Dog Voice Translator Using Deep Learning

Final Project Report

Team 36

1. Introduction

Communication between humans and animals is limited by the absence of a shared language. Dogs express their emotional states through body language and vocalizations such as barks, growls, whining, or silence. However, humans often misinterpret these sounds, leading to confusion, delayed responses, or unrecognized distress signals.

This project explores the use of **deep learning and audio signal processing** to translate dog vocalizations into emotion-based human-readable interpretations. Using the **Mascellina Dog Vocalization Dataset**, a Convolutional Neural Network (CNN) was trained to classify bark audio into emotional categories and provide text-based feedback to users.

2. Problem Statement

Dogs communicate emotions through vocal sounds, but humans struggle to interpret them accurately. Existing technology does not provide a reliable or accessible translation mechanism using real acoustic patterns from dogs.

3. Motivation

This project aims to bridge the communication gap between humans and dogs by analyzing vocal patterns and mapping them to emotional categories. Improved interpretation can enhance dog welfare, owner responsiveness, and pet–human bonding.

4. Related Work

Previous studies in animal sound processing mainly focus on:

- Bird species identification
- Whale communication decoding
- Dog-bark recognition using classical ML (MFCC + SVM)

However, **emotion-based dog bark translation using deep learning** remains an underexplored research area, especially with real-world datasets.

5. Dataset

We used the **Mascellina Dataset**, a publicly available research dataset containing labeled dog vocalizations across categories such as alert, fear, playfulness, aggression, loneliness, warning, curiosity, and silence.

Key Properties:

Feature	Value
Total samples	~9,500+ processed segments
Classes	17 labeled emotional categories
Sampling rate	Standardized to 16 kHz
Format	.wav

RESEARCH PAPER: https://www.sciencedirect.com/science/article/pii/S0957417424020803

6. Dataset Challenges & Handling

A major challenge was **class imbalance** — the *silence (S)* class contained nearly half of all samples.

Instead of deleting this class, we managed imbalance through:

Stratified train-test split

Ensures each class appears in the same proportion in both sets.

Example:

If a class is **10% of the dataset**, it remains **10% in both train and test**.

Balanced evaluation

We did not rely solely on accuracy — macro F1 score and per-class behavior were observed.

7. Methodology

7.1 Preprocessing Pipeline

Step	Description
Resampling	All audio resampled to **16 kHz**
Noise Reduction	Applied noisereduce filtering
Normalization	Standard amplitude normalization
Segmentation	Audio split into **2-second chunks**
Feature Extraction	MFCCs, spectral contrast, chroma, RMS

7.2 Feature Engineering

Extracted features were stacked into a unified **(60 × 100)** representation, where:

- **60 features = MFCC + chroma + spectral contrast + RMS**
- **100 frames = time scaling window**

This ensures constant input size for the neural network.

7.3 CNN Model Architecture

Layer	Details
Conv2D + ReLU	Extracts local time–frequency patterns
Batch Normalization	Stabilizes and accelerates learning

MaxPooling	Reduces spatial complexity
Conv2D (64 + 128 filters)	Deeper feature extraction
Global Average Pooling	Converts feature maps to feature vector
Dense (128 units + dropout)	High-level pattern learning
Output Softmax	Predicts emotion label

Optimizer: **Adam**
Loss: **Sparse categorical cross-entropy**

8. Experiments

We initially attempted an alternative approach using **CRNN + dataset balancing**, but:

- Upsampling minority classes created synthetic noise
- Model overfit rapidly
- Evaluation degraded

Thus, the final model reverted to the CNN architecture with original class distribution preserved.

9. Results

Classification Metrics:

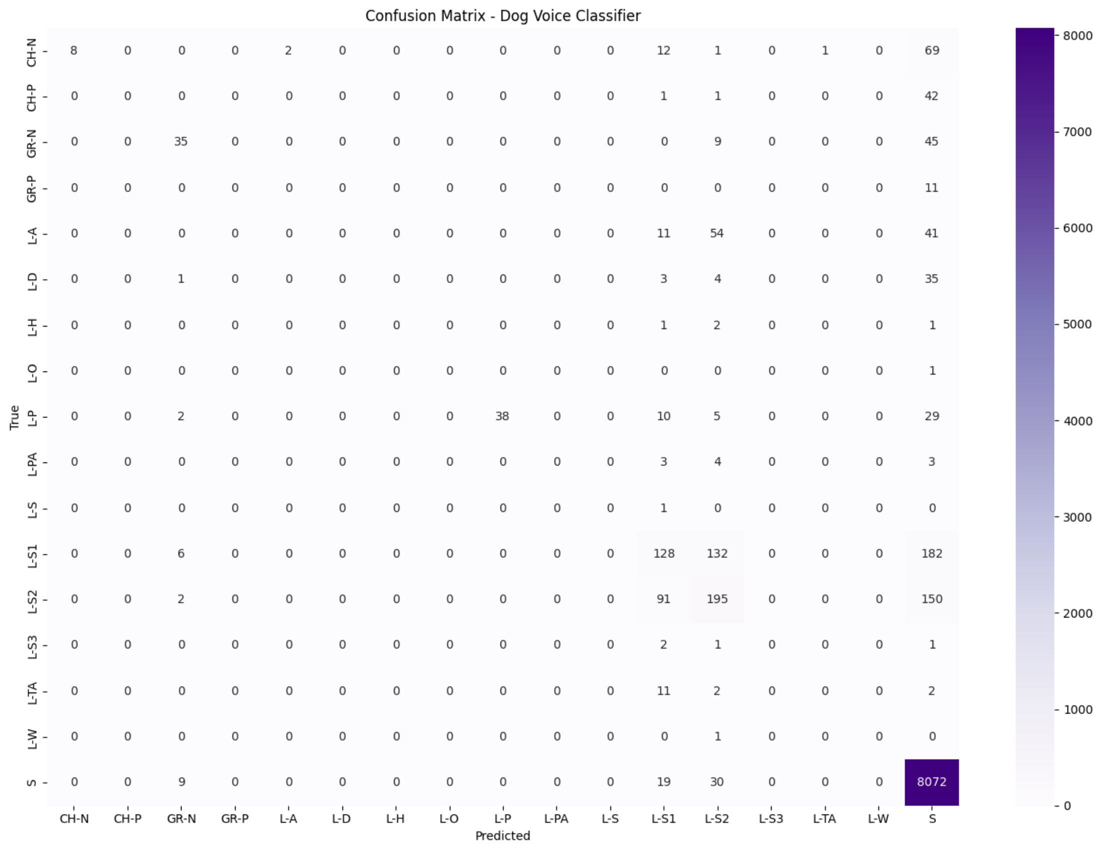
	precision	recall	f1-score	support
CH-N	1.000	0.086	0.158	93
CH-P	0.000	0.000	0.000	44
GR-N	0.636	0.393	0.486	89
GR-P	0.000	0.000	0.000	11

L-A	0.000	0.000	0.000	106
L-D	0.000	0.000	0.000	43
L-H	0.000	0.000	0.000	4
L-O	0.000	0.000	0.000	1
L-P	1.000	0.452	0.623	84
L-PA	0.000	0.000	0.000	10
L-S	0.000	0.000	0.000	1
L-S1	0.437	0.286	0.345	448
L-S2	0.442	0.445	0.444	438
L-S3	0.000	0.000	0.000	4
L-TA	0.000	0.000	0.000	15
L-W	0.000	0.000	0.000	1
S	0.930	0.993	0.960	8130

Accuracy			0.890	9522
Macro Avg	0.261	0.156	0.177	9522
Weighted Avg	0.859	0.890	0.868	9522

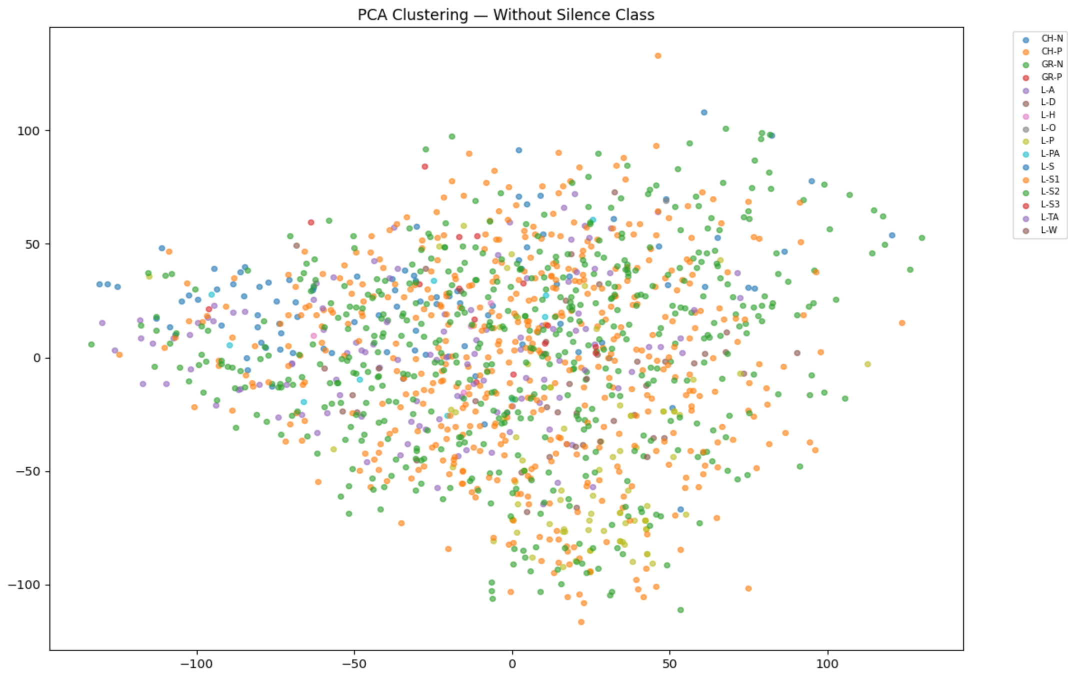
9. Visualization:

CONFUSION MATRIX:

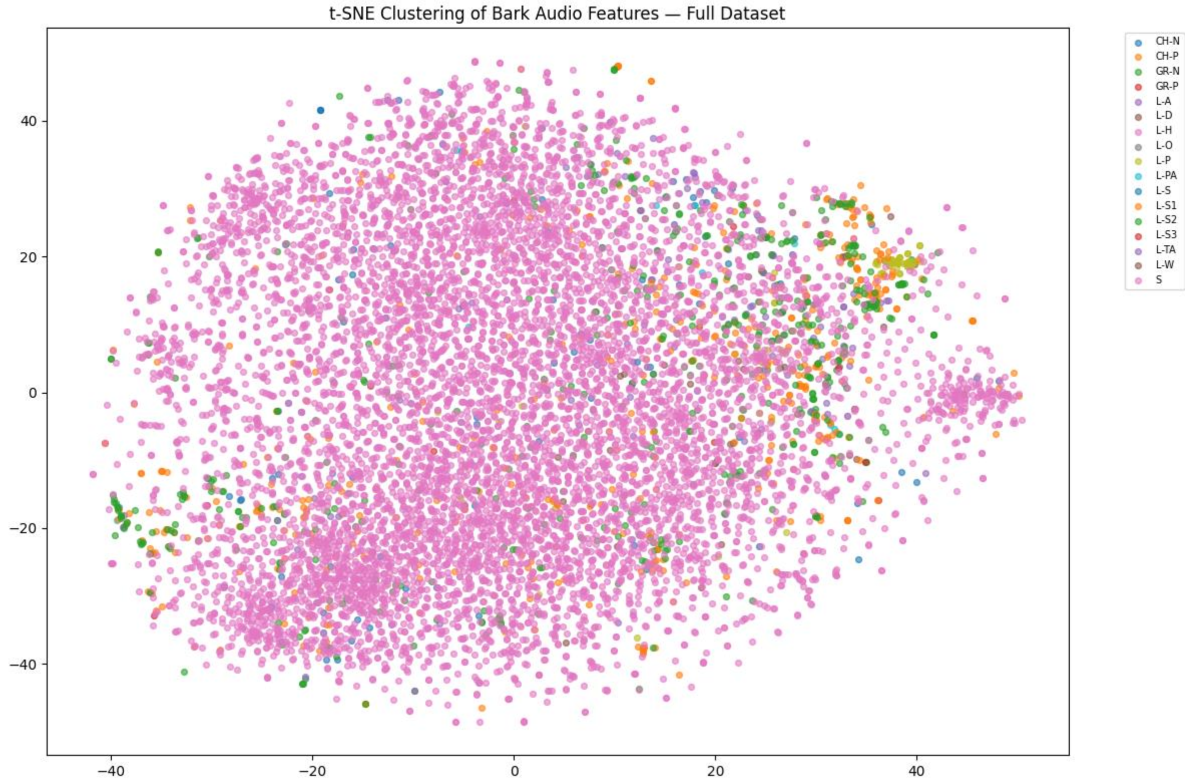


PCA WITH (S) CLASS:

PCA WITHOUT (S) CLASS:



TSNE WITH (S) CLASS:



TSNE WITHOUT(S) CLASS:

10. Analysis & Discussion

- Silence is highly learnable and dominates accuracy.
- Minority emotional classes are challenging due to acoustic similarity.
- t-SNE clusters show partial grouping for aggression, playfulness, and warning tones.
- Model performance suggests feasibility but requires class balancing or augmentation.

11. Application & Demo

A real-time audio application was built using **Streamlit** and **TensorFlow**, enabling:

- Microphone recording
- Feature extraction on-device
- Model inference
- Text translation output

Example output:

🎙 **Dog sound detected:**
Emotion: L-P (Playful)
“Yay! Let’s play!”

12. Conclusion

This work demonstrates a prototype system capable of interpreting dog vocalizations using machine learning. While results show promise, further refinement, larger balanced datasets, and multimodal cues (e.g., body posture) could significantly improve performance.

13. Future Work

Improvement	Benefit
• Data augmentation	Improve learning of rare classes

• CRNN or Transformer models	Better temporal modeling
• Multimodal dataset (audio + video)	More accurate emotional mapping
• Deployment as mobile app	
• Collecting more data of lesser data classes for better translation of those.	Real-world usage