehealth, conversational assistants, call-center analytics, affect-aware robotics, and in-vehicle safety systems. Emotion-aware systems enable more natural, empathetic human–machine interactions by dynamically responding to users’ affective states (Madanian et al., 2023).

## 2.1.2. Emotions Expression in Human Speech

Human speech is expressive, and emotion influences the way vocal signals are produced and perceived. While words communicate propositional meaning, emotional information is conveyed through para-linguistic cues such as pitch variation, loudness, speech rate, articulation, and voice quality. Listeners rely on these cues to infer a speaker’s affective state even when the linguistic content is ambiguous or unknown. Emotional expression arises naturally from physiological processes, such as changes in respiration, muscle tension, and vocal fold dynamics, that alter the acoustic properties of speech. As a result, speech carries consistent patterns associated with different emotional states, enabling both humans and computational systems to distinguish emotions through sound.

## 2.1.3. The Acoustic Basis of Emotion

Emotional states create measurable changes in a range of acoustic parameters, forming the basis for computational emotion analysis. Prosodic features such as fundamental frequency (pitch), intensity, duration, and rhythm reflect broad differences in arousal and valence. High-arousal emotions tend to produce higher pitch, greater energy, and faster speech, whereas low-arousal states result in reduced vocal intensity and slower tempo. Beyond prosody, spectral characteristics and voice quality also vary with emotion. Adjustments in vocal fold tension and airflow modify harmonic structure, spectral balance, and timbre, producing perceptual qualities such as breathiness, sharpness, or roughness. These acoustic patterns provide essential foundations for modern Speech Emotion Recognition systems, which rely on quantifying such variations to infer affective states.

## 2.2. Literature Research in Speech Emotion Recognition

### 2.2.1. Traditional Approaches

Earlier SER research predominantly relied on handcrafted acoustic features combined with classical machine-learning classifiers. These systems extracted multiple descriptors capturing aspects of prosody, spectral shape, and voice quality, which were then fed into algorithms such as Support Vector Machines (SVMs), K-Nearest Neighbor (KNN), and Random Forests.

Aouani et al. (2020) demonstrated the strength of this paradigm by using a diverse set that included prosodic measures, spectral features, and voice-quality indicators together with SVM classifiers trained on four combined datasets (RAVDESS, TESS, SAVEE, and CREMA-D). Their findings showed that broadening the range of acoustic features improved robustness across speakers and recording conditions.

Similarly, Kaloub and Elgabar (2025) employed a multi-feature extraction strategy for natural Arabic speech using an SVM variant (SMO), Random Forest, and KNN. The study showed that combining diverse acoustic descriptors produced the highest accuracy, although challenges persisted in detecting subtle emotions and handling class imbalance.

Transitional approaches have blended handcrafted features with lightweight deep neural networks. Chowdhury et al. (2025) introduced a hybrid technique integrating engineered acoustic features into 1D-CNN and CNN–BiLSTM architectures. The model achieved performance comparable to spectrogram-based deep networks but at a lower computational cost, highlighting that feature engineering still offers advantages in low-resource settings.

Hema (2023) contrasted traditional systems with convolutional models and reported substantially higher accuracy when CNNs were trained on spectral representations. This indicates a broader shift toward deep learning, though traditional approaches continue to serve as strong baselines.

Although traditional SER methods provide strong baselines, the literature consistently shows that handcrafted features struggle to capture the full complexity of emotional expression in speech. Existing studies highlight limitations in generalization across speakers, languages, and datasets, as well as reduced performance when facing subtle or overlapping emotions. Furthermore, most traditional systems rely on manually engineered features, which are neither scalable nor capable of modeling temporal or spectral dynamics at deeper levels.

### 2.2.2. Deep Learning-Based Approaches

Deep learning has significantly advanced SER by enabling automatic learning of discriminative spectral which is temporal patterns without reliance on manually engineered features. A central component of this shift is the use of neural architecture capable of modeling both local spectral structures and long-term temporal dependencies.

Meng et al. (2019) demonstrated the effectiveness of convolutional models by applying CNNs to stacked log-Mel inputs, showing notable improvements over classical machine-learning baselines. Similarly, Zhao, Mao, and Chen (2019) developed hybrid CNN–LSTM architectures, illustrating that CNN layers are effective at extracting spectral variations while LSTM layers capture dynamic emotional evolution across time.

Sequence modeling advances continued in Mustaqeem et al. (2020), where clustering-enhanced BiLSTM architectures improved robustness to interspeaker variability. Complementary work by Barhoumi and BenAyed (2024) confirmed that CNNs, LSTMs, and 3D CNNs represent the dominant methodological direction in the field, consistently outperforming traditional classifiers and enabling more generalizable feature learning.

### 2.2.3. Multi-Dataset and Cross-Dataset Studies

Despite strong performance within individual corpora, SER models struggle to generalize across datasets due to variations in language, speaker demographics, recording conditions, and cultural influences. This challenge underscores the importance of multi-dataset training and cross-dataset evaluation.

Latif et al. (2022) highlighted this limitation through cross-corpus experiments showing considerable degradation when models were tested on unseen datasets. The study proposed the ADDi and sADDi frameworks, which use adversarial discriminators to learn domain-invariant emotional representations without requiring labeled target data. Similarly, Gideon et al. (2019) extended this idea with ADDoG and MADDoG, which iteratively align emotional feature distributions across corpora using a “meet-in-the-middle” adversarial strategy, leading to improved cross-dataset robustness in noisy and in-the-wild conditions.

Beyond adversarial adaptation, ensemble and multimodal strategies have also shown promise. Ahmed et al. (2021) trained a 1D-CNN–LSTM–GRU ensemble on five benchmark datasets, integrating multiple acoustic feature streams and augmentation techniques to achieve strong cross-dataset performance. Ho et al. (2020) demonstrated that multimodal fusion of audio and text reduces dataset-specific biases and enhances generalization across diverse corpora.

Collectively, the literature indicates that cross-dataset SER remains a core research challenge. However, combining adversarial adaptation, multi-corpus training, multimodal learning, and architectural diversity substantially enhances model robustness and reliability for real-world deployment.

## 2.3. Feature Extraction Techniques for SER

Feature extraction is fundamental to SER, as the quality of emotional representation depends on how effectively acoustic information is encoded. Research explores both handcrafted descriptors and learned time–frequency representations, each offering distinct benefits depending on the modeling approach.

**Handcrafted Acoustic Features**

Mel-Frequency Cepstral Coefficients (MFCCs) are the most widely used handcrafted representation, designed to approximate human auditory perception by capturing the spectral envelope of speech. MFCCs compress information efficiently, making them well-suited for classical machine-learning pipelines. Studies such as Koduru et al. (2020) and Afifah (2020) demonstrate that MFCCs consistently outperform other standalone handcrafted features across languages and datasets, reinforcing their reliability for speaker-independent emotion modelling.

Chroma features, Spectral Centroid, Spectral Bandwidth, and Zero-Crossing Rate (ZCR) provide complementary information regarding harmonic content, spectral brightness, timbre, and signal sharpness. Multi-feature fusion studies (Koduru et al., 2020; Ahmed et al., 2021) show that combining these descriptors with MFCCs enhances discrimination, as different emotional states are associated with distinct spectral or prosodic characteristics.

**Frequency-Domain Features**

Frequency-domain features computed through the Fast Fourier Transform (FFT) also contribute significantly to SER performance. FFT-derived coefficients highlight harmonicity patterns, formant distribution, and spectral energy shifts. Koduru et al. (2020) observed that integrating FFT-based spectral features enhanced discrimination between emotions with similar prosodic contours (e.g. anger vs. disgust) by capturing differences in high-frequency spectral energy.

**Learned Time-Frequency Representations**

Deep learning models increasingly rely on Mel-spectrogram and log-Mel spectrogram inputs, which preserve rich spectral which is temporal structures suitable for convolutional and recurrent architectures. These representations allow models to learn emotional cues directly from the time–frequency domain and have been shown to outperform purely handcrafted approaches.

Recent work, such as Pulatov et al. (2023), demonstrates that Mel-spectrogram–based CNNs achieve strong performance in noisy conditions, while Chowdhury et al. (2025) show that spectrogram-based inputs outperform handcrafted feature systems in cross-speaker and cross-condition settings. Dual encoder approaches further highlight the advantages of combining cepstral and spectral representations, enabling deep networks to capture complementary emotional cues.

