Research

Comparative analysis of audio classification with MFCC and STFT features using machine learning techniques

Mahendra Kumar Gourisaria[1] · Rakshit Agrawal[1] · Manoj Sahni[2] · Pradeep Kumar Singh[3]

Received: 28 April 2023 / Accepted: 8 November 2023
Published online: 03 January 2024

Abstract
In the era of automated and digitalized information, advanced computer applications deal with a major part of the data that comprises audio-related information. Advancements in technology have ushered in a new era where cutting-edge devices can deliver comprehensive insights into audio content, leveraging sophisticated algorithms such such as Mel Frequency Cepstral Coefficients (MFCCs) and Short-Time Fourier Transform (STFT) to extract and provide pertinent information. Our study helps in not only efficient audio file management and audio file retrievals but also plays a vital role in security, the robotics industry, and investigations. Beyond its industrial applications, our model exhibits remarkable versatility in the corporate sector, particularly in tasks like siren sound detection and more. Embracing this capability holds the promise of catalyzing the development of advanced automated systems, paving the way for increased efficiency and safety across various corporate domains. The primary aim of our experiment is to focus on creating highly efficient audio classification models that can be seamlessly automated and deployed within the industrial sector, addressing critical needs for enhanced productivity and performance. Despite the dynamic nature of environmental sounds and the presence of noises, our presented audio classification model comes out to be efficient and accurate. The novelty of our research work reclines to compare two different audio datasets having similar characteristics and revolves around classifying the audio signals into several categories using various machine learning techniques and extracting MFCCs and STFTs features from the audio signals. We have also tested the results after and before the noise removal for analyzing the effect of the noise on the results including the precision, recall, specificity, and F1-score. Our experiment shows that the ANN model outperforms the other six audio models with the accuracy of 91.41% and 91.27% on respective datasets.

Keywords Artificial Neural Network · Audio Classification · Audio file management · Audio visualization · Automated Systems · Mel Frequency Cepstral Coefficients · Short-Time Fourier Transform

1 Introduction

Many multimedia, digital, and advanced computerized types of machinery like Audio assistance, Automated Customer Support Sytems, and many more have audio data as one of an integral part for storing various information including environmental sounds, noises, Foley, speech sounds, nonspeech utterances, etc., and even stores more information than video signals [1]. Classifying environmental sounds stands apart from the classification of speech and other

✉ Pradeep Kumar Singh, pradeep.cse@cujammu.ac.in; Mahendra Kumar Gourisaria, mkgourisaria2010@gmail.com; Rakshit Agrawal, rakmak3456@gmail.com; Manoj Sahni, manojsahani117@gmail.com | [1]School of Computer Engineering, KIIT Deemed to Be University, Bhubaneswar, Odisha 751024, India. [2]Department of Mathematics, Pandit Deendayal Energy University, Gandhinagar, Gujarat 382426, India. [3]Central University of Jammu, Bagla Suchani, Jammu & Kashmir, India.

music files due to the paucity of prior knowledge regarding their temporal and frequency characteristics. Unlike more well-structured domains like speech and music, environmental sounds encompass a wide and diverse array of audio sources, necessitating classification models to exhibit an increased level of adaptability and generalization to effectively discern and categorize these often complex and heterogeneous auditory inputs. Moreover, environmental sounds are very random and not having any set fashion to work on, many naïve prediction algorithms (algorithms that are not well hyper-tunned for the desired audio frequencies) tend to fail in obtaining fruitful results which makes the environmental sound classification a challenging task for the researchers. In the present scenario, the main focus of the researchers in the field of auditory is to accurately recognize the speech or music files. However, analysis of sounds in the environment being an immensely mixed group of day to day audios which are unlikely to be categorized as speech or music has left behind in the upcoming improvements despite having various applications available in IoT technologies [2], hearing aids [3], smart room monitoring [4], video content highlight generation [5–7], audio surveillance systems [8]. Over the years, with the advancement in digital technology and broadcast facilities, users are now enabled to make use of the huge amount of multimedia and audio files. Since the environmental sounds contain various types of noises, textured and structural components namely scattering and iterations, it becomes a much harder task to classify such audio signals accurately than the classification of speech and music which becomes one of the challenging parts of our research to acquire the accurate results.

Classification and categorization of audio signals accurately become an important research area [9, 10]. Such classifications and sampling of audio signals come under the pattern recognition field. Over the years, various machine learning models are developed for accurate predictions, classifications in various fields of the industry, medical, etc. [11]. The main obstacles that come in the way of finding accurate results are namely feature selection and categorizing the audio signals based on the extracted features from that audio signal. To overcome such barriers, researchers generally go under various preprocessing of the audio files which include the process of noise removal, feature selection, and feature extraction. Some of the feature selection methods are Linear Predictive Coefficient (LPCs) [12], Linear Predictive Cepstral Coefficient (LPCCs) [13], Short Time Fourier Transform (STFT) [14], Mel Frequency Cepstral Coefficient (MFCCs) [15–17]. About the issues mentioned above, the major contribution of our research work presented in the paper comprises of (1) Choosing the dataset having environmental sounds with a suitable number of sampling records, (2) choosing an appropriate feature selection technique for extracting the features for classification of audio files, and (3) removal or noise and trimming the main and effective part of the audio signals without missing any important features. We use MFCCs and STFTs features being widely used in automatic speech recognition for the classification of audio files in our experimental models which is discussed in detail in Sect. 3. The experiment also includes the classification of audio files from two different datasets having different numbers of samples with variations in the noise level. the rest of the paper is organized in the following sections: 2. *Motivation* and *Contribution,* 3. *Related work*, 4. *Dataset and Methodology*, 5. *Classification Models*, 6. *Experimentation*, 7. *Results*, 8. *Discussion* and 9. *Conclusion* and 10. *Future Work*. For detailed analysis, we have calculated all the parameters namely precision, recall, specificity, and F1-score.

2 Motivation and contribution

In this section, we have presented the reasons which motivated us to conduct our research in the field of audio classification. We have also highlighted our key contributions that have been addressed throughout the research.

Our motivation centers around efficient file management and the recognition of audio files, which can significantly reduce human labor. Our goal is to present models that can seamlessly integrate with AI systems and take actions based on the classified type of audio file. Using Neural Network techniques in our models, we achieved a correct classification rate of 91 out of 100 instances. Our model holds great potential for applications in the development of smart roads, hospitals, and industries.

The main motivation behind our research was to offer a comprehensive comparative study of various models that can be employed based on the results obtained in different workspaces. Our experiments yielded acceptable results, which can be valuable when selecting a model. Furthermore, our comparative analysis lends support to the results obtained in these experiments. Unlike previous studies, which often presented only one or two models for classification and yielded less accurate results than our model, we aim to provide a broader selection of models and apply the same experimental approach to each one, offering a clear understanding of each model's performance in the field of audio classification. Additionally, our work utilizes a dual dataset approach for more comprehensive model classification, based on results, a feature not present in previous proposals. Both datasets have been chosen to maintain consistent sound quality but with varied features, and each dataset undergoes a separate feature extraction process.

3 Related work

In the past few years, many researchers have proposed their algorithms and techniques for classifying the audio signals using various parameters into consideration [18–22]. Many of the researches followed the two basic preprocessing steps namely analysis of the incoming audio signal and the second step was to extract the key features from the incoming audio file. This step of extracting the features reduced the unwanted data to a large scale and the classification is performed based on these extracted features. The feature selection can further broadly be classified into two categories: waveforms [23, 24], and spectrogram [25–27]. The waveform-based classification processes the input data as a 1-D array, while on the other hand, spectrogram-based classification converts the audio signal into spectrograms using time-dependent Fourier transformation (or STFTs). The study presented by Li et al. [28] (2001) discussed the audio classification based on LPC and MFCC features and results showed that cepstral-based features helped in classifying the audio signals accurately. Guo et al. [29] (2003) proposed a new metric namely Distance From Boundary (DFB) for the classification process. This process includes the searching of appropriate boundary which contains the audio file pattern and at last, these distances are sorted by their distances.

The study presented by Cowling M, et al. [30] (2003) experimented with stationary and non-stationary time–frequency-based features extraction process for classifying the environmental sounds. The use of audio surveillance is also considered for the detection and classification of various acoustic incidents such as humans coughing [31], impulsive sounds [32, 33] including gunshots, glass breaking, explosions, alarms, etc. Dargie, W et al. [34] (2009) proposed a study for audio sound classification using MFCC features, however, the performance rate was resulting high but the specific sound results including the accuracy of classifying the audio file remained lacking behind. Another study was given by El-Maleh, K et al. [35] (1999) illustrated many different pattern-based classification models namely QGC (Quadratic Gaussian Classifier), KNN (K-Nearest Neighbor) classifier, and LSLC (Least-Square Linear Classifier). The experiment also included the noise removal process and used LPC features extraction. The QGC classifier achieved the best results with an error rate of 13.6%. However, the study did not compare the results or implemented the models on more than one dataset, unlike our study, and even gained less error rate than the study given by [35].

Seker, H. et al. [36] (2020) proposed the study on classifying the environmental sounds using CNN based model and achieved an accuracy of 82.26% while we achieved an accuracy of 91.41% and 91.27% of accuracies in two different datasets in classifying the environmental sounds. Another study presented by Zhang Z. et al. [37] (2021) illustrated the classification of environmental sounds using ACRNN based model for achieving an accuracy of 86.1% which is still becoming less when compared to our model using the ANN-based architecture. Another contrast is the number of layers used, [36] used 10 layers in ACRNN structure while we have used 2 and 4 layers in two different datasets respectively. About [36], we can state that even simpler ANN architectures can achieve state-of-the-art accuracy instead of having bigger architectures which indirectly lowers the training time and lower computational expenses. The study shown by P. Dhanalakshmi et al. [38] (2009) focused on classifying the audio signals into six categories using different features like LPC, MFCC, LPCC while in contrast to the research presented by us, have classified the audio signals of two different datasets in 10 and 8 categories respectively with better accuracies and other parameters. Although we got the best result in the ANN classifier model, they scored the best accuracies using SVM and the Radial Basis Function Neural Network (RBFNN). Chen Lie et al. [39] (2006) proposed the study on classifying the environmental sounds using various classifiers and shown that SVM achieved better results i.e. 91.41% with a loss reduction of 8–15%, however, the results got reduced when the authors classified environmental sounds having three classes and results came down to 64.76% whereas, in our study, we achieved the better results (91.41% and 91.27% of accuracy in two datasets) using ANN classifier model with least losses.

