By
Batch B - Team 17
Speech Processing (Professional Elective) - 22AIE405

On
March 8, 2025

Multi-feature and multi-model comparative study of

Speech Emotion Recognition

Introduction & Problem Statement

Speech Emotion Recognition (SER) is a vital task in **speech signal processing**, aiming to classify emotions from audio signals.

- SER remains challenging due to **variability in speech patterns, speaker differences, and dataset imbalances**.
- To address these challenges we have done a comparative study with multiple features and multiple models. This analysis helps us to identify the most effective feature model combinations.

Research Paper	Model	Accuracy	Publishing Year
Speech Emotion Recognition with SVM, KNN, and Deep SVM	Deep SVM	82.6	2023
Real-Time Speech Emotion Recognition Using Deep Learning	LSTM + Mel Spectrogram	87.3	2021
Speech emotion recognition based on HMM and SVM	HMM + SVM	88.9	2021
Multi-Modal Speech Emotion Recognition Using Audio and Text Features	Bi LSTM + CNN	89.7	2023
End-to-End Speech Emotion Recognition Using Attention Mechanisms	Attention based RNN's	91.2	2023
Multi-Model Emotion Recognition from Speech Using StarGAN, DCNN, and SVM	SVM + DCNN	96.8	2025
Multi-Task Learning for Speech Emotion and Speaker Recognition	Shared CNN-LSTM Architecture	87.6	2023
End-to-End Multi-Language Speech Emotion Recognition	Transformer-Based Architecture	90.4	2023

Why we chose SVM, XGBOOST, DCNN, LSTM?

https://arxiv.org/pdf/2002.07590 **- SVM Supporting paper**
This paper suggests that SVM can be used in speech emotion recognition to achieve satisfactory results, Instead of taking complex deep learning architectures if feature selection is done properly.

https://journals.plos.org/plosone/article?id=10.1371/journal.pone.0267132 **- XG-BOOST Supporting Paper**
XGBoost is used in SER for its ability to handle complex, high-dimensional feature sets and prevent overfitting through regularization, making it well-suited for learning from limited or imbalanced emotional speech data.

https://doi.org/10.1016/j.specom.2019.06.001 **- DCNN Supporting Paper**
DCNNs are highly effective in SER as they automatically learn hierarchical representations from spectrogram-like features, capturing both local and global emotional patterns in speech.

https://doi.org/10.1109/ACCESS.2019.2945174 - LSTM Supporting Paper
LSTMs are well-suited for SER as they effectively model the temporal dependencies and sequential nature of speech signals, enabling better recognition of emotional dynamics over time.

Tech Stack.




2

Database Used

About the Dataset.

RAVDESS (Ryerson Audio-Visual Database of Emotional Speech and Song):

- 1440 files (audio and video) from 24 actors (12 male, 12 female).
- Emotions: Neutral, calm, happy, sad, angry, fearful, disgust, surprise (with 2 intensity levels).
- File naming: Encodes modality, emotion, intensity, statement, repetition, and actor.
- 48kHz

