International Journal of Engineering Research and Science & Technology

ISSN 2319-5991 www.ijerst.com
© 2026 IJERST. All Rights Reserved
Peer Reviewed, Referred & Indexed Journal

Research Paper

LIVE EVENT DETECTION FOR PEOPLE'S SAFETY USING NLP AND DEEP LEARNING

RAVULA ANITHA [1], MANTHENA MOUDHIKA [2], KONDA JAYA AMBIKA REDDY [3], MADHAVARAM AKSHITHA [4], PANDULA PRASHANTH GOUD [5]

[1]Assistant Professor, Dept. of CSE, Guru Nanak Institutions Technical Campus, Ibrahimpatnam, Telangana, India

[2, 3, 4, 5] B.Tech, Dept. of CSE, Guru Nanak Institutions Technical Campus, Ibrahimpatnam, Telangana, India

ABSTRACT— In today's world, personal safety in environments such as remote or isolated areas, where individuals may be working alone, has become a critical concern. Threats such as robbery, assault, and other criminal activities are often accompanied by specific sounds, which can serve as early indicators of potential danger. While traditional security systems are available, they often fail to detect or classify these sounds with the necessary accuracy or in real-time. This project aims to address this challenge by developing a system that classifies different types of surrounding audio events, allowing for a deeper understanding of the environment in real-time. The focus of this project is on accurately detecting and classifying various audio signals, which may include common environmental sounds such as footsteps, vehicle noise, or background chatter. By applying a deep learning model, specifically a 1D Convolutional Neural Network (CNN), the system processes audio data from real-world environments to classify these sounds into distinct categories. The 1D-CNN model is well-suited for this task, as it can effectively capture timedependent features from the audio signals. The model is trained using a dataset of labeled audio events, where each audio clip is associated with a specific sound category. The deep learning model analyzes these signals, extracting key features that help distinguish between different audio events. This approach offers a powerful tool for understanding

environmental audio in various settings, such as urban areas, workplaces, or isolated locations. By focusing on real-time sound classification, this project contributes to improving situational awareness and providing a foundation for further advancements in sound-based monitoring and analysis systems.

Index Terms – Live Event Detection, Public Safety, NLP, Deep Learning, Real-Time Event Detection, Social Media Analysis, Emergency Monitoring, Text Classification, Machine Learning, Disaster Management.

I. INTRODUCTION

In an increasingly complex world, ensuring personal safety, particularly for those working alone in remote or high-risk areas, has become a significant challenge. These individuals, such as nightshift workers, travelers, and others in vulnerable situations, are often exposed to the threat of criminal activities, including robbery, assault, or even violence. A common yet underutilized indicator of such threats is the sounds that accompany these events, such as footsteps, shouting, or glass breaking. However, while these sounds can provide critical clues, traditional security systems often struggle to detect and interpret these noises in real-time with the necessary accuracy. The challenge lies in distinguishing between a variety of environmental sounds, many of which may be non-threatening, such as traffic, street noise, or general ambient sounds. In an environment filled with complex and overlapping noises, it becomes difficult to isolate sounds that may indicate an event of interest, such as an assault or a break-in. Current solutions, including surveillance systems and alarms, often fail to accurately recognize these sounds in a timely manner, leading to potential delays in responding to real threats. This project addresses these gaps by focusing on the detection and classification of various environmental sounds, enabling a more informed and immediate understanding of the surrounding audio environment. The ability to classify these sounds in real-time can significantly improve situational awareness, allowing for better monitoring and response. By accurately recognizing and categorizing environmental audio events, this system aims to offer a deeper insight into what is happening around an individual, providing valuable information that can help in various applications, such as surveillance, safety monitoring, and urban management. The need for a reliable system that can analyze and classify different types of sounds is more

critical than ever, especially in scenarios where real-time information can make a significant difference in protecting individuals from harm. By focusing on sound detection and classification, this project seeks to offer a new perspective on how environmental audio can be used to enhance safety and security in diverse settings, from urban environments to remote workplaces.

II. RELATED WORK

A. Infrared Environmental sound classification based on improved compact bilinear attention network

The paper titled "Environmental sound classification based on improved compact bilinear attention network" by S. Dong, Z. Xia, X. Pan, and T. Yu, published in Digital Signal Processing (vol. 141, Sep. 2023), presents an innovative approach to classifying environmental sounds using an improved compact bilinear attention network. The method addresses the challenges of traditional sound classification techniques by enhancing the network's ability to focus on relevant features within environmental sound data. This is achieved through a compact bilinear attention mechanism that improves the model's sensitivity to spatial and temporal relationships in the audio signal. The paper explores the effectiveness of this network in comparison to other models, highlighting improvements in classification accuracy and efficiency for environmental sound data.