Apart from these researches, recent studies have also considered deep convolutional neural networks (DNN) for classifying the audio samples. The study presented by Maccagno et al. [40] (2021) incorporated CNN based approach for audio classification at construction sites. The proposed DNN model used spectrograms that were created through the frequency scale and time derivatives. The frame size was considered to be 22,050 Hz and 60 mel bands. The dataset consists of 5 classes and used a fivefold cross-validation process and was able to achieve an accuracy of 97.08%. A similar study was presented by Mehyadin et al. [41] (2021) for bird sound classification. The authors used the model for analyzing the bird sounds and enable the species detection process. All the audio samples that contained noise, were treated with a separate noise filter which used the MFCC feature extraction process. The experiment included three models namely Naïve Bayes, J4.8, and Multilayer Perceptron (MLP) out of which J4.8 achieved the highest accuracy of 78.4%.

Another study presented by Palanisamy et al. [54] where author ckassified audio signals. Authors incorporated dural dataset approach including UrbanSound8k and ESC-50 dataset. In the experiment, ImageNet-Pretrained Standard CNN model was used and achieved a validation accuracy of 92.89% and 87.42% on respective datasets. Similar study was done by Zeghidour et al. [55] where authors proposed LEAF architecture (Learnable Frontend For Audio Classification). The experimental results showed that the proposed model outperformed EfficientNetB0 model. Study also states that the proposed architecture can be integrated with other neural networks at a low parameter cost. The proposed model is fully trainable, and lightweight architectire. The model will learn all the operations for extracting audio features starting from filtering to pooling steps. Based on this research [55], a new efficient, hybrid, and lightweight model can be developed which can provide high and accurate learning rate in helping the fire alarms and other audio signal detection methods.

From the above survey of various former proposals and experiments that have been presented in the classification of audio files, it can be seen that the authors have included only a few models for the categorization process. However, in our research, we have included a total of seven classification models that are used in two different audio datasets having different feature extraction steps.

4 Concepts and dataset

In the following section, we have presented a detailed overview of our two datasets used for classification models followed by the techniques and types of features selected for extracting the key parameters from the audio files for categorizing part. For the feature extraction process, we have used the Librosa library for extracting the features from the audio samples. The features include MFCCs and STFT. In the feature extraction process, a total of 186 features have been extracted for every label in the respective dataset. The length of each audio sample after the noise removal and feature extraction is taken to be 4 s. In the resulted audio samples, only the unique audio is taken which helps in classification. By the unique audio we mean, the audio which has a different frequency, pitch, etc. from the background noise, and other audio. Once the audio sample is noiseless, it is trimmed to the part where the actual identical sound can be distinguished. This can be well illustrated in Figs. 5, 6, and 7. The following is the bifurcation for this section: A. *MFCC features*, B. *STFT features*, C. *UrbanSound8K Audio Dataset*, and D. *Sound Event Classification Dataset*.

4.1 MFCC features

The key step for accurate classification is the extraction of discrete features from the audio sample or the components which can help in identifying the linguistic contents of the sample discarding the other stuff which carries noises and other unrequired sounds. For such feature extractions, MFCCs are used [42]-[43]. These *MFCCs* are one of the features that are widely used for extracting the features from the audio samples having less noise. The MFCCs are computed using fast Fourier transformation (FFT) coefficients filtered by a bandpass filter bank. The mathematical expression for Mel-scale computation is shown in Eq. (1).

$$Freq_{mel} = \frac{x * log((c + f)/x)}{log(2)} \tag{1}$$

In Eq. (1), $Freq_{mel}$ is the logarithmic scale of the normal frequency (f) scale, x plays an important role in calculating the MFCC features. This coefficient helps in converting the high-frequency sound into low frequencies for more accuracy pointing out the changes in the audio sample. It should have an appropriate range of 250 to 350 ie. the number of triangular filters that come in the frequency range of 200-1200 Hz which is the range of dominant audio information. For illustration, a full filter bank can be seen in Fig. 1 [56].

In the final step, MFCCs are calculated using Eq. (2) as illustrated and are denoted as $Feature_{MFCC}$.

$$F_{MFCC} = \sqrt{\frac{2}{N}} \sum_{k=1}^{N} (logS_k) cos\left[n(k - 0.5)\frac{\pi}{N}\right] \tag{2}$$

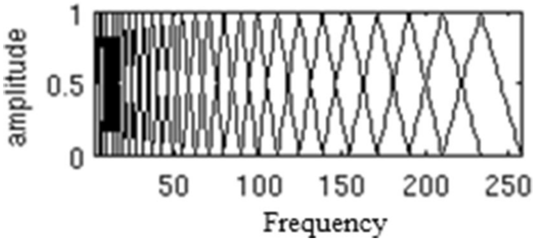


Fig. 1 Basic full filter bank [55]

where, S_k is the output of the filter bank where k varies from 1 to N, where N is the length of the DFT.

4.2 STFT features

The time-dependent signals are decomposed using Fourier Transform into their respective frequencies. One of the Fourier Transform includes Short-Time Fourier Transform (STFT) which is widely used for extracting the features from the audio sample. Although MFCCs are also widely used in the feature extraction process, they are also very sensitive to the background noises in the sample making them less effective in extracting the features while on the other hand, STFTs are used even on audio samples having noises and provides effective results. The Fourier transforms of a function results in its equivalent frequency for the input amplitude signal. Figure 2 shows how a signal is converted into its frequency signal using the Fourier Transform.

STFT is a method in which FFT transforms are applied once the signal is trimmed by the window function. The mathematical expression for the calculation of the STFT features is shown in Eq. (3).

$$Y(t,f) = STFT(y(t)) = \int_{-\infty}^{\infty} y(u)h^{*}(u-t)e^{-2j\pi fu}du \tag{3}$$

where, $y(t)$ is the original audio signal, $h(t)$ are an STFT window function and the center lies at $t=0$ having a length of $L(0 < L \leq 1500)$. The resulting STFTs are in 2-D form and are shown below.

$$Y(t,f) = \left[y_{1,1}y_{1,2}y_{1,3}\cdots y_{1,n}y_{2,1}y_{2,2}y_{2,3}\ldots y_{2,n}\cdots\cdots\cdots\cdots\cdots\cdots y_{m,1}y_{m,2}y_{m,3}\cdots y_{m,n}\right] \tag{4}$$

where STFT values can be verified from $y_{i,j}$.

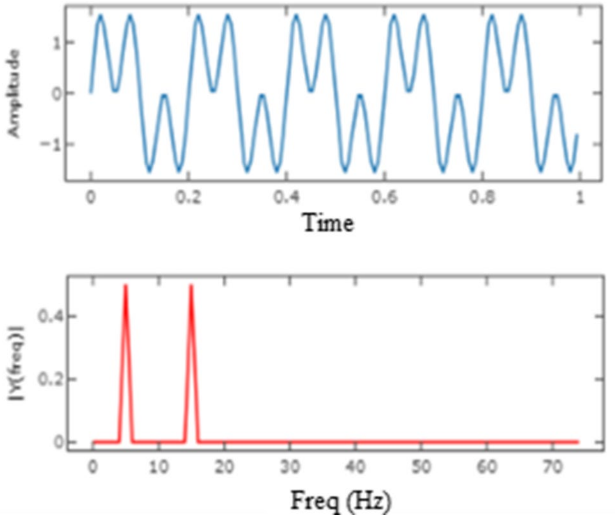


Fig. 2 Equivalent Fourier transformation of the provided digital signal

Table 1 Audio sample distribution in urbansound8k dataset

Class label	Audio samples
Dog bark	1000
Children playing	1000
Car horn	429
Air conditioner	1000
Street Music	1000
Gun shot	374
Siren	929
Engine idling	1000
Jackhammer	1000
Drilling	1000

4.3 Urbansound8k audio dataset

In the study, for the environmental sound classification process, we have used UrbanSound8K audio dataset [44]. The dataset consists of 10 different environmental sound classes as shown in Table 1. All the audio files are generated from real-time instances having a nearly 4 s recording time. We considered a total of 8732 audio samples from this dataset.