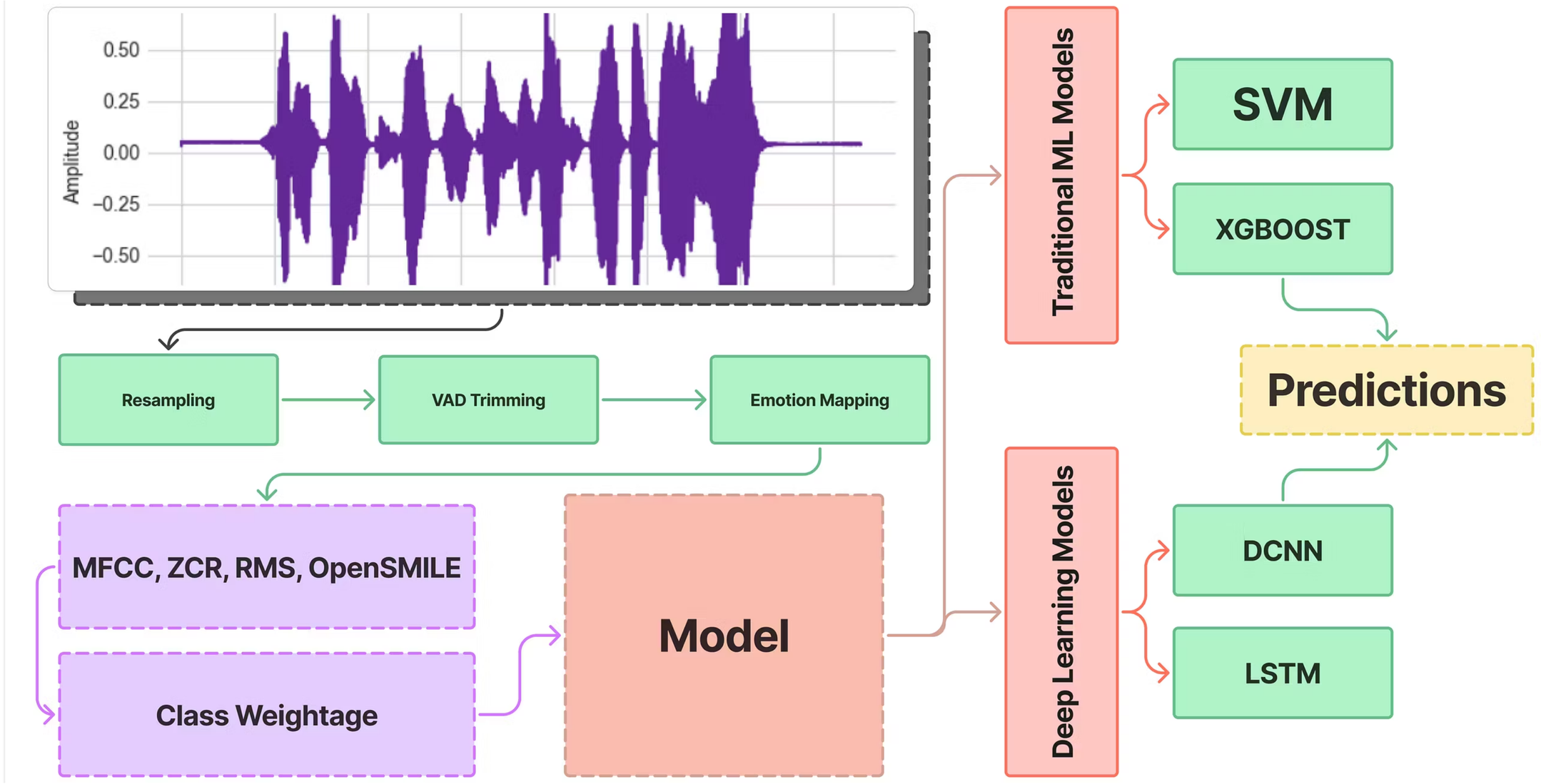
SAVEE (Surrey Audio-Visual Expressed Emotion):

- 480 audio files from 4 male actors.
- Emotions: Neutral, happy, sad, angry, fearful, disgust, surprise.
- File naming: First two characters represent the emotion.
- Each actor recorded 15 utterances per emotion.
- 16 kHz

