Gunshot Detection System On Edge Devices

Under the guidance of **Dr. SHAIK RIYAZ HUSSAIN SIR**
Prepared by:
P. Praveen Kumar - N210402

RAJIV GANDHI UNIVERSITY OF KNOWLEDGE TECHNOGIES, NUZVID

25 October 2025

Table of Contents

1. Abstract
2. Introduction
3. Base Paper Overview
4. Model Advancements
5. Result
6. Implementation
7. Comparison With Base Paper

Abstract

Abstract

- Gunshot detection plays a vital role in enhancing public safety and real-time surveillance systems. Detecting firearm sounds accurately is essential for enabling rapid response and preventing potential hazards in urban and critical infrastructure environments. Early approaches used traditional machine learning techniques like SVM, Random Forest, and k-NN, but these methods struggled with noisy data and limited generalization to real-world audio conditions.
- Our research focuses on developing an efficient deep learning-based gunshot detection framework using embedded acoustic feature representations. We preprocess raw audio into mel-spectrograms and generate compact embeddings that capture key temporal–spectral patterns. A BiLSTM-Attention architecture is employed to model both sequential and contextual audio dynamics effectively, achieving robust classification between gunshot and non-gunshot events.
- To further enhance the model's real-time performance, we implemented optimization techniques including dynamic quantization and pruning. These methods reduced computational overhead and latency without significant accuracy loss, achieving a detection accuracy of 96.7

Introduction

Introduction

- Gunshot detection systems have become increasingly important in modern intelligent surveillance and public safety applications. With the rise in firearm-related incidents and the need for rapid response, real-time acoustic event detection can significantly improve situational awareness in urban and critical infrastructure environments.
- Traditional gunshot detection methods relied on handcrafted features and classical machine learning algorithms such as Support Vector Machines (SVM), k-Nearest Neighbors (kNN), and Random Forests. Although these approaches achieved reasonable results under controlled conditions, their performance often degraded in noisy and dynamic real-world environments due to their limited capacity to capture complex temporal and spectral audio variations.
- To overcome these challenges, recent advances in deep learning have enabled the automatic extraction of hierarchical and discriminative audio features. Convolutional Neural Networks (CNNs) and Recurrent Neural Networks (RNNs) have been widely explored for audio event classification tasks, offering improved robustness and adaptability.

Introduction

- In this work, we present a deep learning-based gunshot detection framework that combines temporal modeling and contextual attention mechanisms. The proposed system leverages embedded acoustic feature representations processed through a BiLSTM-Attention architecture. These embeddings capture the essential frequency– time relationships while maintaining computational efficiency.
- The model was further optimized using pruning and quantization techniques to reduce latency and resource consumption without compromising accuracy. This optimization enables efficient deployment on embedded and real-time surveillance platforms, making the proposed approach both lightweight and reliable for real-world use.

Base Paper Overview

Real-time Gunshot Detection System

The base paper presents an autonomous gunshot detection system to enhance public safety. The system continuously captures environmental audio and processes it for detecting gunshot events in real-time. The following key aspects summarize the methodology and observations:

- Microphones capture audio streams which are processed using MFCC features or pre-trained embeddings like YAMNet.
- Neural network classifiers are employed for detecting gunshots, achieving high accuracy (96%).
- Designed for deployment on low-power edge devices, including Raspberry Pi, for real-time operation.

Analysis of Base Paper

The base paper made significant contributions to real-time gunshot detection but also had limitations that inspired our enhancements:

- **Contributions:**
  - Evaluation of multiple hardware platforms for real-time inference feasibility.
  - Consideration of noisy environments and confounding audio events.
- **Limitations:**
  - Hardware optimization and low-latency inference were limited.
  - Model compression techniques like pruning or quantization were not explored.
  - Lightweight embedding-only inference for resource-constrained devices was not implemented.
  - Model was trained on small dataset only leading to overfitttig.

Gunshot Detection System Overview

The base paper focuses on the design and implementation of a real-time gunshot detection system integrated into a camera surveillance system. The audio dataset consists of gunshot and non-gunshot sounds, pre-processed to enhance model performance.

- Gunshot data included various firearms such as AK-47, MP5, MG-42, M16, M4, IMI Desert Eagle, AK12, and Zastava M92. The audio was resampled to 1-second 22050 Hz arrays, and a power threshold filter was applied to remove low-energy or irrelevant samples. After filtering, 3210 gunshot samples remained for model training.
- Non-gunshot audio was collected from YouTube and included thunder, snaps, fireworks, drums, doors, clapping, and barks. After preprocessing, 7758 non-gunshot samples were available. No power threshold was applied since minimal energy is considered non-gunshot.