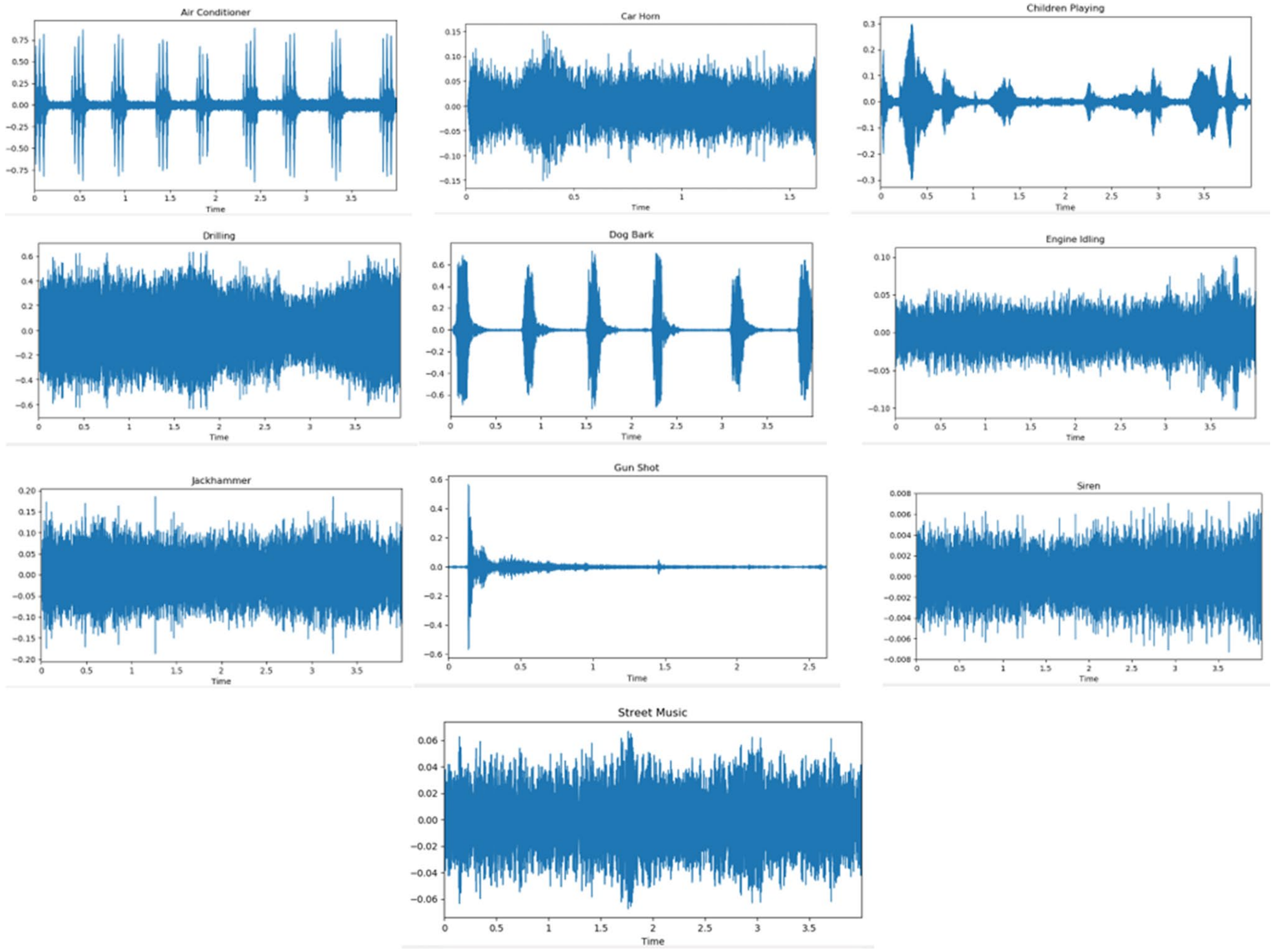


Fig. 3 Waveform visualization for different audio samples for each class present UrbanSound8K Dataset

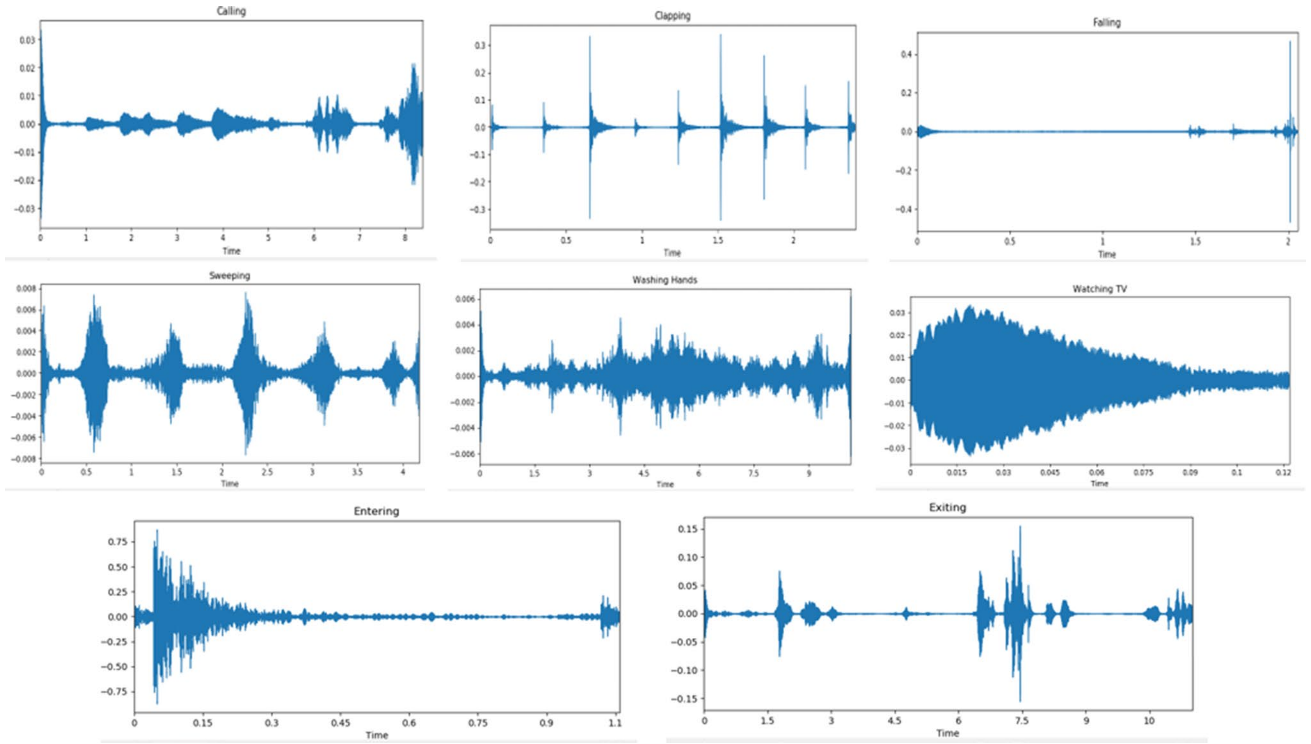


Fig. 4 Audio waveforms for all the classes used in the classification process after the removal of noise and trimming process

The waveforms for different classes have been illustrated in Fig. 3. The samples shown for illustrations are included in the classification. All the figures in Fig. 4 are *Amplitude v/s Time frame* audio graphs. All the waveforms included in Fig. 3 & 4 have amplitude (db) and time (seconds) on Y-axis and X-axis respectively. Furthermore, these images are presented inorder to provide a glimps of how varied frequency waveforms are used for the training of the models used.

For each audio file, we have calculated the MFCC features (40 features for each sample), and depending upon these features we have created the modified dataset which contains all the audio samples with their features extracted from the audio file with their respective class label. After the dataset formation, we have divided the dataset randomly to test and training set for the classification process which is discussed in detail in *Classifier Models* Section.

4.4 Sound event audio dataset

Another dataset used in our research work includes the Sound Event Audio Dataset [44] which is collected from the University of Moratuwa, Sri Lanka. The dataset contains many environmental sound classes, and out of them we selected 7 of them and we also made another category as *others* which includes the sounds from surroundings, door opening, closing, etc. The audio sample distribution for this dataset can be seen in detail in Table 2. All the audio samples were generated from real-time instances using two MOVO USB omnidirectional microphones. From the dataset, all the features of individual audio samples were extracted, and based upon those, classification models were built considering a total of 1288 samples from this dataset.

Since this dataset also contained noise in the samples, we used some preprocessing for removing the noise part and trimming the audio sample without leaving any features behind for getting accurate results. For instance, we have demonstrated the random audio sample from the dataset before removing the noise from it in Fig. 5. Furthermore, Fig. 6 shows the audio sample waveform after the removal of noise from it and results in a much clearer waveform. Finally, the third step includes the trimming of the audio sample after the removal of noise as shown in Fig. 7. For the rest of the audio samples of different classes, we have illustrated the final trimmed waveform which is used for the feature extraction process followed by the classification process in Fig. 4. The samples shown for illustrations are included in the classification. All the figures in Fig. 4 are *Amplitude v/s Time frame* audio graphs.

Table 2 Audio sample distribution in sound event audio dataset

Class label	Audio samples
Calling	102
Clapping	105
Falling	105
Sweeping	51
Washing hands	51
Watching TV	78
Entry/Exit	102
Surrounding	694

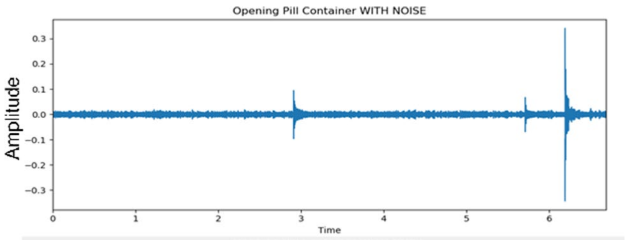


Fig. 5 Opening pill container audio waveform with noise in the sample

This time for every audio sample, we calculated STFT features because of the reason that MFCC features do not hold a good grip on the audio samples which contain noise. The STFT features results in a 2-D array that contains the mentioned frequency amplitude bind for an individual window.

5 Classification models

Machine learning models are widely used for training the various models for either prediction [46, 47] or classifying the given samples in different classes. In the process of classifying the audio samples, we have used seven different machine learning models namely Logistic regression, K-Nearest Neighbors, Support Vector Machine, Naïve Bayes, Decision Tree, and Random Forest classifier. Apart from these six classifiers, we have also used Artificial Neural Network (ANN) architecture-based model for classifying the different classes of audio samples. We have illustrated a comparative study of all the classifiers used as shown in Table 3.