3

Methodology

Model Pipeline

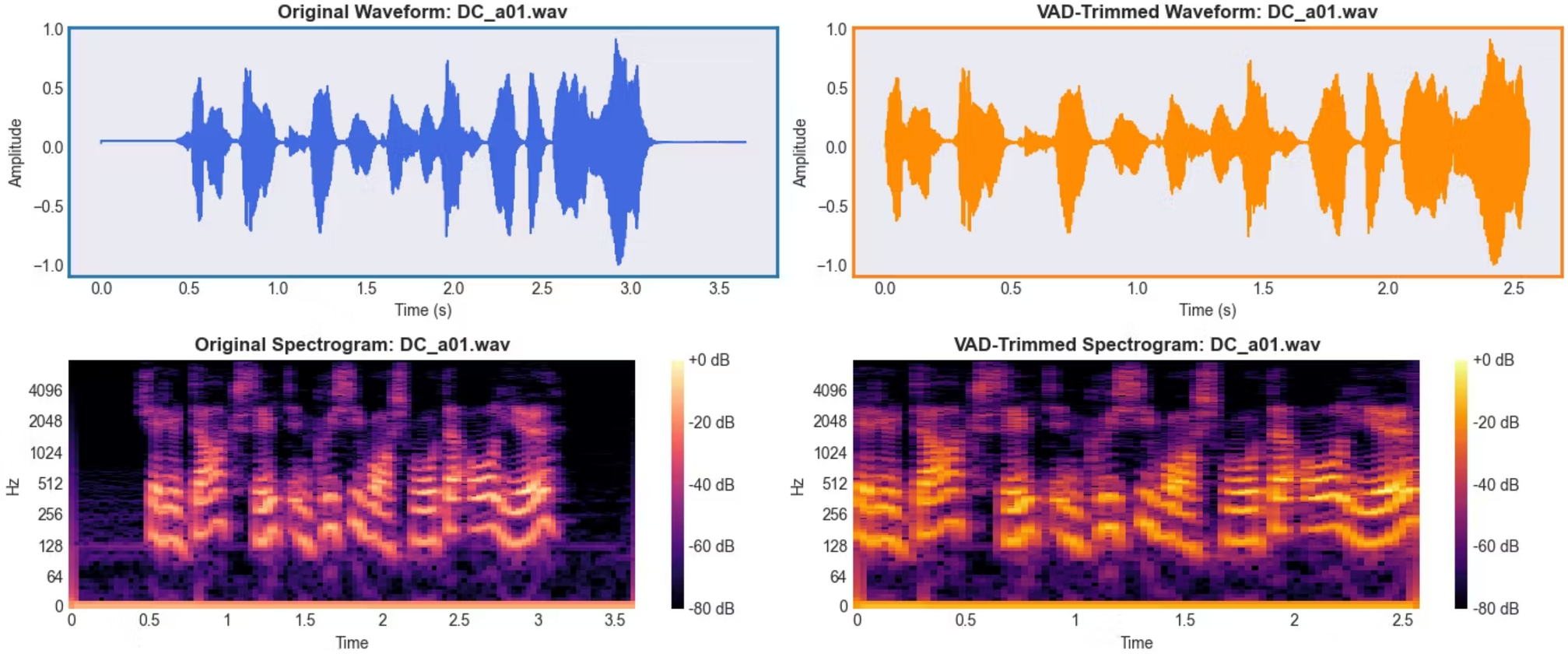


4

Implementation

Step 1 - Loading and preprocessing the data

We take the dataset, load them and down sample them to 16000 kHz. Then we proceed to perform voice activity detection and eliminate the unvoiced regions



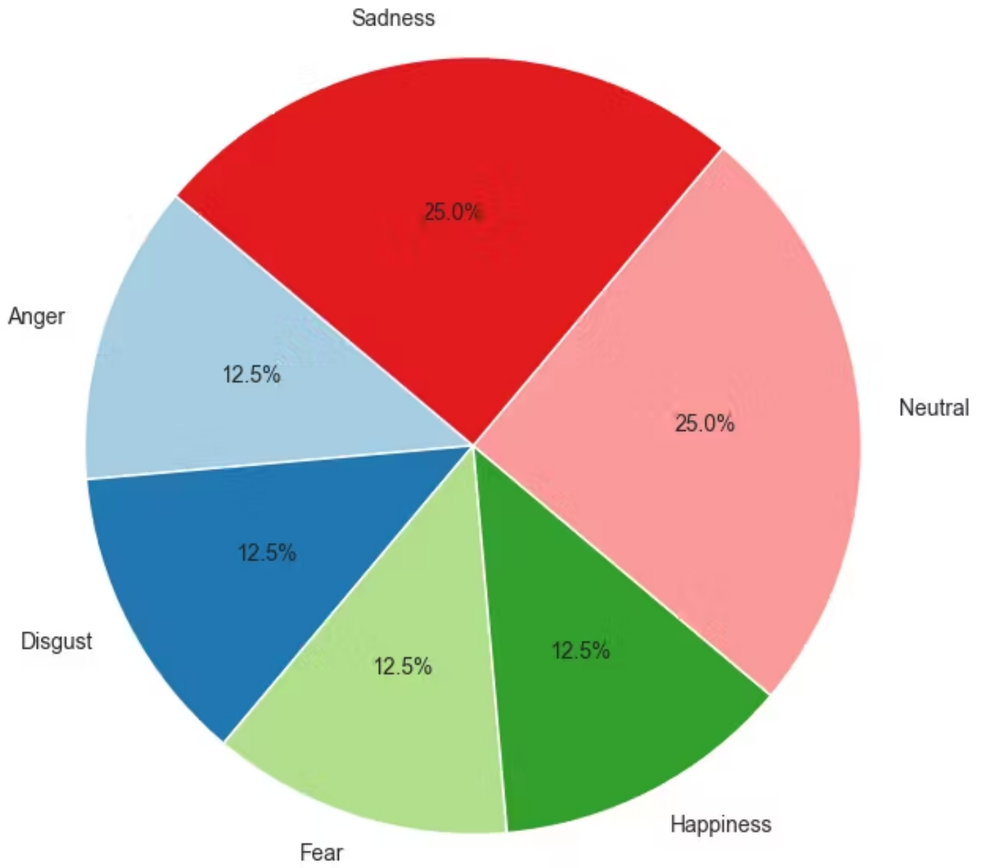
Step 2 - Feature extraction and emotion label mapping