# 3. Technology and Tools

Python is suited for machine learning, deep learning, computer vision applications due to its extensive ecosystem of libraries and frameworks for data processing and model development, as well as having large and active community that provides robust support for specialized tasks of speech and language.

|  |  |
| --- | --- |
| Library/Framework | Purpose |
|  | Deep learning framework for building and training neural network |
|  | High-level API for model construction (sequential, functional API) |
|  | Alternative framework (for potential Wav2Vec2 or other models) |
|  | Audio loading, feature extraction (MFCCs, spectrograms, STFT) |
| Best AI Tools: Python library SciPy | Signal processing (Wiener filtering, signal manipulation) |
|  | Numerical array operations and mathematical functions |
| Pandas: 데이터 조작 및 분석을 위한 도구 - 함께해요 파이썬 생태계 | Dataset management, csv manipulation, data frame, operations |
| Joblib: NumPy memmap in joblib.Parallel - 함께해요 파이썬 생태계 | Efficient serialization of numpy arrays and python objects |
| scikit-learn - NumFOCUS | Train/test split, metrics (confusion matrix, classification report, ROC curves) |
| Matplotlib Logo PNG Vector (SVG) Free Download | Plotting training curves, spectrograms, waveforms, confusion matrices |
| Seaborn: Matplotlib을 기반 통계적 데이터 시각화 - 함께해요 파이썬 생태계 | Enhanced statistical visualizations (heatmap for confusion matrices) |
| JSON ETL/ELT Data Pipeline Solution | Saving/loading training history and metadata |
| Natural language processing services & Consulting Company | Audio stream handling for real-time applications |
| Create a Dashboard with Streamlit | Saturn Cloud | Lightweight web UI to demo model inference and visualizations |

# 4. Software Product Requirements

## 4.1. Data Preprocessing and Organization

Effective data preprocessing and organization are essential to ensure the reliability, consistency and usability of the multi-dataset speech corpus used in this project. Because the study integrates four datasets, each with distinct recording protocols, speaker demographics, emotional categories and file structures, a unified preprocessing workflow is necessary to prepare the data for feature extraction and model training.

The first stage involved examining the structure the metadata of each dataset to identify variations in file naming conventions, audio formats, duration ranges and emotion labels. To achieve consistency across corpora, all audio files were reorganized and renamed using a standardized filename convention based on the RAVDESS structure, ensuring that key attributes are encoded uniformly. This step reduces errors in downstream processing.

Exploratory data analysis (EDA) was conducted to examine duration distributions, label frequencies, sampling rates, and class imbalances across datasets. Visualizations such as histograms and bar charts revealed cross-dataset variability and guided key preprocessing decisions.

The raw audio files were subsequently processed using key operations such as trimming silence, reducing noise, normalizing amplitude, standardizing clip length when appropriate, and converting them into uniform time–frequency representations. These steps ensure compatibility across datasets and improve model robustness.

After preprocessing, all audio samples were systematically organized into structures directories and divided into training, validation and testing sets, enabling efficient data handling and reproducible model development.

## 4.3. Feature Extraction and Analysis

Once the audio signals were preprocessed and organized, the next stage involved extracting informative acoustic features that capture emotional characteristics embedded in speech. Feature extraction plays a central role in SER, as the process transforms raw audio waveforms into structures suitable for machine learning and deep learning models. The primary objective of this stage is to identify spectral, temporal and time-frequency characteristics that differentiate one emotional state from another.

To achieve this, several complementary approaches were employed:  
**Frequency Domain Features**: Frequency-based analysis was conducted to extract descriptors that reflect how energy is distributed across frequency bands. Key features include Mel-Frequency Cepstral Coefficients (MFCCs), spectral centroid, spectral bandwidth, and spectral roll-off. These representations are widely used in SER due to their ability to model vocal tract dynamics and prosodic changes.

**Time – Frequency Features:** To capture both spectral and evolution over time, time-frequency representations were generated, most notably Mel-spectrogram and log-Mel spectrograms. These two-dimensional feature maps preserve temporal transitions essential for emotion modelling and are highly compatible with convolutional neural networks (CNNs).

Preprocessing techniques such as silence trimming and noise reduction helped reduce unwanted artifacts, enabling the extracted features to more accurately reflect emotion-related characteristics. These feature representations collectively form the foundation for the model training pipeline, providing both low-level signal properties and rich spectral–temporal structures for classification.

## 4.4. Deep Learning Model

The deep learning component of this project is designed to evaluate the effectiveness of modern neural architectures for SER across multiple datasets. The objective is to establish a robust baseline, assess cross-dataset generalization, and identify architectures that can reliably capture both spectral and temporal emotional cues.

The study will examine a broad range of model families to capture different aspects of emotional speech signals, including:

* CNN based architectures for spectral feature learning (2D CNNs on Mel/Log-Mel spectrograms).
* RNN based models, including LSTM variants, to model temporal dependencies in raw or feature based sequences.
* 3D-CNN models capable of learning spatio-temporal features directly from stacked spectrogram segments.

Attention based architectures, where feasible, to enhance temporal weighting and focus on salient emotional segments. Evaluate models using standardized time-frequency representations

All deep learning architectures must be trained on consistent input representations, including:

* Mel-spectrograms
* Log-Mel spectrograms
* Stacked spectrogram frames

Assess performance using a comprehensive set of evaluation metrics. Model evaluation will utilize:

* Precision, Recall, and F1-Score (macro and weighted)
* Accuracy
* Confusion matrices to analyse per-emotion strengths and weaknesses
* Generalization behavior across datasets to determine robustness

Ensure reproducibility and efficient model training. All models must:

* Be implemented in PyTorch or TensorFlow
* Use consistent training protocols (optimizer, scheduler, early stopping)
* Include reproducible splits for training/validation/testing
* Be scalable for future extensions (e.g., multimodal SER or real-time inference)

## 4.5. Graphical User Interface (GUI) Development

A lightweight and interactive graphical user interface (GUI) will be developed to support model demonstration, usability testing and end to end workflow evaluation. The GUI serve as the primary medium for users to record speech through a microphone, observe preprocessing results, and obtain emotion predictions generated by the trained SER models. Streamlit is selected as the development framework due to its simplicity, rapid prototyping capabilities and strong compatibility with Python machine learning pipelines.

The interface will present a brief overview of the project, followed by an interactive section where users can record voice. The submitted audio will be processed and forwarded to the deployed SER model hosted on Hugging Face. The model will then return the predicted emotion label along with relevant confidence information.

# 5. Review of Software Development Methodologies

## 5.1. Waterfall

The Waterfall model is a linear and sequential project management methodology introduced by Royce (1970). It organizes development into five distinct phases (requirements, design, implementation, verification, and maintenance) each of which must be completed before progressing to the next stage.

Strengths:

* Clear structure and documentation: Each phase has well defined objectives, deliverables and documentation, which supports transparency and systematic project tracking.
* Predictability: the linear progression enables more accurate early-stage planning, including cost estimation, scheduling and resource allocation.
* Stability for large projects: waterfall is effective when requirements are unchanged, making it suitable for large or highly regulated projects where changes are minimal.

Limitations:

* Limited flexibility: the sequential nature makes it difficult to accommodate evolving requirements or unexpected issues once a phase is completed.
* Risk of late problem discovery: errors made early such as unclear requirements may only be detected during later stages, increasing project risks.
* Slower adaptation and delivery: the absence of iterative cycles can lead to longer development timelines.

## 5.2. Spiral

Spiral methodology combines waterfall and iterative method growth by Barry Boehm (1988). The model is structured as a series of iterative cycles, with each cycles incorporating four key activities. These repeated iterations enable continuous refinement of requirements and solutions while placing strong emphasis on identifying and mitigating risks at every stage.

Strengths:

* High flexibility: the iterative cycles allow continuous refinement of requirements, design, and functionality throughout the project
* Early risk detection: the strong emphasis on risk analysis helps identify and mitigate major technical lor managerial risks at early stages.
* Well-suited for complex and large-scale systems: the model adapts effectively to projects with evolving requirements, high uncertainty, or safety-critical constraints.
* Continuous customer involvement: frequent review ensures stakeholder alignment and helps validate the system incrementally.
* Progressive development: working prototypes are delivered at multiple stages, enabling early feedback and evaluation.
* Strong documentation discipline: each phase produces structured documentation, improving traceability and long-term maintenance.

Limitations:

* High cost and time demand: repeated iterations and extensive risk analysis make the model costly and unsuitable for small, low-risk projects.
* Requires specialized expertise: effective risk analysis depends on experienced professionals, and poor risk analysis can compromise the entire process.
* Process complexity: the model is often considered difficult to manage due to its many phases, review and iterative loops.
* Potential scope uncertainty: continuous refinement may lead to expanding spirals, making timelines and final deliverables harder to define early.
* Heavy documentation load: frequent iterations require additional artifacts, increasing administrative overhead.
* Dependence on strict adherence: success relies on consistently following the model’s protocols, which may be challenging for inexperienced team.

## 5.3. RAD (Rapid Application Development) – Prototyping

RAD is a software development methodology that prioritizes continuous user feedback, iterative prototyping, and rapid refinement rather than adhering to a rigid, predefined plan. (Martin, 1991)

Strengths:

* Accelerates development through iterative prototyping and frequent user involvement.
* Enhances alignment with user needs by incorporating continuous feedback.
* Provides flexibility to modify requirements throughout the development process.

Limitation:

* Requires high user availability and active participation, which may not always be feasible.
* Best suited for small to medium-scale projects. Large complete systems may be difficult to prototype rapidly.

May lead to reduced documentation and structure, potentially effecting long-term maintainability

## 5.4. Agile

Agile is a software development methodology that is iterative process, continuous user involvement and the rapid delivery of function components. Instead of completing all requirements in once at the end, Agile divides the project into smaller, manageable iterations in which working prototypes are produced regularly. This approach enables developers, stakeholders, and end users to collaborate throughout the development process, allowing ongoing feedback and adaptation. (Manifesto, 2001)

Strengths:

* Facilitate adaptability by allowing requirement changes at any stage of development.
* Enhance stakeholder satisfaction through continuous delivery of working components.
* Encourages strong collaboration between developers and stakeholders.
* Supports rapid problem identification and correction through frequent iterations.
* Promotes high product quality through continuous refinement and testing.