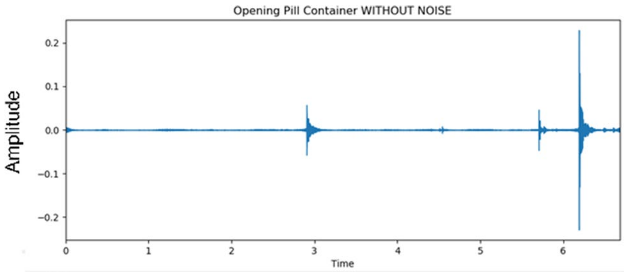


Fig. 6 Opening pill container audio waveform after the removal of noise from the sample

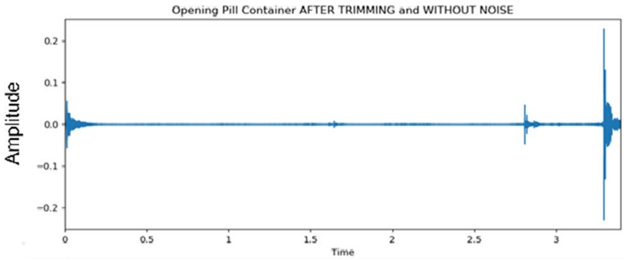


Fig. 7 Opening pill container audio waveform after trimming the noise-free sample

Table 3 Comparative analysis of classifier models

Classifier Model	Advantage	Disadvantage	Method/Working
Logistic Regression	Easy implementation used for optimized training datasets having linear relationships [47] Can be applied to multiple classes as well Fast classifying rate. And have a probabilistic approach Model overfitting can be avoided by using L1 and L2 regularization techniques	Overfitting can be seen in the trained models Works on linear relation among the features and target variables. Non-linear problems can reduce the results and are generally not suitable for this classifier model Based on the working of the model, only linear datasets can be used for this model to provide efficient results	The mathematical expression for finding probability (y) can be: $loglog\left(\frac{y}{1-y}\right) = a_0 + a_1X_1 + a_2X_2 + \cdots + a_nX_n$ Where, a_0 is a constant and the rest $a_1, a_2, \ldots, a_n$ are the regression coefficients, and $X_1, X_2, \ldots, X_n$ are the features which help in the prediction process
K-Nearest Neighbors	Termed as Lazy learner or Instance-based learning models as no training period is required New data records when added, dont affect the results Fast model making, providing efficient results using distance rule for classification	Results lack in large datasets Does not provide good results with high dimensions Normalization of data points are required Noise, outliers, or any missing value is not handled by the model and might result in overfitting Datasets free from noise, or any kind of outliers can be used for efficient results, or else the results might lack in different result parameters	The most frequently used distance formula is Euclidean distance whose mathematical expression is: $d(x_i, x_j) = \sqrt{\sum_{r=1}^{n} (x_{ir} - x_{jr})^2}$
Support Vector Machine	Accurate result outcomes with multi-dimensional data records including variety of kernels Prediction rate is higher in both classification and regression process	Correct configuration selection for model is crucial Unefficient computation in the models for prediction and processing the data samples Scaling is necessary Overfitting can occur and is hard to reduce its effect Although, the model provides effieicnt results, it takes a significant training time, so huge datasets can result in a long training time	It works on the principle of selecting support vectors from each class maximizing the distance between the hyperplane and the nearest vectors. Hyperplane can be described as $px_1 + qx_2$ and the end goal is to find values of q,p and r such that $px_1 + qx_2 \leq r$ is true for class 1 and $px_1 + qx_2 > r$ for class 2
Naïve Bayes	Efficent rate of prediction supporting multi-class classifications Provide acceptable results with minimal data samples assuming features are independent of each other.	Zero frequency phenomena can become hurdle All features are assumed to independent of each other Furhtmore, datasets with multidependent values will form a negative impact on the model because of the inner functioning	The mathematical expression for estimating the probability can be: $P(A\|B) = \frac{P(A) \times P(B\|A)}{P(B)}$
Decision Tree	Simpler data pre-processing with no need of normalization Missing values have no effect while making model Easy interpretations and is readability. Overcome the problem of non-linear variables efficiently	Varying results with minute changes in the data samples. Relatively complex calculations and increased training time and sometimes unstable; not suitable for large data samples. Major effect of noise on the results and easily overfitted	Forms a tree-like structure, starting from the root node and ending at the leaf node. Each leaf node defines a predicted class to which the sample belongs. This prediction depends upon the test conditions of the non-leaf nodes where each branch depicts a decision

Table 3 (continued)

Classifier Model	Advantage	Disadvantage	Method/Working
Random Forest	Overfitting is less likely in classification as well as regression process having either categorical or continuous features More stable when compared with decision tree models without the need of data-preprocessing step	More complex because of large tree computations with increased training time High dimensional dataset will increase the complexity of the model, reducing the interpretability of the model, hence low dimensional dataset is preferable	Forms a group of decision trees by selecting arbitrary data points (data samples). With these selected data points, decision trees are formed and the sample is classified to the class having maximum votes from each tree
Artificial Neural Networks [45]-[47]	Flexible enough so that can be used for regression models as well, trained to handle non-linear values Layers can be multiple based on the requirements with high prediction rate once the model is trained	Complex equations inbetween the layers making it hard to analyze the effect of each neuron on the predicted result High Training time Dependent on training data leading to generalization or overfitting sometimes With large scale datasets, training time by each epoch increases by a slight change. Although it improves the overall accuracy of the model significantly	Every node receives input from the nodes in the previous layer, multiplied by their weights, and mathematically it can be expressed as follows: $S = \sum_{j=1}^{P_{i-1}} W_i^{j,k} N_{i-1}^j, 1 \leq k \leq P_i$ Where, $W_i^{j,k}$ denotes the weight of the respective neuron from the respective layer, N_{i-1}^j denotes the neuron itself, and k iterates through all the neurons present in a specific layer

6 Experimentation

In this section, we have discussed the experimental setup we have used which categorizing the audio samples. One of the unique things of our research work is the comparison of two similar datasets as discussed in Sect. 3 having noises and applying different feature selection techniques. Our experimental aims for the research are as follows.

Extracting the unique and important features from the audio sample using two feature selection techniques namely MFCC and STFT features.
Removal of unwanted features and background noises from the audio sample for achieving better results. The resulted
Applying classifier models as discussed in Sect. 4, and comparing the results based on different parameters namely accuracy, precision, recall, specificity, F1-score, and MCC.

In Sect. 6 we have further compared the results based on the experiments conducted. The basic overview of our approach for categorizing an audio sample from the datasets follows some important steps name: feature extraction, data pre-processing depending upon the sample (whether contains noise or not), and finally various classifier models are applied. All the steps mentioned above can be seen in Fig. 8.

The section is further divided into the following sections: A *Software and Hardware,* B *Data Preprocessing,* and C *Analysis of Classifier Models.*

6.1 Software and hardware

All the classifier models that were used in the experiment were trained through Python 3 with Keras library (using TensorFlow backend) on an anaconda environment. High-level API was used for constructing neural networks as well as other classifier models. we used Intel i5 8th generation processor with 16 GB RAM.

6.2 Data preprocessing

In our experiment, we have considered two datasets having 10 and 8 different classes respectively. The two datasets are discussed in detail in Sect. 3. In UrbanSound8K audio dataset we have applied the MFCC feature selection process because of its noise-free samples. However, in the other dataset, since the samples contain some background noise and other unwanted features, we have applied the STFT feature selection process because MFCCs are sensitive to noise as discussed in the above literature. After the collection of all the features separately for the two different datasets, we have applied the splitting of datasets into training and testing sets respectively. After the splitting, the training set contains 80% of the audio samples from the dataset and the testing set contains the remaining randomly selected 20% audio samples for prediction using the trained models. This distribution is common for both datasets.

6.3 Analysis of classifier models

After categorizing the different classes of the audio sample after data preprocessing using various classifier models, we conclude that the ANN model achieves the best accuracy among all the classifiers. However, some classifiers gave better results depending upon the datasets. This happened because of the internal working and classification criteria of different classifier models. For a better and detailed comparison, we have shown the confusion matrix of different models in Fig. 9. Figure 10 comprises the True Negatives (TN), True Positives (TP), False Positives (FP), and False Negatives (FN). Figures 9 and 10 comprise the results of the UrbanSound8K Audio dataset. Although the Linear Regression model was able to classify most of the audio samples to their correct category as shown in Fig. 9, GS (*Gun Shot*) was poorly classified. One of the reasons that support the outcomes is the GS and DB have similar waveforms after the feature extraction and trimming process. This can be seen from Fig. 3, and because of this, the model misclassified the GS samples to the DB category. Table 4 contains the various abbreviations used in Figs. 9 and 10.

Similarly, Fig. 11 shows the confusion matrix of all the classifier models that are used in the second dataset (Sound Event Audio Dataset). And Fig. 12 shows the FP, FN, TP, and TN parameters of classifier models. In the next section (Analysis and Result) we have discussed the detailed parameters based on which we conclude that the ANN model achieves the best score among all. Table 5 contains the various abbreviations used in Figs. 11 and 12.