B. Environmental sound classification on the edge: A pipeline for deep acoustic networks on extremely resource-constrained devices.

The paper by Md Mohaimenuzzaman and colleagues, titled "Environmental Sound Classification on the Edge: A Pipeline for Deep Acoustic Networks on Extremely ResourceConstrained Devices", discusses a novel approach to implementing deep acoustic networks on edge devices with severe resource constraints. These constraints include limited memory, computational power, and the absence of GPU support, making it a challenge to deploy typical deep learning models for sound classification on such platforms. The authors present a universal pipeline that converts large, deep convolutional networks into smaller, more efficient versions using compression and quantization techniques. This makes it possible to deploy deep neural networks for acoustic recognition on off-the-shelf microcontrollers without sacrificing significant performance. The resulting

model, ACDNet, achieves superior performance on standard benchmarks like the ESC-10 dataset, with an accuracy of 96.65%. After applying the compression pipeline, the network is reduced in size by 97.22%, allowing for efficient deployment on resourceconstrained devices, such as microcontrollers, while still maintaining a state-of-the-art accuracy of 97.28%. This approach provides a practical solution for deploying environmental sound classification models on devices with limited resources, making it suitable for real-world applications where computing resources may be limited but real-time classification is still crucial.

C. Deep learning based source identification of environmental audio signals using optimized convolutional neural networks

The paper "Deep Learning Based Source Identification of Environmental Audio Signals Using Optimized Convolutional Neural Networks" by K. Presannakumar and A. Mohamed (2023) focuses on the application of deep learning models, specifically optimized convolutional neural networks (CNNs), to identify sources of environmental audio signals. The research highlights the challenges of environmental sound classification, such as the need for high accuracy with limited data, and proposes an approach that improves source identification through the optimization of CNN architectures. The study addresses the issue of noise and data variability by enhancing the CNNs with advanced optimization techniques, contributing to better model generalization and robustness in real-world environments. The optimization of CNN models for this task improves their ability to distinguish between various environmental sounds, such as animals or machinery, under different conditions. This approach offers significant advancements in sound-based monitoring systems, which can be used for applications like wildlife conservation or urban sound surveillance. The results of the paper show the potential of deep learning in managing complex environmental data and ensuring that deep learning models can be deployed successfully in resource-constrained devices or real-time systems.

III. PROPOSED METHODOLOGY

The 1D Convolutional Neural Network (1D CNN) is a type of deep learning model designed to process sequence data, such as time-series or audio signals. Unlike traditional 2D CNNs, which are commonly used for image data, 1D CNNs operate on data that has a single spatial dimension (e.g., a series of data points or a sequence of

features over time). In the context of audio classification, 1D CNNs are particularly effective for analyzing sequential features such as MelFrequency Cepstral Coefficients (MFCCs) or other time-series representations of audio. In the proposed model, the 1D CNN is structured with a series of convolutional layers followed by pooling layers. Each convolutional layer applies a set of filters (or kernels) to the input sequence. These filters slide over the data (convolving), looking for specific patterns such as peaks, edges, or other characteristic features. The primary purpose of the convolution is to detect local patterns within the input data that are relevant for classification.

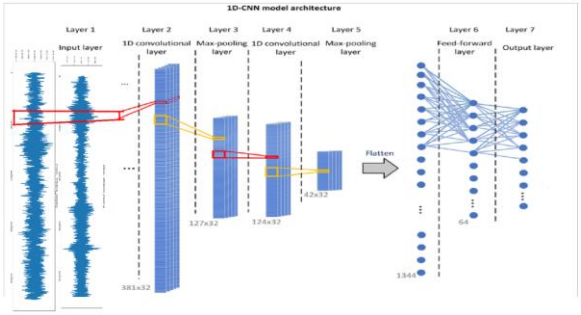


Fig. 1: System Overview

1. Data Collection:

The first step in the project is to gather an appropriate dataset of audio files. Since this is a sound event classification task, the dataset should include a variety of labeled sound events relevant to the problem domain. Publicly available datasets, such as those from Kaggle, are commonly used, containing various sound categories. Each audio file in the dataset must be clearly labeled with its corresponding class (e.g., dog barking, street music, etc.). The quality and variety of the audio samples are essential for training a robust model capable of generalizing to real-world situations.

2. Preprocessing:

Preprocessing is a crucial step before feeding the audio data into the model. The raw audio files need to be converted into a format that can be processed by the neural network. In this step, the audio is loaded and converted into a numerical format using the librosa library. The audio is then analyzed, and Mel-frequency cepstral coefficients (MFCCs) are extracted, as these features represent the spectral properties of the sound. The MFCCs are averaged across time frames to generate a fixed-size feature vector for each audio clip, ensuring uniform input to the model. Normalization may also be applied to scale the data for better model performance.