Limitation:

* Requires consistent stakeholder engagement, which may be difficult to maintain.
* This can lead to scope of uncertainty due to evolving requirements.
* Less effective for projects requiring strict documentation and rigid timelines.
* May challenge inexperienced teams, such as relying heavily on self-management and communication.
* It is difficult to estimate budget and schedule precisely due to iterative changes.

## 5.5. Selection of Software Development Methodology

Agile is suitable for the SER project due to the experimental and iterative nature of machine learning development. SER involves multiple cycles of data preprocessing, feature extraction, model training, evaluation and refinement. These processes is benefit from continuous feedback and adaptation. The ability to adjust model architectures, incorporate new datasets and refine preprocessing strategies aligns Agile’s flexibility. Frequent iteration allows early testing of prototypes, enabling the team to evaluate classification performance, identify weaknesses. Moreover, Agile’s emphasis on collaboration complements the multidisciplinary nature of SER, where decisions on data engineering, model selection, and system deployment must be coordinated. For these reasons, Agile provides a practical and effective framework for managing the development of the SER system.

# 6. Software Product Requirements

## 6.1. Workflow Overview

The following diagram illustrates the system workflow for the SER project, outlining all major processing stages from data acquisition to model evaluation and insight communication. This workflow ensures a systematic, reproducible, and modular approach to building a deep learning-based SER system that integrates multiple datasets and applies standardized preprocessing, feature extraction, modelling and evaluation procedures.

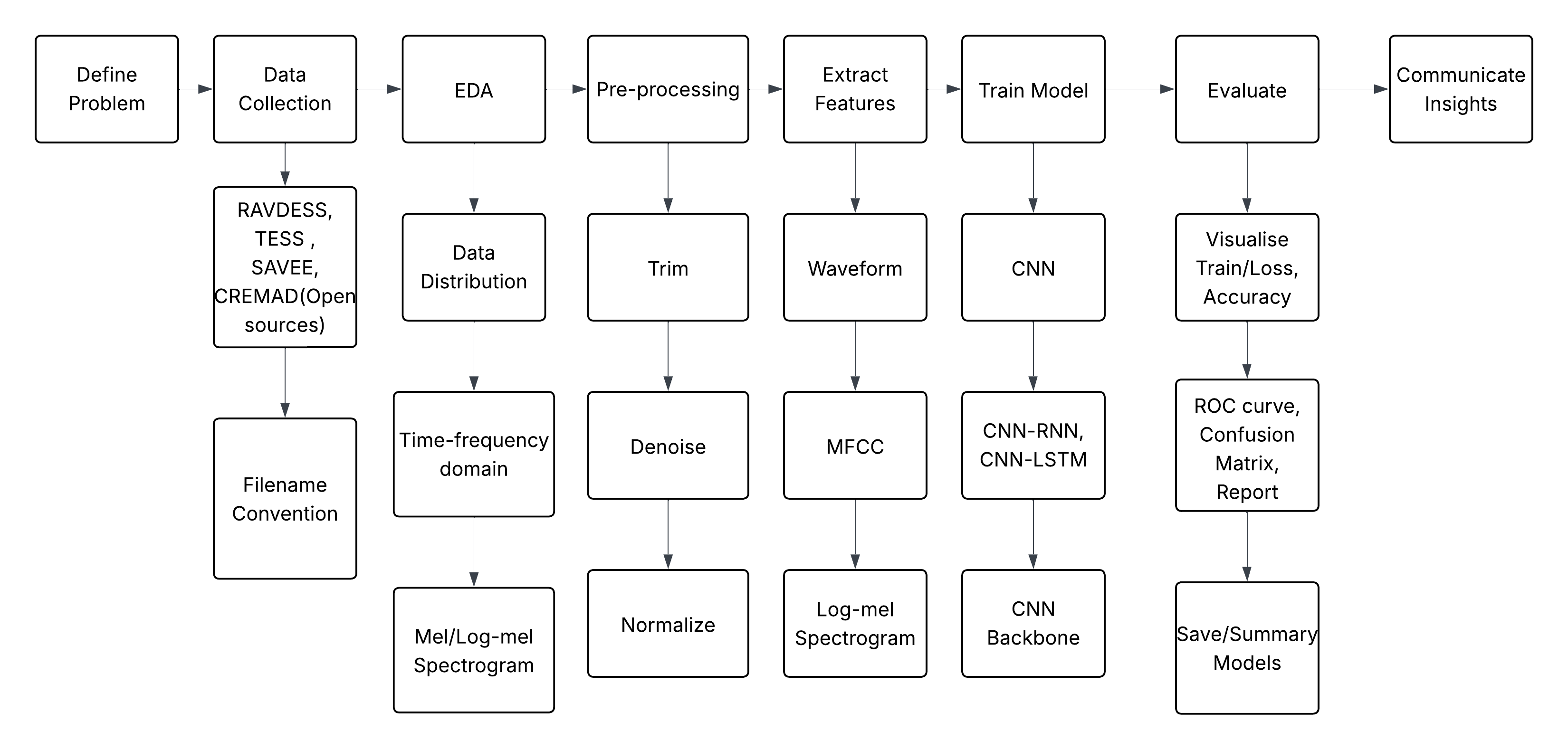


Figure : The architecture of the project

**Define Problem:** This stage established research motivation, objectives and scope. This step verifies the target emotions, modelling approach (CNN, CNN-LSTM), and constraints such as dataset diversity and cross-dataset consistency.

**Data Collection:** Datasets are gathered from RAVDESS, TESS, SAVEE, CREMA-D. Because these datasets use different naming schemes and formats, a filename convention is implemented to unify all recordings under a consistent structure, enabling easier merging and processing in later steps.

**Exploratory Data Analysis (EDA):** EDA provides an initial understanding of dataset composition, including emotion distribution, sapling rates and duration, time-frequency inspection.

**Pre-processing:** Raw audio files undergo several transformations to ensure quality and consistency: trimming, denoising, normalization, spectrogram generation. These steps prepare the data for robust feature extraction.

**Feature Extraction:** Both time-domain and frequency-domain features are extracted, including waveforms, MFCCs, Mel/Log-mel spectrograms. The representations are used as input to models.

**Train model:** Machine learning and deep learning models are trained using processed features. Training progress is monitored using accuracy, loss and validation curves.

**Evaluation:** Model performance is assessed using confusion matrix, ROC curves, classification reports, summary of saved model versions. This enables comparison between architecture and identification of strengths/weaknesses.

Finally, results are compiled and communicated via reports, visualizations or applications dashboards.

## 6.2. Data Collection

**RAVDESS** – The Ryerson Audio-Visual Database of Emotion Speech and Song

This dataset is obtained from Kaggle. The database consists of recordings from 24 professional actors (12 female, 12 male) who vocalize lexically matched statements in a neutral North American accent. The dataset includes seven emotional categories which are calm, happy, sad, angry, fearful, surprise and neutral, although only the speech only subset (2452 audio files) is used in this project. All recordings are provided in high-quality voice-only format.

**TESS** – Toronto Emotional Speech Set

The TESS dataset is downloaded from Kaggle, contains 2,800 stimuli produced by two actresses aged 26 and 64. Each actress recorded 200 target words across seven emotional categories: anger, disgust, fear, happiness, pleasant surprise, sadness, and neutrality. The dataset provides clean, clearly articulated speech samples suitable for emotion classification tasks.

**SAVEE** – Surrey Audio-Visual Expressed Emotion

SAVEE consists of 480 British English utterances recorded by four male actors expressing seven emotional states (anger, disgust, fear, happiness, sadness, surprise and neutral). All sentences were carefully selected from the TIMIT corpus to ensure phonetic balance across emotions. The recordings are labelled, processed, and structured for both audio and audiovisual emotion research.

**CREMA-D** – Crowd-source Emotion Multimodal Actors Dataset

CREMA-D is a labelled multimodal dataset containing 7442 clips of 91 actor (48 male, 43 female) ranging from 20 to 74 years old and representing diverse racial and ethic backgrounds. Actors produced sentences in six emotional categories: anger, disgust, fear, happiness, sadness and neutrality. Each clip includes vocal and facial expressions, though only the audio component is used in this study.

Across all four datasets, the combined corpus used in this project consists of 13174 audio samples.

## 6.3. Data Preprocessing and Organization

### 6.3.1. RAVDESS Filename Convention

Because this project integrates multiple datasets, it is essential to standardize filenames to ensure consistency and to streamline subsequent preprocessing steps. Therefore, all datasets are reformatted to follow the RAVDESS filename structure, which consists of seven components:

Table : Filename components of RAVDESS dataset

|  |  |  |
| --- | --- | --- |
| Component | Description | Value |
| Modality | Specifies the type of data | 01 = Speech, 02 = Song |
| Vocal channel | Indicates the vocal mode used in the recording | 01 = Speech, 02 = Song |
| Emotion | Encodes the emotional category expressed in the utterance | 01=Neutral, 02=Calm, 03=Happy, 04=Sad, 05=Angry, 06=Fearful, 07=Disgust, 08=Surprised |
| Emotional intensity | Specifies the intensity level of the expressed emotion | 01 = Normal, 02 = Strong (not used for neutral emotion) |
| Statement | Identifies which scripted sentence is spoken | 01 = “Kids are talking by the door”, 02 = “Dogs are sitting by the door” |
| Repetition | Indicates the repetition number of the same sentence. | 01 = First repetition, 02 = Second repetition |
| Actor | A unique numerical identifier for the speaker. Gender also encoded (even = female, odd = male) | 01 - 24 |

### 6.3.2. Exploratory Data Analysis

An Exploratory Data Analysis (EDA) was conducted to examine the characteristics, distribution, and variability of the four emotional speech datasets used in this project. The analysis focuses on class distribution, speaker diversity, acoustic properties, and cross-dataset variability, all of which directly influence model performance and generalization.