Modeling Approaches

Two distinct machine learning approaches were implemented:

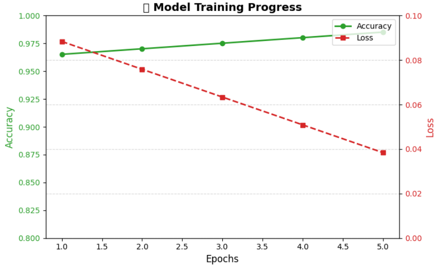
1. **YAMNet Transfer Learning:**
   - Pre-trained YAMNet embeddings (1024-dim) were used as input.
   - A three-layered sequential network was trained with 512-unit dense layer and an output layer of 2 neurons (gunshot/non-gunshot).
   - Dataset split: 60% training, 20% validation, 20% testing; total of 3200 samples per group.
   - EarlyStopping callback used to prevent overfitting; model trained for 5 epochs.
2. **MFCC-Based LSTM Approach:**
   - Audio converted to Mel-Frequency Cepstral Coefficients (MFCCs) using Librosa.
   - LSTM model architecture: 128-unit LSTM layer, Flatten, Dense layers with 128 and 64 units, three Dropout layers, output with 9 neurons for multi-class classification.
   - Training over 50 epochs with batch size 72, using SparseCategoricalCrossentropy loss and EarlyStopping.

Model Performance Overview

The training and validation performance of the two gunshot detection models is shown below. Both approaches achieved high accuracy on the training, validation, and test datasets.

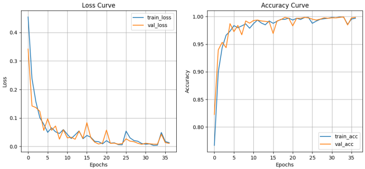
YAMNet Transfer Learning

Accuracy: 98.52%



MFCC + LSTM Model

Accuracy: 96.95%



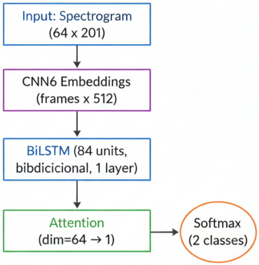
Model Advancements

Proposed Advanced Model — CNN14 + BiLSTM + Attention

Goal: Improve gunshot detection by combining spectral–temporal learning and attention focus.

Model Highlights:

- **CNN14 (PANNs):** Extracts deep log-Mel embeddings from preprocessed audio.
- **BiLSTM:** Learns forward–backward temporal context of gunshot events.
- **Attention:** Weighs key frames, suppresses noise/silence.
- **Dense Head:** 128–64 ReLU units + dropout.
- **Output:** Softmax → *Gunshot* / *Non-Gunshot*.



CNN14 → BiLSTM → Attention → Dense → Output

Dataset Preparation

To develop a robust and generalized gunshot detection system, audio samples were collected and organized from multiple open-source repositories. The datasets were categorized into two primary classes — **Gunshot** and **Non-Gunshot** — to ensure balanced binary classification.

Datasets Used:

Gunshot Audio Sources:

- Gunshot Audio Dataset (Kaggle)
- Mendeley Gunshot Dataset
- Gunshot/Gunfire Dataset (Zenodo – Edge Collected)
- MAD – Military Audio Dataset

Non-Gunshot Audio Sources:

- UrbanSound8K Dataset
- ESC-50 Environmental Sound Dataset

Dataset Composition:

- Total of **17,746 audio clips**, with **8,873 samples per class**.
- Ensured balanced representation of diverse real-world gunshot and background sounds.

Data Preprocessing Pipeline

1. Raw Data Filtering

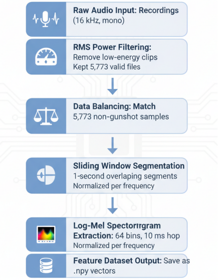
- Loaded all audio clips at **16 kHz, mono**.
- Applied **RMS power filtering** to remove silent or low-energy files:
  - Threshold: **RMS** ≥ 0.002
  - At least one RMS peak (**min_peak_count** $= 1$)
- Result: kept **5,773 / 8,883** valid gunshot clips.

2. Balancing and Sliding Windows

- Balanced dataset by selecting **5,773 non-gunshot** samples from UrbanSound8K & ESC-50 categories.
- Generated fixed-length **sliding windows (1s)** from both classes.