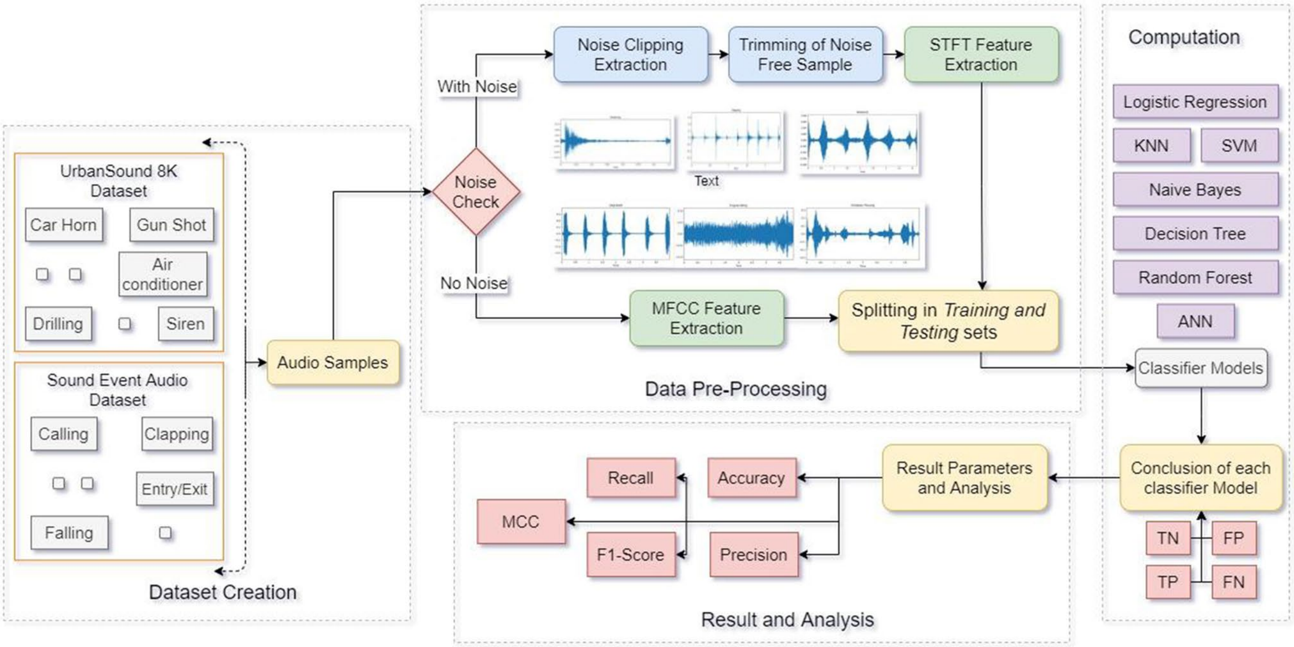
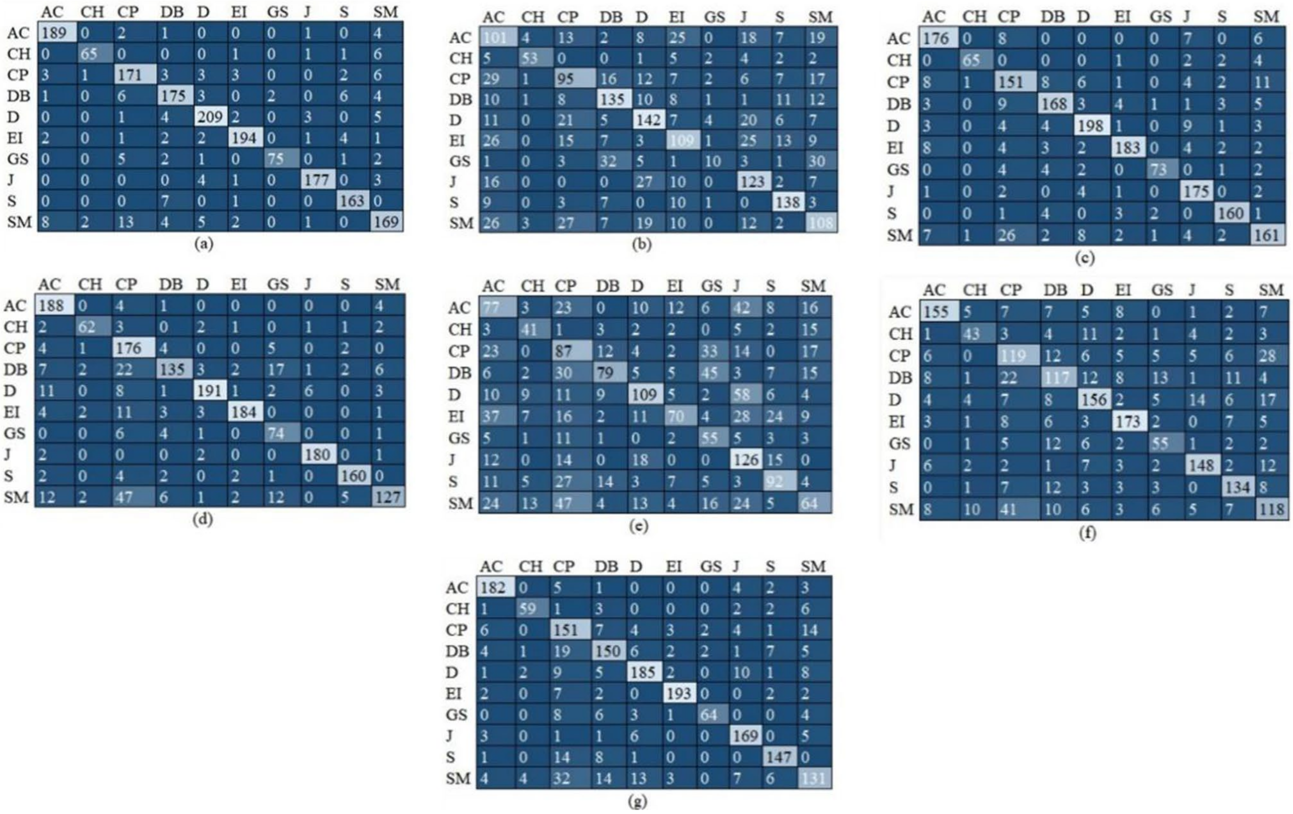


Fig. 8 Flowchart of various steps followed for classification of audio samples



	AC	CH	CP	DB	D	EI	GS	J	S	SM
AC	189	0	2	1	0	0	0	1	0	4
CH	0	65	0	0	0	1	0	1	1	6
CP	3	1	171	3	3	3	0	0	2	6
DB	1	0	6	175	3	0	2	0	6	4
D	0	0	1	4	209	2	0	3	0	5
EI	2	0	1	2	2	194	0	1	4	1
GS	0	0	5	2	1	0	75	0	1	2
J	0	0	0	0	4	1	0	177	0	3
S	0	0	0	7	0	1	0	0	163	0
SM	8	2	13	4	5	2	0	1	0	169

(a)

	AC	CH	CP	DB	D	EI	GS	J	S	SM
AC	101	4	13	2	8	25	0	18	7	19
CH	5	53	0	0	1	5	2	4	2	2
CP	29	1	95	16	12	7	2	6	7	17
DB	10	1	8	135	10	8	1	1	11	12
D	11	0	21	5	142	7	4	20	6	7
EI	26	0	15	7	3	109	1	25	13	9
GS	1	0	3	32	5	1	10	3	1	30
J	16	0	0	0	27	10	0	123	2	7
S	9	0	3	7	0	10	1	0	138	3
SM	26	3	27	7	19	10	0	12	2	108

(b)

	AC	CH	CP	DB	D	EI	GS	J	S	SM
AC	176	0	8	0	0	0	0	7	0	6
CH	0	65	0	0	0	1	0	2	2	4
CP	8	1	151	8	6	1	0	4	2	11
DB	3	0	9	168	3	4	1	1	3	5
D	3	0	4	4	198	1	0	9	1	3
EI	8	0	4	3	2	183	0	4	2	2
GS	0	0	4	4	2	0	73	0	1	2
J	1	0	2	0	4	1	0	175	0	2
S	0	0	1	4	0	3	2	0	160	1
SM	7	1	26	2	8	2	1	4	2	161

(c)

	AC	CH	CP	DB	D	EI	GS	J	S	SM
AC	188	0	4	1	0	0	0	0	0	4
CH	2	62	3	0	2	1	0	1	1	2
CP	4	1	176	4	0	0	5	0	2	0
DB	7	2	22	135	3	2	17	1	2	6
D	11	0	8	1	191	1	2	6	0	3
EI	4	2	11	3	3	184	0	0	0	1
GS	0	0	6	4	1	0	74	0	0	1
J	2	0	0	0	2	0	0	180	0	1
S	2	0	4	2	0	2	1	0	160	0
SM	12	2	47	6	1	2	12	0	5	127

(d)

	AC	CH	CP	DB	D	EI	GS	J	S	SM
AC	77	3	23	0	10	12	6	42	8	16
CH	3	41	1	3	2	2	0	5	2	15
CP	23	0	87	12	4	2	33	14	0	17
DB	6	2	30	79	5	5	45	3	7	15
D	10	9	11	9	109	5	2	58	6	4
EI	37	7	16	2	11	70	4	28	24	9
GS	5	1	11	1	0	2	55	5	3	3
J	12	0	14	0	18	0	0	126	15	0
S	11	5	27	14	3	7	5	3	92	4
SM	24	13	47	4	13	4	16	24	5	64

(e)

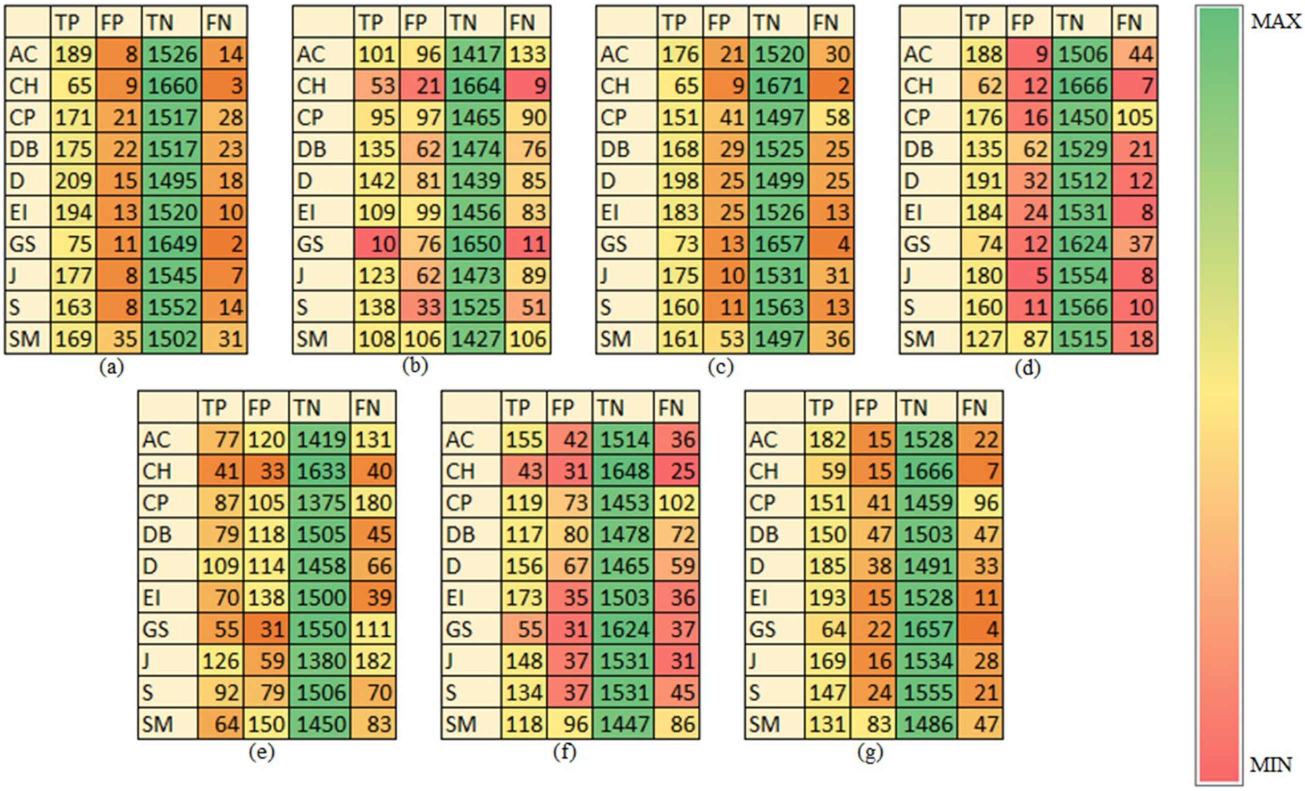
	AC	CH	CP	DB	D	EI	GS	J	S	SM
AC	155	5	7	7	5	8	0	1	2	7
CH	1	43	3	4	11	2	1	4	2	3
CP	6	0	119	12	6	5	5	5	6	28
DB	8	1	22	117	12	8	13	1	11	4
D	4	4	7	8	156	2	5	14	6	17
EI	3	1	8	6	3	173	2	0	7	5
GS	0	1	5	12	6	2	55	1	2	2
J	6	2	2	1	7	3	2	148	2	12
S	0	1	7	12	3	3	3	0	134	8
SM	8	10	41	10	6	3	6	5	7	118