**Dataset composition and class distribution:**

The four datasets vary substantially in size and emotional category coverage. RAVDESS and CREMA-D offer the largest number of samples, whereas SAVEE is limited to four speakers and TESS contains recordings from only two actors. Common emotions such as happiness, sadness, and neutral appearance across all datasets, while others (e.g., fear, disgust) are inconsistently represented.

A graph with purple bars

AI-generated content may be incorrect.

Figure : The emotion contribution of each dataset.

**Audio duration:**

The datasets differ in duration:

* RAVDESS: ~3 sec clips
* CREMA-D: 2–4 sec
* TESS: single-word recordings (~1 sec)
* SAVEE: variable lengths across sentences

The longest duration is 7.14 seconds. 95th percentile of duration is 84610 with approximately 3.84 seconds

The attribute duration of all 4 datasets without trimming silent:

A diagram of a distribution of audio durations

AI-generated content may be incorrect.

Figure :Distribution of audio duration

**Spectral and Temporal Characteristics:**

Preliminary inspection of waveforms and Mel-spectrograms revealed that high-arousal emotions (e.g. anger and fear) exhibit higher energy and a broader spectral spread, whereas low-arousal emotions (e.g. sadness) display lower intensity and a narrower spectral pattern. These observations confirm the suitability of spectrogram-based inputs for deep learning models.

|  |  |
| --- | --- |
| A close-up of a sound wave  AI-generated content may be incorrect. | A close-up of a sound wave  AI-generated content may be incorrect. |
| A close-up of a sound wave  AI-generated content may be incorrect. | A close-up of a sound wave  AI-generated content may be incorrect. |

Figure : Waveform and spectrogram of emotions

**Recording Quality and Environmental Variability:**

The speech records having the silence before and after:

A close-up of a sound wave

AI-generated content may be incorrect.

Figure : Waveform of before and after trimming

The audio duration distribution of 4 datasets:

A graph of a distribution of audio durations

AI-generated content may be incorrect.

Figure : Distribution of audio duration after trimming

Environmental characteristics differ among datasets:

A group of blue sound waves

AI-generated content may be incorrect.

Figure : Environment characteristics

### 6.3.3. Determining Significant Frequency Ranges

To identify frequency bands that exhibit the greatest discriminative power across different emotional speech categories, thereby informing optimal feature extraction and preprocessing strategies for speech emotion recognition.

**Per-emotion average power computation:**

For each emotion class , all corresponding samples were extracted and transformed into log-mel spectrograms S(t, f) using the short-time Fourier transform framework (Oppenheim & Schafer, 2010). The temporal average power for each frequency bin was then calculated as

Where denotes the number of time frames. These per-sample averages were subsequently aggregated across all samples in class to obtain an emotion-level average spectrum .

**Inter-emotion variance analysis:**

For each frequency bin f, a vector of average powers across emotions was constructed:

And the across-emotion variance was computed as

where is the number of emotions and is the mean power for frequency . Frequency bins exhibiting high variance were interpreted as containing meaningful spectral distinctions between emotion classes and thus providing potential discriminative value for feature selection.

A graph showing a line

AI-generated content may be incorrect.

Figure : Variance of average power

**ANOVA-based significance testing:**

To validate the statistical significance of the observed variance, a one-way Analysis of Variance (ANOVA) was performed:

* Null Hypothesis (H₀): The mean spectral power at frequency *f* is equal across all emotion categories.
* Alternative Hypothesis (H₁): At least one emotion exhibits a significantly different mean power at frequency *f*.

**For each frequency bin (f):**

* Collect samples representing spectral power values at f for emotion category.
* Compute F-statistic: F = (between group variance) / (within-group variance)
* Obtain p-value from F-distribution with k-1, N-k degree of freedom (k = number of emotions (8), N = total number of samples)
* Apply significance threshold α = 0.05. Frequency bins with p < 0.05 were considered statistically significant discriminators.

**Contiguous range identification:**

To convert discrete significant frequency bins into interpretable ranges:

* Extract indices of all bins where p < α
* Identify consecutive sequences of significant bins
* Merge bins separated by ≤ 100Hz into continuous ranges
* Filter out narrow ranges with bandwidth < 200Hz

A graph with a line

AI-generated content may be incorrect.

Figure : ANOVA significance across frequences

**Cumulative discriminative power curve:** To verify that discriminative spectral patterns were consistent across datasets, the analysis was repeated for RAVDESS , TESS , CREMA-D , and SAVEE . Frequency bins were considered robust if significant in at least two datasets, with a stricter consensus requiring significance across all four. A cumulative discriminative power curve was then constructed by converting p-values into discriminative scores and computing their normalized cumulative sum. This curve highlights how much discriminative information is captured below key cutoff frequencies (e.g., 2 kHz, 3.7 kHz, 8 kHz) and helps identify the spectral ranges most relevant for emotion classification.

A graph with a line graph

AI-generated content may be incorrect.

Figure : Cumulative discriminative power vs frequency

**Summary of results:**

The analysis showed that most emotion relation information in speech is concentrated on two main frequency regions. The first and strongest range is 200-800 Hz, where fundamental pitch and the first formant lie. The second important region is 1500-3700 Hz, which includes higher formants and vocal resonance patterns

When comparing all four datasets, most of the significant frequency ranges fell between 0-3740 Hz and the frequency range consistently important across all datasets was 300-2800Hz

A cumulative analysis of discriminative power showed that:

* 200 Hz captures about 50% of the useful emotional information.
* 3700 Hz capture 69-75% of the information
* 8000 Hz captures over 90% of the information

Overall, these results indicate that the low to mid frequency ranges carry most emotional cues in human speech, making them especially important for feature extraction on SER models.

### 6.3.4. Preprocessing

Raw speech recordings contain various artifacts which may degrade model performance, including background noise, silence padding, and amplitude variations. A systematic preprocessing pipeline was implemented to enhance signal quality and standardize audio characteristics across all datasets prior to feature extraction and model training.

The preprocessing pipeline was implemented using the Librosa library.

|  |  |  |
| --- | --- | --- |
| Step | Purpose | Key Advantage |
| Load and resample | Normalize sampling rate | Unified temporal resolution |
| Coarse trim | Remove silence padding | Reduces irrelevant segments |
| Denoise | Suppress background noise | Enhances speech clarity |
| Fine trim | Remove residual noise artifacts | Cleaner speech boundaries |
| Normalize | Standardize dynamic range | Stabilizes training |

The pipeline consists of five sequential stages:

**1. Audio loading and resampling:**

All audio files were resampled to 22050 Hz, ensuring a consistent temporal resolution across the dataset. This sampling rate preserves essential speech information (fundamental frequency, formants, and high-frequency emotional cues) while reducing computational cost compared to higher rates.

**2. Coarse silence trimming (30 dB threshold)**

Leading and trailing silence was removed based on a 30 dB below peak threshold. This step reduces unnecessary padding and focuses the signal on speech-active regions, lowering computational overhead.

**3. Wiener denoising:**

A Wiener filter was applied to suppress steady-state background noise while preserving key speech characteristics.

* Provides adaptive noise reduction across different recording conditions.
* Improves clarity of prosodic and spectral emotion cues with minimal distortion of formants and consonant transitions.

However, since Wiener filtering assumes stationary noise, it may be less effective under strong non-stationary conditions.

**4. Fine silence trimming (25 dB threshold)**

A secondary trimming pass with a stricter threshold removes residual low-energy artifacts that may remain after denoising, ensuring clearer speech boundaries without discarding emotionally relevant phonemes.

**5. Peak amplitude normalization**

Each sample was normalized to the range [-1, 1] using peek normalization.

Advantages:

* Standardizes loudness levels across recordings.
* Prevents models from learning speaker loudness instead of emotional patterns.

Stabilizes training by ensuring consistent input scales.

RMS normalization was evaluated but discarded due to sensitivity to outliers.

### 6.3.5. Batch preprocessing and dataset organization

All 13,173 samples were processed automatically and saved using a speaker-wise directory structure, preserving dataset provenance and supporting efficient speaker-independent train/validation/test splitting.

* **Processing success rate:** 100%
* **Mean duration reduction:** ~18% (due to silence trimming)
* **Estimated SNR improvement:** +3–5 dB, following denoising and boundary refinement
* **Output consistency:** All samples normalized, noise-reduced, resampled to 22.050 Hz, and trimmed using standardized threshold.

This standardized and reproducible preprocessing structure provides a clean and uniform dataset, improving the reliability of downstream feature extraction (e.g., Mel-spectrograms) and enhancing model performance and generalization. The processed dataset is subsequently partitioned into train, validation, and test sets based on speaker ID to prevent data leakage and ensure robust evaluation.