After preprocessing, we take the dataset and define emotion label mapping. Once this is done we now can extract all features (MFCC 40, Root mean square energy, Zero crossing rate. Upon extraction they are normalised and concatenated into a input vector

Try Pitch

Step 3 - Checking class distribution and assigning class weightage to ensure balance.

We are using Class weightage to assign higher weightage to classes with lower samples so that the model does not show poor performance due to imbalance



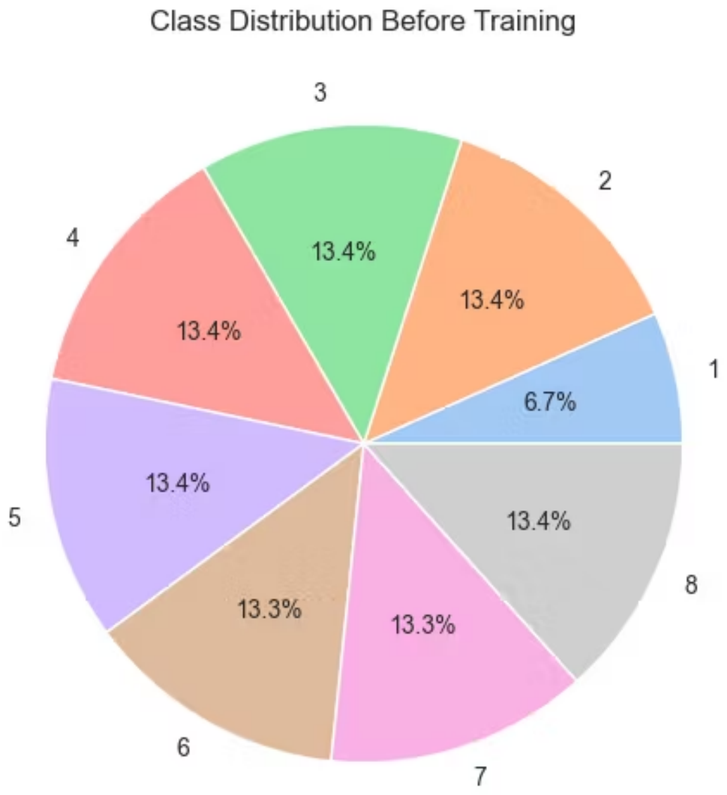
Class distribution pie-chat

```
Class Weights: {0: 1.3333333333333333,
                1: 1.3333333333333333,
                2: 1.3333333333333333,
                3: 1.3333333333333333,
                4: 0.6666666666666666,
                5: 0.6666666666666666}
```

Class weights assigned to each class to ensure balance

Step 3 - Checking class distribution and assigning class weightage to ensure balance.

We are using Class weightage to assign higher weightage to classes with lower samples so that the model does not show poor performance due to imbalance



Class distribution pie-chat

```
class_weights = {
    1: 1.87,2: 0.93,
    3: 0.93,4: 0.93,
    5: 0.93,6: 0.94,
    7: 0.94,8: 0.93,
}
```

Class weights assigned to each class to ensure balance

Step 4 - Model Training and evaluation for SAVEE

We are using a 80:20 training and testing split

- In SAVEE 384 samples are set up for training and 96 samples are set up for testing
- In RAVDESS there are 1152 samples for training and 288 samples for testing
- We are using **GRID SEARCH CV** method to find out the best set of hyperparameters.
- We are using **OPTUNA** library to find out the best set of hyperparameters for deep learning models.