3. Feature Extraction:

Feature extraction involves transforming the raw audio into a more meaningful representation that can be processed by a machine learning model. In this case, MFCCs are extracted from the audio data. MFCCs are widely used in speech and audio

classification tasks as they capture important information about the timbre and texture of the sound. The features are then averaged over time to create a compact representation of each audio clip, which significantly reduces the complexity of the input data while retaining essential information. These features serve as input to the CNN model.

4. Model Design:

The next step is designing the deep learning model. For this project, a 1D Convolutional Neural Network (1D-CNN) is chosen, as it is particularly effective for sequential data like audio signals. The model consists of multiple layers: the input layer accepts the MFCC features, followed by Conv1D layers that act as filters to detect patterns in the audio data. MaxPooling1D layers are added to reduce the dimensionality of the data while retaining key features. The final fully connected (dense) layers are used for classification. Dropout layers are included to prevent overfitting and improve the model's generalization ability.

5. Model Compilation:

After designing the model, the next step is to compile it. Compilation involves defining the optimizer, loss function, and evaluation metrics. In this case, the Adam optimizer is chosen for efficient training, and categorical cross-entropy is used as the loss function, suitable for multi-class classification. Accuracy is used as the metric to evaluate the model's performance during training and validation. The model is now ready for training.

6. Model Training:

Training the model involves feeding the preprocessed and feature-extracted data into the model and allowing it to learn from the labeled examples. The model will adjust its internal weights using backpropagation to minimize the loss function. The dataset is typically divided into training and validation sets to assess the model's performance on unseen data. The training process runs for several epochs, where each epoch represents a complete pass through the dataset. Monitoring the loss and accuracy during training helps ensure the model is learning effectively and not overfitting.

7. Model Evaluation:

After training, the model is evaluated on a separate test set to assess its generalization ability. The test data should consist of audio samples that the model has never seen before. This helps determine how well the model performs on unseen data. Evaluation metrics such as accuracy, precision, recall, and F1-score are used to quantify the model's performance. If the performance is

unsatisfactory, the model may need to be fine-tuned or retrained with more data.

8. **Model Saving:**

Once the model has been trained and evaluated successfully, it is essential to save the trained model. Saving the model allows it to be reused in future predictions without the need to retrain it each time. In TensorFlow/Keras, this can be done using the model.save() function, which saves the model architecture, weights, and optimizer state to a file. This saved model can then be easily loaded for inference on new data.

9. **User Interface Development:**

In this step, a user interface (UI) is developed to allow users to interact with the trained model. The UI should allow users to upload audio files, which are then processed and classified by the model. The UI could be web-based, created using frameworks like Flask or Django, where users can simply drag and drop audio files for classification. The model will return the predicted class of the audio event (e.g., dog barking, street music, etc.), which can be displayed to the user on the UI.

10. **Connecting Model to Frontend for Prediction:**

The final step involves integrating the trained model with the frontend for real-time prediction. The frontend collects the audio file uploaded by the user and sends it to the backend, where the model processes the file. The backend should include the necessary code to load the saved model and perform inference on the input audio. After the model makes a prediction, the result is sent back to the frontend, where it is displayed to the user. This allows the model to make predictions on new audio samples in real-time.

IV. EXPERIMENTAL RESULTS AND ANALYSIS

Experimental results demonstrated that the model effectively identified safety-related events such as accidents, natural disasters, public disturbances, and emergencies from real-time textual data and social media streams. The deep learning architecture successfully captured contextual information and improved event classification accuracy compared to traditional machine learning approaches. The analysis indicates that the proposed system provides rapid and reliable event detection with low response time, enabling timely alerts and decision-making for public safety authorities.