## 6.4. Feature Extraction and Analysis

After preprocessing, several acoustic feature representations were extracted to support model development and experimentation. Although the project primarily focuses on log-mel spectrograms for deep-learning, two additional representations which are raw waveforms and MFCCs, were generated to benchmark traditional and learning approaches. This multi-feature strategy allows comparison between handcrafted features, classical time-domain signals and spectrogram deep models.

### 6.4.1. Raw waveform representation

Raw waveforms were extracted by loading each pre-processed audio sample at the unified sampling rate and standardizing duration to 4 seconds. These waveforms were used only for one baseline experiment with a simple 1D-CNN model to evaluate end-to-end learning performance.

Advantages:

* Retains the complete time-domain structure
* Requires no handcrafted transformations

Limitations:

* Very high dimensionality
* Computationally heavy and less informative than spectral features
* Not suited for small datasets

As expected, waveform-based models achieved lower performance and were not used in final experiments.

### 6.4.2. MFCC Extraction

MFCCs were computed as a compact alternative to spectrograms. Each sample was transformed into 40 MFCC coefficients over a fixed number of frames. This approach was evaluated once using a simple CNN classifier to replicate traditional SER baselines.

**Advantages**

* Low dimensionality
* Well-studied in classical SER

**Limitations**

* Lossy transformation discarding fine-grained spectral details
* Less effective than spectrograms for deep learning
* Sensitive to variations in background noise

MFCCs provided acceptable baseline accuracy but underperformed compared to spectrogram-based models.

### 6.4.3. Log-Mel Spectrogram

Log-Mel spectrograms serve as the core feature in this project due to their strong compatibility with CNN-based architectures and their ability to represent both spectral shape and temporal evolution of speech.

Three spectrogram variants were generated to support different experiments:

**1. Fixed Size Log-Mel Spectrograms**

A fixed 2D representation (64 Mel bands × 128 frames) was created for efficient batch training with standard CNNs. The frequency range was restricted to 0–4,000 Hz, which covers the most informative speech frequencies.

**Characteristics**

* Uniform size for all samples
* Suitable for CNNs requiring fixed spatial dimensions
* Compact and computationally efficient

**2. Full-Frequency Spectrograms (0–11,025 Hz)**

A full-spectrum version (256 Mel bins) was generated for comparison with the filtered variant. This variant preserves the complete Nyquist range and is used to evaluate whether high-frequency components improve model performance.

**Findings**

* Larger size and higher computational cost
* Minimal performance gain compared to the 3,700 Hz spectrograms
* Best suited for hybrid CNN–RNN architectures

Using Log-Mel spectrograms as the main representation offers several benefits:

* Capture emotional cues in pitch, energy, timbre, and harmonic structure
* Spatially structured (2D), ideal for CNN feature extraction
* Robust to variation in speech rate and intensity
* Visually interpretable for qualitative error analysis
* Flexible for both fixed-size and variable-length model designs

## 6.5. Machine Learning and Deep Learning Implementation and Evaluation

### 6.5.1. Feature-Specific Baseline Experiments

To establish comparative benchmarks and validate the superiority of log-mel spectrograms for speech emotion recognition, two additional baseline models were trained on alternative acoustic representations: raw waveforms and Mel-Frequency Cepstral Coefficients (MFCCs). These experiments isolate the impact of feature engineering choices on model performance.

#### a. Raw Waveform – 1D CNN

The raw waveform model uses the preprocessed audio signal directly in the time domain. All samples were standardized to a fixed length of 4 seconds, and a shallow 1D CNN was applied to capture local temporal variations in amplitude without any transformation to the frequency domain.

The model achieved low classification accuracy and exhibited highly unstable validation performance, as shown in Figure 11. The large input dimensionality led to rapid overfitting, and the network struggled to learn emotion-discriminative patterns from amplitude-only signals, which lack explicit pitch and timbral information.

A graph with orange and blue lines

AI-generated content may be incorrect.

Figure : Validation accuracy of the raw waveform model

Limitations:

* Does not encode frequency structure; pitch and formant-related emotion cues are lost.
* High dimensionality leads to inefficient learning and long training times.
* Performs poorly on small datasets due to lack of inductive bias.

#### b. MFCC – 1D CNN Baseline

MFCCs (40 coefficients × 199 frames) were extracted from each audio sample and used to train a lightweight 1D CNN. MFCCs provide a compact representation of the spectral envelope and are widely adopted in traditional speech recognition systems.

The MFCC-based model achieved moderate performance, clearly outperforming the raw waveform baseline. However, its accuracy remained significantly below spectrogram-based CNNs. Misclassifications were common in emotions with overlapping cepstral patterns, particularly happy, surprised, and fearful. This occurred because MFCCs compress fine-grained harmonic and energy variations that are important for distinguishing emotional states.

A graph with blue and orange lines

AI-generated content may be incorrect.

Figure : Accuracy of MFCC 1D CNN model

**Limitation:**

* MFCCs remove detailed spectral and harmonic information through DCT compression.
* Lack sensitivity to pitch contours, affecting recognition of high-arousal emotions.
* Fixed-length normalization may truncate emotionally relevant segments.

### 6.5.2. Spectrogram-Based CNN Models

Spectrograms are served as the primary input format for deep learning models in this project. Owing to their two-dimensional structure, spectrograms allow convolutional neural networks (CNNs) to learn both spectral (frequency-based) and temporal (time-varying) patterns that are characteristic of emotional speech.

#### a. Fixed-Size Spectrogram CNN

For the baseline architecture, log-Mel spectrograms of dimension 64 × 128 were generated using standardized padding and truncation to ensure consistent input shape across all datasets. The CNN architecture comprised three convolutional blocks, each followed by max-pooling, enabling hierarchical extraction of local time–frequency features. A global average pooling layer aggregated the learned feature maps, and a fully connected SoftMax layer produced the final emotion predictions.

Performance overview:

The model achieved stable improvements in training accuracy across epochs, reaching approximately 94–95%. Test accuracy remained lower (≈ 84–86%) and exhibited greater fluctuation, as illustrated in Figure 13. This divergence indicates mild overfitting, which is common in spectrogram-based CNNs trained on heterogeneous emotional speech datasets.

A graph with blue and orange lines

AI-generated content may be incorrect.

Figure : Training vs testing accuracy for spectrogram CNN model

**Comparation table:**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Feature type | Model style | Parameters | Performance | Strengths |
| Raw waveform | 1D CNN | High | Low | No handcrafted feature required |
| MFCC | 1D CNN | Low | Moderate | Efficient, compact representation |
| Log-Mel spectrogram | 2D CNN | Moderate | High | Preserve spectral structure. Best for CNNs |

A close-up of a graph

AI-generated content may be incorrect.

Figure : Accuracy comparation between waveform, MFCC, spectrogram CNN

#### b. Sliding-Window CNN (Overlapping Windows)

To enhance temporal feature representation and improve model generalization, the spectrogram-based input was processed using a sliding-window segmentation technique. Instead of resizing entire spectrograms, each audio sample was divided into overlapping windows of size 128 × 128 pixels, with 10% overlap along the time axis. This approach increased the effective number of training samples and preserved temporal continuity without introducing distortion. A CNN architecture like the baseline spectrogram model was applied to these windows.

For full audio clip predictions, outputs from individual windows were aggregated using majority voting or probability averaging, ensuring robust final classification.

**Evaluation results:**

|  |  |  |  |
| --- | --- | --- | --- |
|  | Precision | Recall | F1-score |
| Neutral | 0.79 | 0.66 | 0.71 |
| Calm | 0,67 | 0.80 | 0.73 |
| Happy | 0.56 | 0.48 | 0.52 |
| Sad | 0.50 | 0.47 | 0.49 |
| Angry | 0.55 | 0.55 | 0.55 |
| Fearful | 0.51 | 0.73 | 0.60 |
| Disgust | 0.64 | 0.57 | 0.61 |
| Surprised | 0.82 | 0.77 | 0.80 |
| Accuracy |  |  | 0.59 |

The classification report for the sliding-window CNN model is summarized below:

Accuracy trends: Training accuracy increased steadily to 0.72, while while validation accuracy plateaued around ~0.60, indicating mild overfitting.

Loss trends: Training loss decreased consistently, whereas validation loss exhibited fluctuations, suggesting variability in generalization across epochs.

A graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of a graph of

AI-generated content may be incorrect.

Figure : Training vs validation accuracy and loss

The confusion matrix figure showed that strong performance for neutral and surprised classes. Frequent misclassification occurred among happy, sad and fearful. This result reflects overlapping acoustic features. Fearful and disgust achieved high recall, but precision remained moderate.

|  |  |
| --- | --- |
| A chart with numbers and symbols  AI-generated content may be incorrect.  Figure :Confusion matrix of spectrogram overlap | A graph of different colored lines  AI-generated content may be incorrect.  Figure : ROC curve |

The sliding-window CNN approach significantly improved temporal feature capture and enhanced class separability compared to fixed-size spectrogram models. While the overall accuracy of 59% indicates room for improvement, high AUC scores for most classes confirm strong discriminative capability. Future improvements could include:

* Incorporating attention mechanisms for better temporal weighting.
* Addressing class imbalance through advanced augmentation or focal loss.
* Exploring hybrid CNN–RNN architectures for sequential context modeling.