(f)

	AC	CH	CP	DB	D	EI	GS	J	S	SM
AC	182	0	5	1	0	0	0	4	2	3
CH	1	59	1	3	0	0	0	2	2	6
CP	6	0	151	7	4	3	2	4	1	14
DB	4	1	19	150	6	2	2	1	7	5
D	1	2	9	5	185	2	0	10	1	8
EI	2	0	7	2	0	193	0	0	2	2
GS	0	0	8	6	3	1	64	0	0	4
J	3	0	1	1	6	0	0	169	0	5
S	1	0	14	8	1	0	0	0	147	0
SM	4	4	32	14	13	3	0	7	6	131

(g)

Fig. 9 Confusion matrix of different classifier models which are used in the *UrbanSound8K* dataset (**a**. ANN model, **b**. Logistic Regression model, **c**. SVM (rbf) model, **d**. KNN model, **e**. Naïve Bayes model, f. Decision Tree model, g. Random Forest model)



(a)

	TP	FP	TN	FN
AC	189	8	1526	14
CH	65	9	1660	3
CP	171	21	1517	28
DB	175	22	1517	23
D	209	15	1495	18
EI	194	13	1520	10
GS	75	11	1649	2
J	177	8	1545	7
S	163	8	1552	14
SM	169	35	1502	31

(b)

	TP	FP	TN	FN
AC	101	96	1417	133
CH	53	21	1664	9
CP	95	97	1465	90
DB	135	62	1474	76
D	142	81	1439	85
EI	109	99	1456	83
GS	10	76	1650	11
J	123	62	1473	89
S	138	33	1525	51
SM	108	106	1427	106

(c)

	TP	FP	TN	FN
AC	176	21	1520	30
CH	65	9	1671	2
CP	151	41	1497	58
DB	168	29	1525	25
D	198	25	1499	25
EI	183	25	1526	13
GS	73	13	1657	4
J	175	10	1531	31
S	160	11	1563	13
SM	161	53	1497	36

(d)

	TP	FP	TN	FN
AC	188	9	1506	44
CH	62	12	1666	7
CP	176	16	1450	105
DB	135	62	1529	21
D	191	32	1512	12
EI	184	24	1531	8
GS	74	12	1624	37
J	180	5	1554	8
S	160	11	1566	10
SM	127	87	1515	18

(e)

	TP	FP	TN	FN
AC	77	120	1419	131
CH	41	33	1633	40
CP	87	105	1375	180
DB	79	118	1505	45
D	109	114	1458	66
EI	70	138	1500	39
GS	55	31	1550	111
J	126	59	1380	182
S	92	79	1506	70
SM	64	150	1450	83

(f)

	TP	FP	TN	FN
AC	155	42	1514	36
CH	43	31	1648	25
CP	119	73	1453	102
DB	117	80	1478	72
D	156	67	1465	59
EI	173	35	1503	36
GS	55	31	1624	37
J	148	37	1531	31
S	134	37	1531	45
SM	118	96	1447	86

(g)

	TP	FP	TN	FN
AC	182	15	1528	22
CH	59	15	1666	7
CP	151	41	1459	96
DB	150	47	1503	47
D	185	38	1491	33
EI	193	15	1528	11
GS	64	22	1657	4
J	169	16	1534	28
S	147	24	1555	21
SM	131	83	1486	47

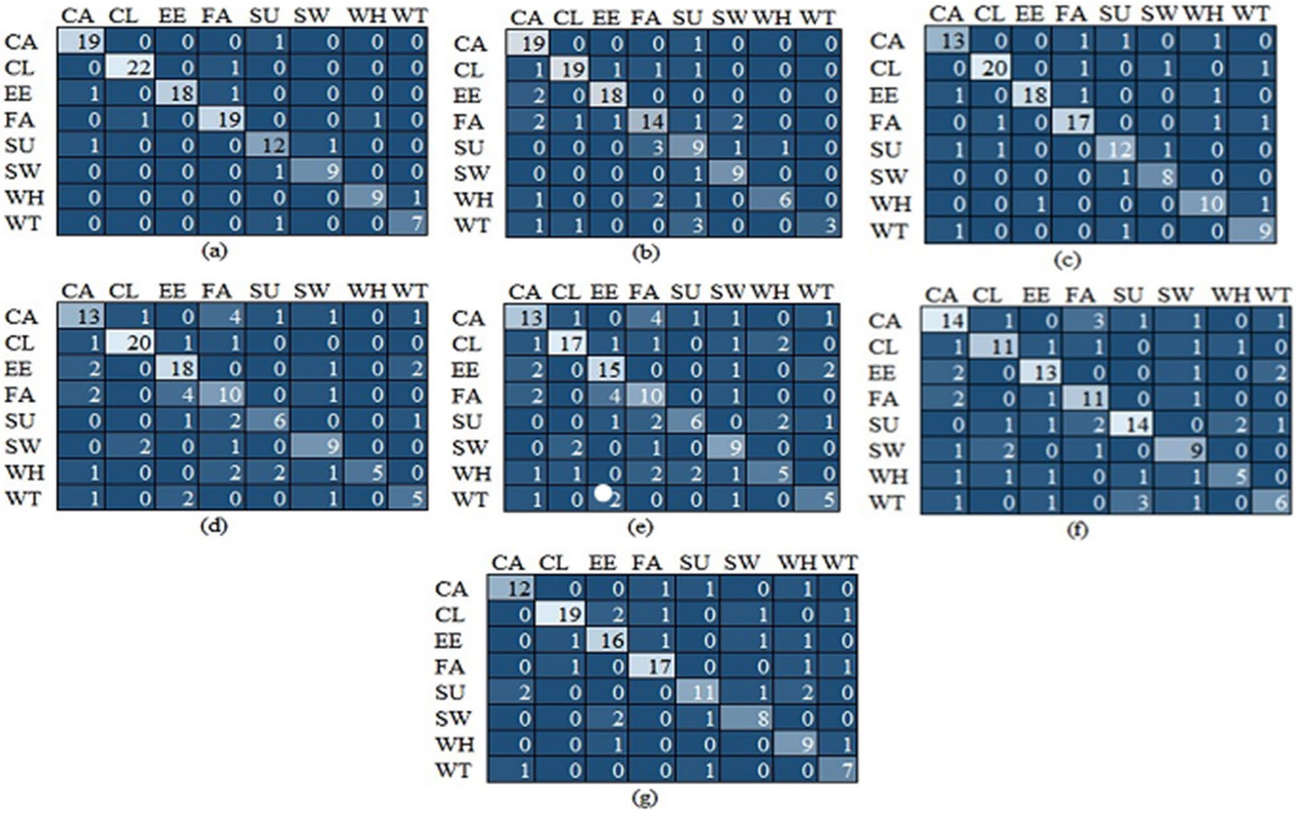
Fig. 10 TP, TN, FP, and FN parameters for different classifier models used in *UrbanSound8K* audio dataset (**a**. ANN model, **b**. Logistic Regression model, **c**. SVM (rbf) model, **d**. KNN model, **e**. Naïve Bayes model, **f**. Decision Tree model, **g**. Random Forest model)

Table 4 Abbreviations used in urbansound8k dataset

Abbreviations	Meaning
AC	Air conditioner
CH	Car horn
CP	Children playing
DB	Dog bark
D	Drilling
EI	Engine idling
GS	Gun shot
J	Jackhammer
S	Siren
SM	Street music

7 Results

We have evaluated different parameters namely accuracy, precision, recall, specificity, F1-score, and Matthews Correlation Coefficient (MCC) for comparison of different models. In our experiment, we can conclude that we got the best results in the ANN model for both datasets. The mathematical expressions used for evaluating different parameters are shown as follows.