BEST HYPER-PARAMETERS

SVM	C = 10, Gamma = 'scale', Kernel = RBF
XGBoost	Learning Rate = 0.05, Max Depth = 9, n_estimators = 300
GMM	Components = 8, Covariance Type = Full
DCNN	Learning Rate = 0.001, Dropout = 0.3, Layers = 4
LSTM	Hidden Dim = 128, Dropout = 0.25, Learning Rate = 0.002

4

Results

SAVEE - Results

Models	Features	Accuracy	F1 Score	Precision	Recall
SVM	MFCC 13	42.71	0.33	0.36	0.35
	MFCC 40	52.08	0.47	0.65	0.45
	MFCC DD	25	0.16	0.17	0.18
	MFCC 40, ZCR, RMS	55.2	0.51	0.68	0.48

Models	Features	Accuracy	F1 Score	Precision	Recall
XGBOOST	MFCC 13	44.79	0.35	0.38	0.38
	MFCC 40	57.29	0.53	0.58	0.52
	MFCC DD	38	0.28	0.33	0.3
	MFCC 40, ZCR, RMS	59.38	0.54	0.67	0.53

Models	Features	Accuracy	F1 Score	Precision	Recall
DCNN	MFCC 40, ZCR, RMS	71.45	0.65	0.67	0.65
	OpenSmile	60.56	0.56	0.55	0.59

Models	Features	Accuracy	F1 Score	Precision	Recall
LSTM	MFCC 40, ZCR, RMS	47.29	0.40	0.39	0.39
	OpenSmile	40.83	0.32	0.47	0.41

RAVDESS - Results

Models	Features	Accuracy	F1 Score	Precision	Recall
SVM	MFCC 13	59.03	0.59	0.6	0.59
	MFCC 40	61.46	0.61	0.62	0.62
	MFCC DD	58.33	0.58	0.59	0.58
	MFCC 40, ZCR, RMS	62.85	0.62	0.63	0.63

Models	Features	Accuracy	F1 Score	Precision	Recall
XGBOOST	MFCC 13	53.12	0.52	0.54	0.52
	MFCC 40	48.26	0.47	0.47	0.47
	MFCC DD	48	0.46	0.48	0.47
	MFCC 40, ZCR, RMS	49.65	0.47	0.48	0.48

Models	Features	Accuracy	F1 Score	Precision	Recall
DCNN	MFCC 40, ZCR, RMS	94.44	93.80	94.55	93.40
	OpenSmile	91.15	91.05	91.97	90.71

Models	Features	Accuracy	F1 Score	Precision	Recall
LSTM	MFCC 40, ZCR, RMS	85.76	0.86	0.85	0.85
	OpenSmile	80.73	0.80	0.80	0.80

5

Inference

Inference

1. **Feature Effectiveness**
   - *MFCC DD* underperformed across the board, suggesting limited discriminative power.
   - *OpenSMILE*, though not always best, gave strong results with LSTM, proving its versatility.

2. **Best Model-Feature Combos**
   - *DCNN + MFCC 40 + ZCR + RMS* delivered top results, especially on RAVDESS (94.44% accuracy).
   - *LSTM + OpenSMILE* worked best on SAVEE, highlighting LSTM's strength with rich temporal features.

3. **Impact of Dataset Size**
   - *SAVEE's smaller size* hurt LSTM's performance.
   - *RAVDESS*, being larger and more balanced, enabled deeper models to shine.

4. **Model Behavior**
   - *DCNN consistently outperformed LSTM*, excelling in learning spatial patterns.
   - *XGBoost* performed well on SAVEE but overfitted on RAVDESS, needing better regularization.

5. **Metric Insights**
   - High *accuracy correlated with strong F1 scores*, showing overall balance.
   - Some ML models had *high precision but low recall*, pointing to conservative predictions.

Thank you

Pitch

Want to make a presentation like this one?

Start with a fully customizable template, create a beautiful deck in minutes, then easily share it with anyone.

Create a presentation (It's free)