### 6.5.3. CNN Hybrid Model for Spectrogram

Fixed-size CNNs fail to capture the temporal progression of acoustic patterns across an utterance. To address this, two hybrid architectures were developed by combining convolutional front ends with recurrent layers (GRU and LSTM). CNN layers extract local spectral features from log-Mel spectrograms, while RNN layers model temporal evolution which is critical for emotion recognition.

**Data Representation**

Full-frequency spectrograms (0–11 kHz) were generated using a 256-Mel filter bank. Due to variable durations across four datasets, an adaptive padding/truncation strategy was applied based on the 95th percentile of training lengths. Data splits: 70% train, 15% validation, 15% test across eight emotion classes.

**Model Architectures**

* **CNN-GRU:** Three Conv2D blocks (Conv – BatchNorm – MaxPool) followed by bidirectional GRU, global pooling and dense layer.
* **CNN-LSTM:** Same CNN extractor, but recurrent module replaced with bidirectional LSTM for long-range dependencies.

Both models used categorical cross-entropy, Adam optimizer, batch size 32, and early stopping over 100 epochs.

**Results:**

|  |  |
| --- | --- |
| A screenshot of a computer screen  AI-generated content may be incorrect.  Figure : Classification report of CNN\_GRU | A screen shot of a computer  AI-generated content may be incorrect.  Figure : Per-class accuracy of CNN-GRU |

* **CNN-GRU:** Faster training, fewer parameters, competitive performance better for deployment. Both significantly outperformed fixed CNN baselines by capturing temporal cues such as pitch contours, rhythm, and energy variations.
* **CNN-LSTM**: Slightly higher accuracy but heavier and slower.

|  |  |
| --- | --- |
| Figure : Classification report of CNN-LSTM | Figure : Per-class accuracy of CNN-LSTM |

A screenshot of a grid

AI-generated content may be incorrect.

Figure : Confusion matrix of CNN hybrid models

CNN–GRU and CNN–LSTM deliver robust, temporally-aware architectures for SER, bridging CNNs and future transformer-based models. Future work may explore attention mechanisms, deeper recurrent layers, or transformer embeddings (e.g., Wav2Vec2).

### 6.5.4. Sentiment-Level Classification with CNN–RNN, CNN–LSTM and CNN–ResNet

To complement eight-class emotion classification, a secondary experiment grouped emotions into three broader sentiment categories:

* **Positive:** happy, surprised.
* **Moderate:** neutral, calm.
* **Negative:** sad, angry, fearful, disgust.

This approach reduces label noise across heterogeneous datasets and enables evaluation on coarser affective dimensions.

**Dataset Preparation:**

Four datasets (RAVDESS, TESS, SAVEE, CREMA-D) were merged and balanced using stratified down sampling.

|  |  |
| --- | --- |
| Figure : Distribution of sentiment classes imbalanced | Figure : Distribution of sentiment classes balanced |

Fine-grained emotion metadata was preserved for misclassification analysis. All audios underwent standardized preprocessing (denoising, silence trimming, normalization), and log-Mel spectrograms were extracted (128 Mel bins, 16 kHz sampling rate, 0–8 kHz coverage). Data split: 70% train, 15% validation, 15% test.

**Models developed:**

* CNN-RNN: Time-distributed CNN front-end + bidirectional GRU for sequential modeling.
* CNN-LSTM: Similar CNN extractor with bidirectional LSTM for long-range dependencies.
* CNN-ResNet: Residual CNN on RGB spectrograms (magma colormap) for high-capacity baseline.

Training used categorical cross-entropy, Adam optimizer, early stopping, and model checkpointing.

**Evaluation metrics:**

* **CNN-RNN:**

The CNN–RNN architecture achieved an overall accuracy of 83% on the test set, demonstrating strong performance in sentiment-level classification. Precision and recall values were balanced across classes, with the Moderate sentiment achieving the highest recall (0.90) and Positive sentiment showing the highest precision (0.85).

The confusion matrix indicates that most errors occurred between Negative and Positive sentiments, reflecting acoustic similarities in certain emotional expressions. ROC analysis confirmed excellent separability, with AUC scores exceeding 0.90 for all classes.  
Training curves reveal stable accuracy and decreasing training loss, although validation loss fluctuations suggest mild overfitting. Overall, the CNN–RNN model provides an effective trade-off between accuracy and computational efficiency for sentiment grouping tasks.

**A graph of a graph of a training

AI-generated content may be incorrect.**

Figure :Accuracy and loss (CNN-RNN)

**A screenshot of a computer screen

AI-generated content may be incorrect.**

Figure : Classification report of sentiment CNN-RNN

|  |  |
| --- | --- |
| **A blue squares with white text  AI-generated content may be incorrect.** | **A graph of a curve  AI-generated content may be incorrect.** |

Figure : Confusion matrix and ROC of sentiment CNN-RNN

* **CNN-LSTM:**

The CNN–LSTM architecture achieved an overall accuracy of 81%, slightly lower than CNN–RNN but with improved recall for Positive sentiment (0.83). Moderate sentiment showed the highest recall (0.89), confirming the model’s ability to capture neutral tones.  
Confusion matrix analysis revealed that most errors occurred between Negative and Positive sentiments, consistent with acoustic similarities in emotional speech. ROC curves demonstrated strong separability, with AUC values above 0.90 for all classes**.**

Training curves excellent convergence for training accuracy and loss; however, rising validation loss after epoch 10 suggests overfitting. Overall, CNN–LSTM provides robust temporal modeling but at the cost of longer training time and higher complexity compared to CNN–GRU**.**

**A graph of a graph

AI-generated content may be incorrect.**

Figure : Accuracy and loss (CNN-LSTM)

**A screenshot of a computer

AI-generated content may be incorrect.**

Figure : Classification report of sentiment CNN-LSTM

|  |  |
| --- | --- |
| **A blue squares with white text  AI-generated content may be incorrect.** | **A graph of a curve  AI-generated content may be incorrect.** |

Figure : Confusion matrix and ROC of sentiment CNN-LSTM

**Ground true:** Pie charts illustrate error patterns for cross-sentiment misclassifications. Negative → Positive errors mainly involve Fearful (34%) and Angry (32%), while Positive →Negative errors are dominated by Happy (87%).

**A pie chart with different colored circles

AI-generated content may be incorrect.**

Figure : Misclassification Analysis for CNN–LSTM

* **CNN-ResNet:**

The CNN–ResNet model achieved 76.15% accuracy, lower than CNN–RNN and CNN–LSTM, but excelled in precision for Positive (0.82) and recall for Moderate (0.83) sentiments. Confusion matrix analysis showed most errors between Negative and Positive classes, consistent with other models. Training curves demonstrated steady convergence, though validation loss fluctuations indicated overfitting risks. Despite this, CNN–ResNet's superior feature extraction for Negative and Moderate sentiments makes it suitable for resource-rich deployment environments.

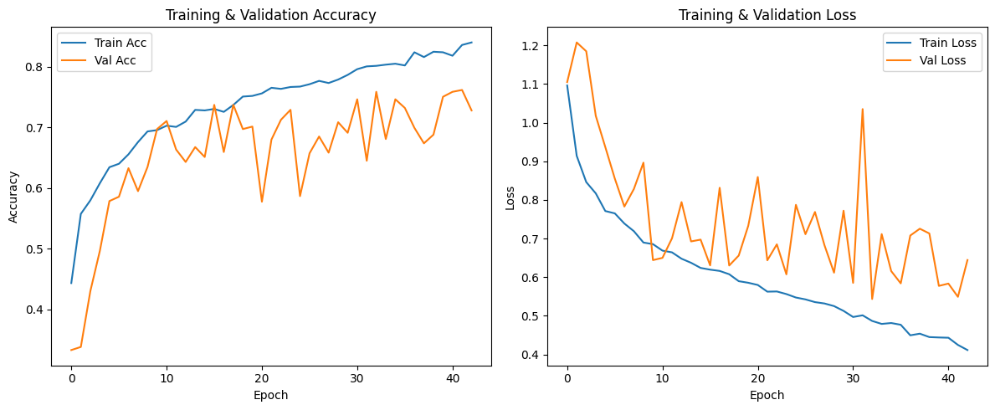
****

Figure : Accuracy and loss (CNN-ResNet)

**A screenshot of a computer screen

AI-generated content may be incorrect.**

Figure : Classification report of sentiment CNN-ResNet

**A blue squares with white text

AI-generated content may be incorrect.**

Figure : Confusion matrix of sentiment CNN-ResNet

### 6.5.5. Transfer Learning and Cross-Dataset Models

This section presents three advanced modelling approaches developed after the baseline CNN and CNN–RNN experiments:

* Wav2Vec2 fine-tuning for 8 emotion classification
* Wav2Vec2 base sentiment classification
* Hierarchical negative emotion recognition using CNN and ResNet

A fourth component evaluates cross-dataset to evaluate the general of model using a leave one dataset out (LODO) strategy.

#### a. Wav2Vec2 Fine-Tuned Model for Multi-Class Emotion Recognition

A Wav2Vec2-Base model was fine-tuned for 8-emotion classification (angry, happy, sad, neutral, fearful, disgust, surprised, calm). The model was trained on the unified, pre-processed audio dataset using a speaker-independent split (80% train, 10% validation, 10% test). Each audio clip was resampled to 16 kHz and processed with the Wav2Vec2 feature extractor. Training used the Adam optimizer (lr = 3e-5), batch size 32, FP16 mixed precision, and 50 epochs.