(a)

	CA	CL	EE	FA	SU	SW	WH	WT
CA	19	0	0	0	1	0	0	0
CL	0	22	0	1	0	0	0	0
EE	1	0	18	1	0	0	0	0
FA	0	1	0	19	0	0	1	0
SU	1	0	0	0	12	1	0	0
SW	0	0	0	0	1	9	0	0
WH	0	0	0	0	0	0	9	1
WT	0	0	0	0	1	0	0	7

(b)

	CA	CL	EE	FA	SU	SW	WH	WT
CA	19	0	0	0	1	0	0	0
CL	1	19	1	1	1	0	0	0
EE	2	0	18	0	0	0	0	0
FA	2	1	1	14	1	2	0	0
SU	0	0	0	3	9	1	1	0
SW	0	0	0	0	1	9	0	0
WH	1	0	0	2	1	0	6	0
WT	1	1	0	0	3	0	0	3

(c)

	CA	CL	EE	FA	SU	SW	WH	WT
CA	13	0	0	1	1	0	1	0
CL	0	20	0	1	0	1	0	1
EE	1	0	18	1	0	0	1	0
FA	0	1	0	17	0	0	1	1
SU	1	1	0	0	12	1	0	0
SW	0	0	0	0	1	8	0	0
WH	0	0	1	0	0	0	10	1
WT	1	0	0	0	1	0	0	9

(d)

	CA	CL	EE	FA	SU	SW	WH	WT
CA	13	1	0	4	1	1	0	1
CL	1	20	1	1	0	0	0	0
EE	2	0	18	0	0	1	0	2
FA	2	0	4	10	0	1	0	0
SU	0	0	1	2	6	0	0	1
SW	0	2	0	1	0	9	0	0
WH	1	0	0	2	2	1	5	0
WT	1	0	2	0	0	1	0	5

(e)

	CA	CL	EE	FA	SU	SW	WH	WT
CA	13	1	0	4	1	1	0	1
CL	1	17	1	1	0	1	2	0
EE	2	0	15	0	0	1	0	2
FA	2	0	4	10	0	1	0	0
SU	0	0	1	2	6	0	2	1
SW	0	2	0	1	0	9	0	0
WH	1	1	0	2	2	1	5	0
WT	1	0	2	0	0	1	0	5

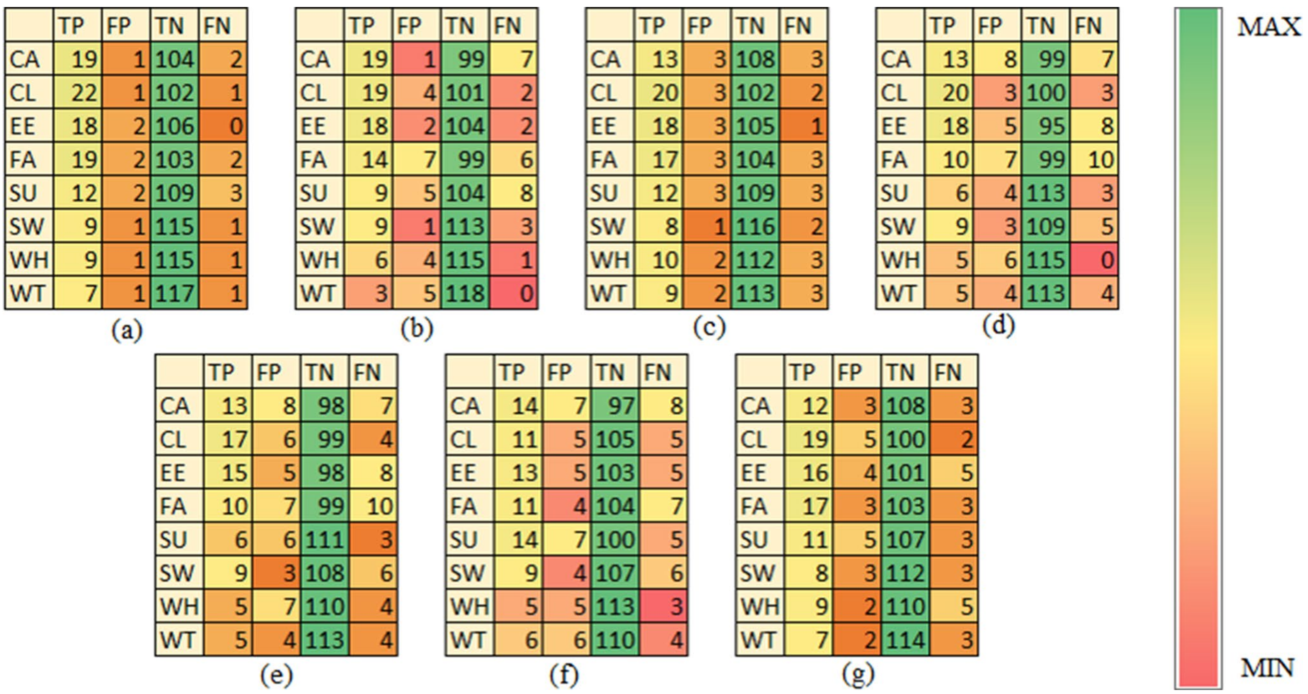
(f)

	CA	CL	EE	FA	SU	SW	WH	WT
CA	14	1	0	3	1	1	0	1
CL	1	11	1	1	0	1	1	0
EE	2	0	13	0	0	1	0	2
FA	2	0	1	11	0	1	0	0
SU	0	1	1	2	14	0	2	1
SW	1	2	0	1	0	9	0	0
WH	1	1	1	0	1	1	5	0
WT	1	0	1	0	3	1	0	6

(g)

	CA	CL	EE	FA	SU	SW	WH	WT
CA	12	0	0	1	1	0	1	0
CL	0	19	2	1	0	1	0	1
EE	0	1	16	1	0	1	1	0
FA	0	1	0	17	0	0	1	1
SU	2	0	0	0	11	1	2	0
SW	0	0	2	0	1	8	0	0
WH	0	0	1	0	0	0	9	1
WT	1	0	0	0	1	0	0	7

Fig. 11 Confusion matrix of different classifier models which are used in the *Sound Event Audio Classification* dataset (**a** ANN model, **b** Logistic Regression model, **c** SVM (rbf) model, **d** KNN model, **e** Naïve Bayes model, **f** Decision Tree model, **g** Random Forest model)



(a)

	TP	FP	TN	FN
CA	19	1	104	2
CL	22	1	102	1
EE	18	2	106	0
FA	19	2	103	2
SU	12	2	109	3
SW	9	1	115	1
WH	9	1	115	1
WT	7	1	117	1

(b)

	TP	FP	TN	FN
CA	19	1	99	7
CL	19	4	101	2
EE	18	2	104	2
FA	14	7	99	6
SU	9	5	104	8
SW	9	1	113	3
WH	6	4	115	1
WT	3	5	118	0

(c)

	TP	FP	TN	FN
CA	13	3	108	3
CL	20	3	102	2
EE	18	3	105	1
FA	17	3	104	3
SU	12	3	109	3
SW	8	1	116	2
WH	10	2	112	3
WT	9	2	113	3

(d)

	TP	FP	TN	FN
CA	13	8	99	7
CL	20	3	100	3
EE	18	5	95	8
FA	10	7	99	10
SU	6	4	113	3
SW	9	3	109	5
WH	5	6	115	0
WT	5	4	113	4

(e)

	TP	FP	TN	FN
CA	13	8	98	7
CL	17	6	99	4
EE	15	5	98	8
FA	10	7	99	10
SU	6	6	111	3
SW	9	3	108	6
WH	5	7	110	4
WT	5	4	113	4

(f)

	TP	FP	TN	FN
CA	14	7	97	8
CL	11	5	105	5
EE	13	5	103	5
FA	11	4	104	7
SU	14	7	100	5
SW	9	4	107	6
WH	5	5	113	3
WT	6	6	110	4

(g)

	TP	FP	TN	FN
CA	12	3	108	3
CL	19	5	100	2
EE	16	4	101	5
FA	17	3	103	3
SU	11	5	107	3
SW	8	3	112	3
WH	9	2	110	5
WT	7	2	114	3

Fig. 12 TP, TN, FP, and FN parameters for different classifier models used in *Sound Event Audio* dataset (**a** ANN model, **b** Logistic Regression model, **c** SVM (rbf) model, **d** KNN model, **e** Naïve Bayes model, **f** Decision Tree model, **g** Random Forest model)

Table 5 Abbreviations used in sound event audio dataset

Abbreviations	Meaning
CA	Calling
CL	Clapping
EE	Entry/Exit
FA	Falling
SU	Surrounding
SW	Sweeping
WH	Washing hands
WT	Watching TV

$$Accuracy = \frac{TN + TP}{TN + TP + FP + FN} \tag{5}$$

$$Precision = \frac{TP}{TP + FP} \tag{6}$$

$$Recall = \frac{TP}{TP + FN} \tag{7}$$

$$F1 - Score = \frac{2 \times TP}{2 \times TP + FP + FN} \tag{8}$$

$$MCC = \frac{TN \times TP - FP \times FN}{\sqrt{(TN + FN)(FP + TP)(TN + FP)(FN + TP)}} \tag{9}$$

In our experiment, precision is referred to as the number of correctly predicted audio classes that turned out to be positive, recall tells about the number of actual positive cases that are predicted correctly with our models, specificity is the proportion of negative cases that are being predicted correctly, and F1-score refers to the harmonic mean of recall and precision or in other words, it provides a combined idea about the two results (recall and precision). F1-score is maximum when precision is equal to recall.

Apart from these matrices, there exists another parameter namely *phi-coefficient* (φ). From Eq. (9) we can see that MCC takes all the parameters (TP, FP, TN, and FN) into account, while other metrics like accuracy, precision, recall, etc. lacks in taking all the four parameters hence making it sensitive to class imbalance and are asymmetric. MCC value can range from 1 to −1 depending upon the correlation. If the MCC is 1 (FP = FN = 0), indicates perfect positive correlation. On the other hand, if the MCC is −1 (TP = TN = 0), indicates a perfect negative correlation, such conditions depict that the classifier always misclassifies the classes.

However, if the MCC is 0, represents that classifier randomly choosing any class. For instance, taking the ANN model and Logistic Regression model into account from Table 5, one can see that MCC is 0.9380 and 0.2177 respectively for different subclasses. From this, we can imply that the ANN model has predicted more positively correlated results than the Logistic Regression model.

Tables 6 and 7 shows every parameter evaluated for each model applied to the *UrbanSound8K* and *Sound Event Audio* dataset in classifying each class of the audio sample respectively.