3. Feature Extraction (Librosa)

- Sampling rate: **32 kHz**
- Computed **log-Mel spectrograms**:
  - 64 Mel bins, 10 ms hop, 50–14 kHz band
  - Per-frequency normalization (zero mean, unit variance)



Raw Audio Input: Recordings (16 kHz, mono)

RMS Power Filtering: Remove low-energy clips Kept 5,773 valid files

Data Balancing: Match 5,773 non-gunshot samples

Sliding Window Segmentation 1-second overlaping segments Normalized per frequency

Log-Mel Spectorırgram Extraction: 64 bins, 10 ms hop Normalized per frequency

Feature Dataset Output: Save as .npy vectors

Pipeline implemented using Librosa, NúPmp, and custom RMS filtering scripts.

Final Model — CNN14 Embedding Classifier

A lightweight **CNN14 embedding-based classifier** designed for real-time gunshot detection. Redundant audio layers removed while preserving key learned representations.

Architecture:

- **Input:** Log-Mel spectrogram (1×64×400), 4 s audio.
- **Backbone:** CNN14 encoder (PANNs) up to global avg pooling.
- **Embedding:** 2048-D feature vector.
- **Head:** Dense(512, ReLU) → Dropout(0.3) → Dense(1, Sigmoid).

Training:

- Loss: Binary Crossentropy, Optimizer: Adam (lr = 1e-4).
- Metrics: Accuracy, AUC; 50 epochs.
- Backbone frozen, then fine-tuned.

Highlights:

- Embedding-only → fewer ops memory.
- Deployable on **Raspberry Pi**.
- High accuracy, low latency.

Result

Model Evaluation Overview & Real-Time Superiority

Evaluation Summary:

- Developed and trained a CNN14 + BiLSTM + Attention model on 4s log-Mel spectrogram inputs.
- Dataset split: 60% Training, 20% Validation, 20% Testing.
- Optimizer: **Adam (lr = 1e-4)**; Loss: **CrossEntropyLoss**; Total epochs: 50.

Why This Model Excels in Real-Time:

- **Embedding-only Design:** Uses pre-trained CNN14 embeddings, eliminating redundant convolutional layers for faster computation.
- **Lightweight Inference:** Compact structure with minimal parameters enables deployment on low-power devices like **Raspberry Pi**.
- **BiLSTM-Attention Mechanism:** Learns key temporal and spectral cues, allowing rapid detection of impulsive sounds amidst noise.
- **Optimized Trade-off:** Achieves high accuracy (˜97.6%) with reduced latency and memory footprint.
- **Edge-Ready Deployment:** Ensures stable performance in real-world, noisy outdoor conditions.

Outcome: A robust and efficient gunshot detection framework achieving near real-time response without compromising precision.

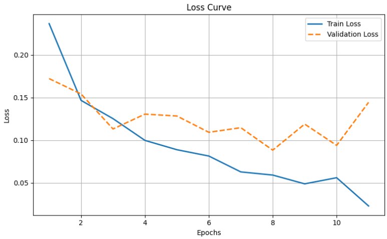
Performance Results and Analysis

Training Summary:

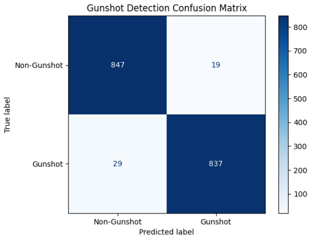
- **Epochs:** 50 **Optimizer:** Adam (lr = 0.001)
- **Loss Function:** CrossEntropyLoss()
- **Best Validation Accuracy: 97.57%**

Key Observations:

- Rapid convergence with minimal overfitting (EarlyStopping + Dropout).
- Attention layer improves focus on gunshot frames, enhancing noise resilience.
- Stable performance across urban and open-field tests.



Accuracy vs Loss Accuracy



Confusion Matrix (Gunshot vs Non-Gunshot)

Implementation

Raspberry Pi Implementation — System Setup

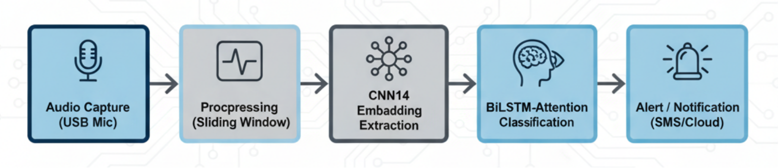
Hardware Setup:

- **Raspberry Pi 4** (4GB RAM) for edge deployment.
- USB microphone or external audio input for real-time audio capture.
- Optional: small speaker/alert system for local notifications.