**Evaluation:**

The Wav2Vec2 model achieved an overall accuracy of 76% on the test set, outperforming traditional CNN-based baselines in feature representation. Precision and recall analysis revealed excellent performance for calm (F1 = 0.9) and angry (F1 = 0.82), while happy and surprise remained challenging due to limited samples and acoustic similarity.

Training curves indicate rapid convergence and strong initial generalization, though rising validation loss suggests overfitting. The confusion matrix confirms robust classification for high-energy emotions but highlight confusion between neutral and surprise.

Overall, Wav2Vec2 demonstrates the effectiveness of transformer-based architectures for speech emotion recognition, offering superior temporal and contextual modeling compared to CNN–RNN hybrids. Future improvements could include data augmentation, class balancing, and domain adaptation for cross-corpus robustness.

|  |  |
| --- | --- |
| A screenshot of a computer screen  AI-generated content may be incorrect.  Figure : Classification report of wav2vec2 (8 emotions) | A graph with numbers and symbols  AI-generated content may be incorrect.  Figure : Confusion matrix wav2vec2 (8 emotions) |

A graph of different colored lines

AI-generated content may be incorrect.

Figure : Validation accuracy over epochs for wav2vec2

**Testing the model with 3 sentiments as output:**

To assess the robustness of the fine-tuned Wav2Vec2 model, its eight emotion predictions were mapped into three sentiment categories: Positive (happy, surprised), Moderate (neutral, calm), and Negative (angry, sad, fearful, disgust). Instead of retraining, raw predictions were post-processed using this mapping, aggregating per-emotion probabilities into sentiment-level scores. The final sentiment was assigned based on the highest aggregated probability.

Evaluation using a speaker-independent test set produced a classification report and a 3×3 confusion matrix. Results showed that sentiment-level prediction was more stable than fine-grained emotion classification, reducing ambiguity and improving reliability. Positive and Negative sentiments were recognized accurately, while Moderate showed slightly higher confusion due to acoustic similarity with neighbouring emotions. These findings confirm that Wav2Vec2 maintains strong affective discrimination even when repurposed for coarse sentiment recognition without retraining.

A diagram of a negative reaction

AI-generated content may be incorrect.

Figure : Confusion matrix for wav2vec2 sentiment-level

The confusion matrix confirms that Wav2Vec2 generalizes well when repurposed for sentiment-level classification. Negative and positive sentiments are recognized with high reliability, while neutral shows minor overlap with negative. This supports the conclusion that coarse-grained sentiment grouping improves stability and reduces ambiguity compared to fine-grained emotion classification.

#### b. Cross-Dataset Generalization via Leave-One-Dataset-Out (LODO)

To assess cross-corpus robustness, a leave-one-dataset-out (LODO) strategy was employed. In each configuration:

* The model was trained on three datasets and tested on the remaining unseen dataset.
* This process was repeated for RAVDESS, TESS, SAVEE, and CREMA-D.
* The classifier architecture matched the Wav2Vec2 fine-tuned emotion model, ensuring consistency across experiments.

**Evaluation:**

This figure presents accuracy, precision, recall, and F1-score for each test dataset under the LODO strategy.

* **Accuracy:** Highest for RAVDESS and SAVEE (≈0.465), lowest for TESS (0.388).
* **Precision:** Peaks for TESS (0.571), indicating strong bias toward certain acoustic features despite low recall.
* **Recall:** Mirrors accuracy trends, with SAVEE and RAVDESS performing best.
* **F1-score:** SAVEE achieved the highest (0.459), while TESS was lowest (0.353).

A screenshot of a graph

AI-generated content may be incorrect.

Figure : Summary table of LODO result

The table consolidates all metrics for quick comparison. Performance degradation on unseen datasets confirms domain mismatch challenges caused by differences in language, speaker demographics, and recording conditions.

A group of colored bars

AI-generated content may be incorrect.

Figure : LODO cross-dataset performance comparison

#### c. Wav2Vec2 Sentiment Classification (3 Classes)

A custom Wav2Vec2 sentiment model was trained by freezing all encoder layers and learning only a lightweight classification head (768 → 3). This approach enabled fast adaptation and minimized overfitting. A sentiment grouping layer was added before the output layer to classify audio into three categories: Negative, Neutral, and Positive.

**Results:**

The report shows outstanding performance across all sentiment classes, with overall accuracy of 96%. Precision, recall, and F1-scores are consistently high:

A screenshot of a computer

AI-generated content may be incorrect.

Figure : Classification report for Wav2Vec2 sentiment model

The confusion matrix confirms strong classification reliability. Negative (424 correct), Neutral (400 correct), and Positive (432 correct) show minimal misclassification, with only minor overlap between Neutral and Negative.

A blue squares with white text

AI-generated content may be incorrect.

Figure : Confusion matrix for Wav2Vec2 sentiment model

The custom Wav2Vec2 sentiment model demonstrates exceptional accuracy and robustness, outperforming previous CNN-based architectures. Freezing encoder layers while fine-tuning a small classification head proved effective for rapid adaptation and reduced overfitting. High precision and recall across all sentiment classes indicate strong generalization and suitability for real-world deployment.

After fine-tuning the Wav2Vec2 model for sentiment-level classification, the same architecture was adapted to predict eight emotion classes. The previous model was simply reloaded and applied to the full preprocessed dataset. Using a standardized evaluation pipeline the model achieved 95.29% overall accuracy.

All eight emotion classes maintained high F1-scores (≥ 0.93), with angry (F1 = 0.972) and surprised (F1 = 0.971) performing best. The calm class showed slightly lower recall (0.920), likely due to its smaller sample size. The confusion matrix indicates minimal cross-emotion confusion and well-preserved class boundaries, confirming that the fine-tuned Wav2Vec2 model retains strong fine-grained emotion discriminability even after being used for sentiment inference. This demonstrates that sentiment-level prediction does not degrade the original model’s emotion recognition capability.

A screenshot of a computer screen

AI-generated content may be incorrect.

Figure : Classification Report for Wav2Vec2 Eight-Emotion

A graph of emotions

AI-generated content may be incorrect.

Figure : Confusion Matrix for Wav2Vec2 Eight-Emotion

#### d. Hierarchical Negative-Emotion Recognition

This section presents a hierarchical two-stage speech affect recognition pipeline. In the first stage, a fine-tuned Wav2Vec2 sentiment classifier predicts one of three coarse sentiment categories (Negative, Neutral, Positive). In the second stage, audio samples predicted as Negative are forwarded to a spectrogram-based CNN model to perform fine-grained emotion recognition among four negative emotions (angry, sad, fearful, disgust). This hierarchical architecture is designed to reduce confusion across wide emotional categories and to improve discrimination within high-arousal negative states.

Stage 1: Sentiment prediction using fine-tuned wav2vec2

A custom three-class sentiment classifier was developed by attaching a lightweight linear classification head to a frozen Wav2Vec2-Base encoder. The architecture computes the temporal average of the encoder’s hidden states and feeds the pooled vector into a linear layer predicting three sentiment classes. All samples predicted as Negative are extracted and saved to csv file. These samples (1643 samples) are the input to the second-stage negative-emotion classifier.

A graph with blue squares and white text

AI-generated content may be incorrect.

Figure : Prediction for next stage

Stage 2: Filter spectrogram and fine-grained negative emotion classification

After obtaining sentiment predictions from the Wav2Vec2-based model, all test samples predicted as Negative were isolated for a second-stage fine-grained classification. This hierarchical design reduces the complexity of the downstream task and focuses computational resources only on emotionally dense segments (angry, sad, fearful, disgust).

To support this stage, each negative sample was transformed into two complementary spectrogram channels. A low-pass (<3 kHz) spectrogram captures prosodic cues such as pitch, energy contour, and vowel resonance, while a high-pass (>3 kHz) spectrogram isolates turbulent noise, vocal tension, and other high-frequency components associated with anger or fear. These dual-band features were saved as Mel-spectrogram tensors and combined to form a 2-channel input representation.

These paired spectrograms were then used to train a dedicated negative-emotion classifier, implemented using a ResNet-18 backbone adapted for two-channel audio features. The convolutional layers extract localized spectral patterns from both bands, while a multi-head attention module aggregates temporal dependencies across the full utterance. A lightweight feed-forward classifier predicts one of four negative emotions (angry, sad, fearful, disgust).

**Result and evaluation:**

**A screenshot of a computer screen

AI-generated content may be incorrect.**

Figure : Classification Report for ResNetAudio Model

A diagram of emotions

AI-generated content may be incorrect.

Figure : Confusion Matrix for ResNetAudio Model

The ResNetAudio model achieved 74.16% accuracy, which is lower than transformer-based approaches but competitive for CNN-based architectures. The model demonstrates robust feature extraction for high-energy emotions like angry, while lower recall for disgust and fear suggests challenges in distinguishing subtle emotional cues. Misclassification patterns highlight the need for:

* Class balancing to address underrepresented emotions.
* Advanced augmentation to improve generalization.
* Potential integration of temporal modeling to capture sequential dynamics.

## 6.6. GUI Implementation

Provide a simple, responsive interface to record speech, transcribe it, and display the detected emotion in a chat-style UI.