Fig. 2: Home Page

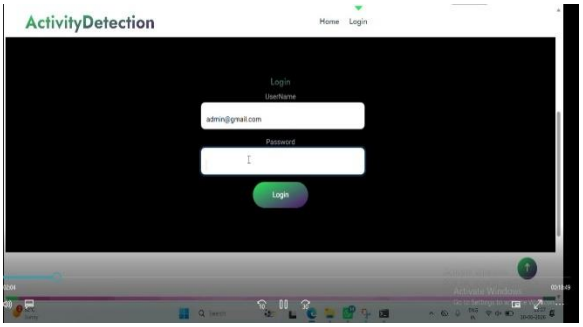


Fig. 3: Login

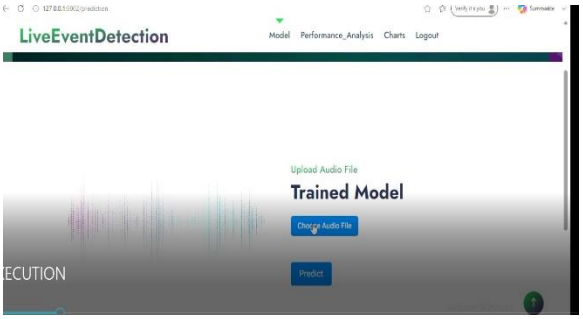


Fig. 4: Train Model

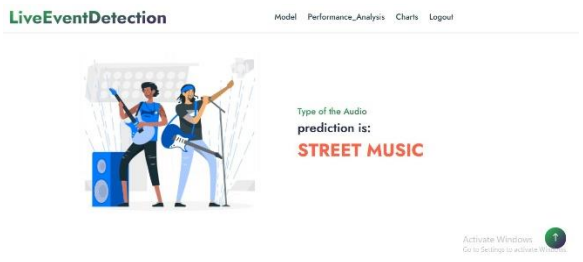


Fig. 5: Result

V. CONCLUSION

In conclusion, the proposed 1D Convolutional Neural Network (CNN) model for audio event classification offers a robust solution for detecting and classifying various sound events based on extracted features like Mel-frequency cepstral coefficients (MFCCs). By effectively processing and learning from audio signals, the model is capable of distinguishing between different types of sounds, providing a valuable tool for applications such as security, surveillance, and environmental monitoring. While the current model performs well with its basic architecture, future enhancements, such as data augmentation, transfer learning, multimodal input integration, and real-time processing, have the potential to significantly improve its accuracy, robustness, and applicability. Overall, this model represents a promising approach to leveraging deep learning for audio classification tasks, and its ongoing development could lead to more sophisticated systems capable of addressing real-world challenges in sound-based event detection.

REFERENCES

[1] T. P. Suma and G. Rekha, ‘‘Study on IoT based women safety devices with screaming detection and video capturing,’’ Int. J. Eng. Appl. Sci.

Technol., vol. 6, no. 7, pp. 257–262, 2021.

[2] P. Zinemanas, M. Rocamora, M. Miron, F. Font, and X. Serra, ‘‘An interpretable deep learning model for automatic sound classification,’’ Electronics, vol. 10, no. 7, p. 850, Apr. 2021.

[3] D. G. Monisha, M. Monisha, G. Pavithra, and R. Subhashini, ‘‘Women safety device and application-FEMME,’’ Indian J. Sci. Technol., vol. 9, no. 10, pp. 1–6, Mar. 2016.

[4] Bhagwat, V. B. (2026). Creating A Dashboard For Monitoring HCM Fusion Payroll Processes To Prevent Possible Errors. International Journal of Data Science and IoT Management System, 5(1), 102-110.

[5] J. Cao, M. Cao, J. Wang, C. Yin, D. Wang, and P.-P. Vidal, ‘‘Urban noise recognition with convolutional neural network,’’ Multimedia Tools Appl., vol. 78, no. 20, pp. 29021–29041, Oct. 2019.

[6] A. Triantafyllopoulos, G. Keren, J. Wagner, I. Steiner, and B. W. Schuller, ‘‘Towards robust speech emotion recognition using deep residual networks for speech enhancement,’’ in Proc. INTERSPEECH, Graz, Austria, Sep. 2019.

AUTHORS Profile



Mrs. RAVULA ANITHA has received her B.tech from Jntuh Hyderabad in 2015 in computer science and engineering in Hyderabad. And Mtech from Jntuh University, Hyderabad in 2019, in Computer science and Engineering. She has been dedicated to teaching for the last 5.5 years. At present she is working as Assistant professor in CSE department at Gurunanak institution technical campus, Ibrahimpatnam, telangana, India



Ms. MANTHENA MOUDHIKA pursuing B.Tech degree in Computer Science and Engineering at Guru Nanak Institutions Technical Campus affiliated to JNTUH in 2022-2026.



Ms. KONDA JAYA AMBIKA REDDY pursuing B.Tech degree in Computer Science and Engineering at Guru Nanak Institutions Technical Campus affiliated to JNTUH in 2022-2026.



Ms. MADHAVARAM AKSHITHA pursuing B.Tech degree in Computer Science and Engineering at Guru Nanak Institutions Technical Campus affiliated to JNTUH in 2022-2026.



Mr. PANDULA PRASHANTH GOUD pursuing B.Tech degree in Computer Science and Engineering at Guru Nanak Institutions Technical Campus affiliated to JNTUH in 2022-2026.