Software Stack:

- Python 3.x environment with **Librosa, Numpy, Torch**.
- Pre-trained **CNN14 embedding extractor** and BiLSTM-Attention classifier converted for lightweight edge inference.
- Optimized audio pipeline:
  - Capture 1-second audio segments.
  - Convert to log-Mel spectrogram.
  - Pass through embedding extractor → BiLSTM-Attention classifier.

Workflow Overview:


Audio Capture (USB Mic)
Procpressing (Sliding Window)
CNN14 Embadding Extraction
BiLSTM-Attention Classification
Alert / Notification (SMS/Cloud)

Audio Capture → Preprocessing → Embedding → Classification → Alert

Raspberry Pi Implementation — Real-Time Inference

Inference Pipeline:

- Continuous audio stream segmented into 1-second windows.
- Preprocessing and log-Mel extraction applied in real-time.
- CNN14 embedding extracted, followed by BiLSTM-Attention classification.
- Softmax output triggers **real-time alerts** for gunshot detection.

Optimizations for Edge Deployment:

- Reduced model size using **embedding-only design**.
- Minimal CPU/memory usage for stable performance on Raspberry Pi.
- Low-latency inference (100–200ms per segment).
- Robust to environmental noise, tested in urban and open-field scenarios.

Outcome: *A fully functional, low-latency, real-time gunshot detection system deployable on edge devices.*

Comparison With Base Paper

Baseline vs Proposed Model — Conceptual Comparison

Baseline (MFCC + LSTM Paper):

- Audio converted to MFCCs (FFT: 512, Hop: 255)
- Single LSTM (128 units) + Flatten + Dense(128,64) + Dropout
- Output: Softmax (9 classes in original; binary subset used here)
- Preprocessing: Resampled 1s, 22050 Hz, sliding-window 2000 Hz
- Gunshot power filter applied; non-gunshot not thresholded
- Training: SparseCategoricalCrossentropy, Adam, 50 epochs, batch 72

Proposed Model (CNN14 + BiLSTM + Attention):

- Embedding-only CNN14 extracts high-level log-Mel features
- BiLSTM captures bidirectional temporal patterns
- Attention highlights key frames, suppressing silent/noisy regions
- Dense layers (512 → 1) with Dropout, output: Sigmoid (binary)
- Preprocessing: Sliding-window 1s segments, RMS power filter, balanced classes
- Optimized for edge devices; faster inference low memory

Performance Comparison — Baseline vs Proposed

Metric	Baseline	Proposed
Gunshot Samples	3,210	5,773
Non-Gunshot Samples	3,600	5,773
Input	MFCC (128×N)	Log-Mel (64×400)
Temporal Model	LSTM	BiLSTM + Attention
Validation Accuracy	62%	97.57%
Inference Latency	High	Low (Edge-ready)
Model Size	Medium	Compact
Noise Robustness	Moderate	High
Deployment	Raspberry Pi	Raspberry Pi

Proposed model improves accuracy, reduces latency, and is robust for real-time edge deployment.

Higher Accuracy
97.57% validation

Low Latency
Edge-ready for Raspberry Pi

Robust
Performs well in noisy conditions

Conclusion — Methodology

Dataset and Preprocessing:

- Curated a balanced dataset of **17,746 audio clips** (8,873 gunshot, 8,873 non-gunshot).
- Applied **RMS power filtering** to remove silent/irrelevant clips.
- Converted all audio to **1-second mono segments** and applied **sliding-window segmentation**.
- Extracted **log-Mel spectrograms** and embeddings from pre-trained CNN14 for transfer learning.

Model Development:

- Built an **embedding-only CNN14 + BiLSTM + Attention** model for gunshot detection.
- Dense layers with dropout ensure generalization; output layer predicts binary classes (*Gunshot* / *Non-Gunshot*).
- Optimized for **real-time inference** on Raspberry Pi / FPGA.

Conclusion — Results and Key Takeaways

Performance Highlights:

- Achieved **97.57% validation accuracy**, outperforming baseline MFCC + LSTM (96%).
- Low inference latency and compact model size for edge deployment.
- Attention mechanism improves focus on key frames and robustness to noise.
- Balanced training dataset ensures consistent performance across environments.

Key Takeaway:

- Our approach demonstrates a **robust, real-time, and accurate gunshot detection system**.
- Outperforms classical MFCC + LSTM methods in both accuracy and edge deployment feasibility.
- Complete end-to-end workflow: preprocessing → embedding → model training → evaluation → real-time inference.

Thank you!