**Tech stack:**

* Streamlit for UI and routing (multi‑page app).
* PyTorch + torchaudio as backend.
* Whisper (OpenAI) for ASR.
* Hugging Face Transformers (AutoFeatureExtractor + AutoModelForAudioClassification) with a Wav2Vec2‑based emotion classifier.
* st\_audiorec for in‑browser microphone recording.
* Custom CSS

**Application structure:**

Home page: Introduce the SER system and use cases

* Title and short overview
* Instructions of using web app

Data collection page: Summarize datasets used for training and evaluation

* Short descriptions of RAVDESS, TESS, CREMA-D, SAVEE.
* Sample counts, audio specs, and dataset links.
* Combined dataset summary (total samples, classes, sample rates).

EDA page: Present exploratory analysis

* Class distributions, duration histograms.
* Example spectrograms/MFCCs.
* Tabs for dataset-specific views.

Models page: Explain model architecture and performance

* Model training
* Evaluation metrics
* Pipeline overview

Prediction page: record audio and run emotion analysis

* User records speech via st\_audiorec
* Silence/quality checks prevent unnecessary inference
* Model from saved in HuggingFace
* Results displayed in a chat-style message bubble

# 7. Conclusion

## 7.1. Knowledge Gained

This project provided both practical and theoretical insights into Speech Emotion Recognition. Important experience was gained in dataset construction, audio preprocessing, feature extraction, and managing the challenges of training deep learning models on multiple emotional speech corpora. The work involved applying a range of architectures, from convolutional neural networks to transformer-based models such as Wav2Vec2, as well as designing custom CNN including attention hybrids and incorporating signal, processing methods such as low-pass and high-pass filtering to improve emotion discrimination.

The project also strengthened understanding of full-pipeline development, including model evaluation, deployment for real-time prediction, and user-interface design through a Streamlit application. A central theme throughout this work was making informed decisions about feature representation. This requires comparing traditional handcrafted features such as MFCCs with learned time-frequency representations such as log-Mel spectrograms. The selection process involved analyzing discriminative power across different frequency regions, evaluating stability under noise, and ensuring compatibility between features and model architectures.

## 7.2. Result

This project successfully developed a comprehensive Speech Emotion Recognition (SER) pipeline, integrating data preparation, preprocessing, feature extraction, and model training. The system demonstrated strong performance across multiple architectures, with log-Mel spectrograms emerging as the most reliable features for CNN-based models and Wav2Vec2 achieving the highest overall accuracy. A hierarchical two-stage design was implemented, combining sentiment classification with fine-grained negative-emotion recognition to enhance discrimination among acoustic similar emotions.

Additionally, a functional real-time interface was developed using Streamlit, enabling users to record speech, view transcriptions, and receive emotion predictions through an intuitive chat-style UI. While the system performs well across diverse datasets and exhibits robust sentiment-level classification, further refinement is needed to improve cross-corpus generalization and fine-grained emotion recognition.

## 7.3. Limitation

Despite promising results, several constraints remain:

* Performance varies across speakers and recording conditions due to dataset imbalance and acoustic variability.
* Cross-corpus generalization is limited, particularly for emotions with subtle acoustic cues.
* Real-time performance depends on hardware capabilities, especially for transformer-based inference.
* The system currently supports only speech; multimodal cues (facial, contextual, physiological) are not incorporated.

## 7.4. Further Development

Future improvements will expand the system beyond SER into a comprehensive emotion-aware conversational framework:

* Chatbot integration: Combine SER with NLP to enable emotionally adaptive dialogue.
* Real-time interaction: Reduce latency for deployment in assistants, social robots, or therapy tools.
* Multimodal emotion recognition: Integrate facial expressions and physiological signals for higher robustness.
* Domain adaptation: Improve cross-language and cross-corpus transfer using adversarial learning and self-supervised methods.
* Application prototypes: Explore real-world use cases in mental health support, customer service, and education.

# Reference

Afifah, I.N., Santoso, T.B. and Dutono, T. (2025). Indonesian speech emotion recognition: feature extraction and neural network approaches. *International Journal of Electrical and Computer Engineering (IJECE)*, 15(4), p.3769. Available at: https://doi.org/10.11591/ijece.v15i4.pp3769-3778.

Alluhaidan, A.S., Saidani, O., Jahangir, R., Nauman, M.A. and Neffati, O.S. (2023). Speech Emotion Recognition through Hybrid Features and Convolutional Neural Network. *Applied Sciences*, 13(8), p.4750. Available at: https://doi.org/10.3390/app13084750.

Aouani, S., Bouguila, N. and Alimi, A.M. (2020). Speech emotion recognition using HNR, MFCC, ZCR and TEO with SVM classifier. *Procedia Computer Science*, 176, pp.1461–1470. Available at: https://doi.org/10.1016/j.procs.2020.09.221.

Barhoumi, C. and BenAyed, Y. (2024). Real-time speech emotion recognition using deep learning and data augmentation. *Artificial Intelligence Review*, 110, pp.1–30. Available at: <https://link.springer.com/article/10.1007/s10462-024-11065-x>.

Boehm, B. W. (1988). “A spiral model of software development and enhancement”. In: Computer 21.5, pp. 61–72. DOI: 10.1109/2.59.

Chowdhury, J.H., Ramanna, S. and Kotecha, K. (2025). Speech emotion recognition with lightweight deep neural ensemble model using handcrafted features. *Scientific Reports*, 15, Article 11824. Available at: https://doi.org/10.1038/s41598-025-95734-z.

‌Gideon, J., McInnis, M. and Mower Provost, E. (2019). Improving Cross-Corpus Speech Emotion Recognition with Adversarial Discriminative Domain Generalization (ADDoG). *IEEE Transactions on Affective Computing*, pp.1–1. Available at: https://doi.org/10.1109/taffc.2019.2916092.

Hema, C. and Marquez, F.P.G. (2023). Emotional speech recognition using CNN and Mel-value MFCC with SVM classifier. *Applied Acoustics*, 211. Available at: https://doi.org/10.1016/j.apacoust.2023.109492.

‌Ho, N.-H., Yang, H.-J., Kim, S.-H. and Lee, G. (2020). Multimodal Approach of Speech Emotion Recognition Using Multi-Level Multi-Head Fusion Attention-Based Recurrent Neural Network. *IEEE Access*, 8, pp.61672–61686. Available at: <https://doi.org/10.1109/access.2020.2984368>.

Kaloub, A. and Abed Elgabar, E. (2025). Speech-based techniques for emotion detection in natural Arabic audio files. *The International Arab Journal of Information Technology (IAJIT)*, 22(1), pp.139–157. Available at: https://doi.org/10.34028/iajit/22/1/1.

Koduru, A., Valiveti, H.B. and Budati, A.K. (2020). Feature extraction algorithms to improve the speech emotion recognition rate. *International Journal of Speech Technology*, 23, pp.45–55. Available at: https://doi.org/10.1007/s10772-020-09672-4.

‌Latif, S., Rana, R., Khalifa, S., Jurdak, R. and Schuller, B.W. (2022). Self-Supervised Adversarial Domain Adaptation for Cross-Corpus and Cross-Language Speech Emotion Recognition. *IEEE Transactions on Affective Computing*, pp.1–1. Available at: https://doi.org/10.1109/taffc.2022.3167013.

Madanian, S., Chen, T., Adeleye, O., Templeton, J.M., Poellabauer, C., Parry, D. and Schneider, S.L. (2023). Speech emotion recognition using machine learning — A systematic review. *Intelligent Systems with Applications*, 20, p.200266. Available at: https://doi.org/10.1016/j.iswa.2023.200266.

Manifesto, Agile (2001). Manifesto for agile software development.

Martin, James (1991). Rapid application development. Macmillan Publishing Co., Inc.

Meng, H., Yan, T., Yuan, F. and Wei, H. (2019). Speech emotion recognition from 3D log-Mel spectrograms with deep learning network. *IEEE Access*, pp.1–12.

Mustaqeem, Sajjad, M. and Kwon, S. (2020). Clustering-based speech emotion recognition by incorporating learned features and deep BiLSTM. *IEEE Access*, pp.1–12.

Oppenheim, A.V. and Schafer, R.W., 2010. *Discrete-Time Signal Processing*. 3rd ed. Pearson.

‌Pulatov, I., Oteniyazov, R., Makhmudov, F. and Cho, Y.-I. (2023). Enhancing Speech Emotion Recognition Using Dual Feature Extraction Encoders. *Sensors*, [online] 23(14), p.6640. Available at: https://doi.org/10.3390/s23146640.

‌Rayhan Ahmed, Md., Islam, S., Muzahidul Islam, A.K.M. and Shatabda, S. (2023). An ensemble 1D-CNN-LSTM-GRU model with data augmentation for speech emotion recognition. *Expert Systems with Applications*, 218, p.119633. Available a t: <https://doi.org/10.1016/j.eswa.2023.119633>.

Royce, Winston W. (1970). “Managing the Development of Large Software Systems”. In: Proceedings of IEEE WESCON. Originally published as a technical report at TRW, pp. 1–9.

Zhao, J., Mao, X. and Chen, L. (2019). Speech emotion recognition using deep 1D & 2D CNN LSTM networks. *Biomedical Signal Processing and Control*, 52, pp.101–110. Available at: <https://doi.org/10.1016/j.bspc.2018.08.035>.

# Appendix

Streamlit link: [Streamlit](https://sergui.streamlit.app/)

GitHub link: [anhhong225/Chatbot](https://github.com/anhhong225/Chatbot)