8 Discussion

To have accurate classification of audio samples using various models, feature extraction and noise cancellation play an important role. Apart from the features and noise present in the audio sample, the difference between two different audio samples belonging to different classes should be appropriate. This can be justified by seeing Figs. 13 and 14 which are being plotted based on the results generated from the experiment on the two datasets. The graphs

Table 6 Results of various models used in urbansound8k dataset

Classifier model	Audio class	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)	MCC (%)
ANN	Air conditioner	98.73	95.94	93.10	99.48	94.50	93.80
	Car horn	**99.31**	87.84	95.59	99.46	91.55	91.28
	Children playing	97.18	89.06	85.93	98.63	87.47	85.90
	Dog bark	97.41	88.83	88.38	98.57	88.61	87.15
	Drilling	98.10	93.30	92.07	99.01	92.68	91.59
	Engine idling	98.68	93.72	95.10	99.15	94.40	93.66
	Gun shot	99.25	87.21	97.40	99.34	92.02	91.79
	Jackhammer	99.14	95.68	96.20	99.48	95.93	95.45
	Siren	98.73	95.32	92.09	99.49	93.68	92.99
	Street music	96.20	82.84	84.50	97.72	83.66	81.52
Logistic Regression	Air conditioner	86.89	51.27	43.16	93.65	46.87	39.64
	Car horn	**98.28**	71.62	85.48	98.75	77.94	77.38
	Children playing	89.30	49.48	51.35	93.79	50.40	44.41
	Dog bark	92.10	68.53	63.98	95.96	66.18	61.76
	Drilling	90.50	63.68	62.56	94.67	63.11	57.66
	Engine idling	89.58	52.40	56.77	93.63	54.50	48.68
	Gun shot	95.02	11.63	47.62	95.60	18.69	21.77
	Jackhammer	91.36	66.49	58.02	95.96	61.96	57.28
	Siren	95.19	80.70	73.02	97.88	76.67	74.11
	Street music	87.86	50.47	50.47	93.09	50.47	43.55
Support Vector Machine (rbf)	Air conditioner	97.08	89.34	85.44	98.64	87.34	85.72
	Car horn	**99.37**	87.84	97.01	99.46	92.20	91.99
	Children playing	94.33	78.65	72.25	97.33	75.31	72.20
	Dog bark	96.91	85.28	87.05	98.13	86.15	84.42
	Drilling	97.14	88.79	88.79	98.36	88.79	87.15
	Engine idling	97.82	87.98	93.37	98.39	90.59	89.42
	Gun shot	99.03	84.88	94.81	99.22	89.57	89.21
	Jackhammer	97.65	94.59	84.95	99.35	89.51	88.36
	Siren	98.63	93.57	92.49	99.30	93.02	92.26
	Street music	94.91	75.23	81.73	96.58	78.35	75.55
K-Nearest Neighbors	Air conditioner	96.97	95.43	81.03	99.41	87.65	86.30
	Car horn	98.91	83.78	89.86	99.28	86.71	86.20
	Children playing	93.07	91.67	62.63	98.91	74.42	72.29
	Dog bark	95.25	68.53	86.54	96.10	76.49	74.51
	Drilling	97.48	85.65	94.09	97.93	89.67	88.37
	Engine idling	98.17	88.46	95.83	98.46	92.00	91.06
	Gun shot	97.20	86.05	66.67	99.27	75.13	74.34
	Jackhammer	**99.26**	97.30	95.74	99.68	96.51	96.10
	Siren	98.80	93.57	94.12	99.30	93.84	93.18
	Street music	93.99	59.35	87.59	94.57	70.75	69.13

Table 6 (continued)

Classifier model	Audio class	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-Score (%)	MCC (%)
Naïve bayes	Air conditioner	85.63	39.09	37.02	92.20	38.02	29.92
	Car horn	**95.82**	55.41	50.62	98.02	52.90	50.78
	Children playing	83.69	45.31	32.58	92.91	37.91	29.32
	Dog bark	90.67	40.10	63.71	92.73	49.22	45.82
	Drilling	89.70	48.88	62.29	92.75	54.77	49.51
	Engine idling	89.87	33.65	64.22	91.58	44.16	41.67
	Gun shot	91.87	63.95	33.13	98.04	43.65	42.25
	Jackhammer	86.20	68.11	40.91	95.90	51.12	45.59
	Siren	91.47	53.80	56.79	95.02	55.26	50.57
	Street music	86.66	29.91	43.54	90.62	35.46	28.93
Decision tree	Air conditioner	95.54	78.68	81.15	97.30	79.90	77.40
	Car horn	**96.79**	58.11	63.24	98.15	60.56	58.95
	Children playing	89.98	61.98	53.85	95.22	57.63	52.14
	Dog bark	91.30	59.39	61.90	94.87	60.62	55.75
	Drilling	92.79	69.96	72.56	95.63	71.23	67.13
	Engine idling	95.94	83.17	82.78	97.72	82.97	80.67
	Gun shot	96.11	63.95	59.78	98.13	61.80	59.79
	Jackhammer	96.11	80.00	82.68	97.64	81.32	79.16
	Siren	95.31	78.36	74.86	97.64	76.57	73.99
	Street music	89.58	55.14	57.84	93.78	56.46	50.57
Random Forest	Air conditioner	97.88	92.39	89.22	99.03	90.77	89.59
	Car horn	**98.74**	79.73	89.39	99.11	84.29	83.78
	Children playing	92.16	78.65	61.13	97.27	68.79	65.06
	Dog bark	94.62	76.14	76.14	96.97	76.14	73.11
	Drilling	95.94	82.96	84.86	97.51	83.90	81.49
	Engine idling	98.51	92.79	94.61	99.03	93.69	92.85
	Gun shot	98.51	74.42	94.12	98.69	83.12	82.97
	Jackhammer	97.48	91.35	85.79	98.97	88.48	87.12
	Siren	97.42	85.96	87.50	98.48	86.73	85.30
	Street music	92.56	61.21	73.60	94.71	66.84	63.02

Bold represent the highest values

show the plot of accuracy for each class in the respective dataset classified by each machine learning model. We have also illustrated class-wise best accurate models in Table 8.

From Fig. 13, the Naïve Bayes model had performed the poorest among the models in accurately classifying the CP (*Children Playing*) class and has achieved poor results in comparison to other classifier models. This can be due to noise present in the audio samples and the internal processing of the Naïve Bayes model for determining the class to which the audio sample should belong. However, this does not imply that Naïve Bayes is a not-so-good approach for classification. Some studies show that Naïve Bayes performs very well in other fields [52, 53]. Similarly, in Fig. 14, FA (*Falling*) class has been poorly classified by Naïve Bayes and KNN model. By looking at one of the waveforms shown in Fig. 7 belonging to the FA class, one can see that the segments of that waveform can be matched with other classes too, resulting in false classification as stated above. From Figs. 13 and 14, the Artificial Neural Network model has achieved maximum accuracy for all the audio sample classes. This is because of the working of the ANN model, having multi-layer neural networks for classifying and analyzing every feature of the audio sample, and depending upon the result generated from each neuron, the final prediction is noted.

Table 7 Results of various models used in sound event audio dataset

Classifier model	Audio class	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-score (%)	MCC (%)
ANN	Calling	97.62	95.00	90.48	99.05	92.68	91.30
	Clapping	98.41	95.65	95.65	99.03	95.65	94.68
	Entry/exit	**98.41**	90.00	100.00	98.15	94.74	93.99
	Falling	96.83	90.48	90.48	98.10	90.48	88.57
	Surrounding	96.03	85.71	80.00	98.20	82.76	80.58
	Sweeping	98.41	90.00	90.00	99.14	90.00	89.14
	Washing Hands	98.41	90.00	90.00	99.14	90.00	89.14
	Watching TV	98.41	87.50	87.50	99.15	87.50	86.65
Logistic regression	Calling	93.65	95.00	73.08	99.00	82.61	79.82
	Clapping	95.24	82.61	90.48	96.19	86.36	83.61
	Entry/exit	**96.83**	90.00	90.00	98.11	90.00	88.11
	Falling	89.68	66.67	70.00	93.40	68.29	62.16
	Surrounding	89.68	64.29	52.94	95.41	58.06	52.57
	Sweeping	96.83	90.00	75.00	99.12	81.82	80.49
	Washing Hands	96.03	60.00	85.71	96.64	70.59	69.79
	Watching TV	96.03	37.50	100.00	95.93	54.55	59.98
Support vector machine (rbf)	Calling	95.28	81.25	81.25	97.30	81.25	78.55
	Clapping	96.06	86.96	90.91	97.14	88.89	86.53
	Entry/exit	96.85	85.71	94.74	97.22	90.00	88.29
	Falling	95.28	85.00	85.00	97.20	85.00	82.20
	Surrounding	95.28	80.00	80.00	97.32	80.00	77.32
	Sweeping	**97.64**	88.89	80.00	99.15	84.21	83.07
	Washing Hands	96.06	83.33	76.92	98.25	80.00	77.90
	Watching TV	96.06	81.82	75.00	98.26	78.26	76.19
K-nearest neighbors	Calling	88.10	61.90	65.00	92.45	63.41	56.40
	Clapping	**95.24**	86.96	86.96	97.09	86.96	84.04
	Entry/exit	89.68	78.26	69.23	95.00	73.47	67.29
	Falling	86.51	58.82	50.00	93.40	54.05	46.42
	Surrounding	94.44	60.00	66.67	96.58	63.16	60.26
	Sweeping	93.65	75.00	64.29	97.32	69.23	65.96
	Washing Hands	95.24	45.45	100.00	95.04	62.50	65.73
	Watching TV	93.65	55.56	55.56	96.58	55.56	52.14
Naïve bayes	Calling	88.10	61.90	65.00	92.45	63.41	56.33
	Clapping	92.06	73.91	80.95	94.29	77.27	72.59
	Entry/exit	89.68	75.00	65.22	95.15	69.77	63.81
	Falling	86.51	58.82	50.00	93.40	54.05	46.42
	Surrounding	92.86	50.00	66.67	94.87	57.14	53.99
	Sweeping	92.86	75.00	60.00	97.30	66.67	63.21
	Washing Hands	91.27	41.67	55.56	94.02	47.62	43.49
	Watching TV	**93.65**	55.56	55.56	96.58	55.56	52.14
Decision tree	Calling	88.10	66.67	63.64	93.27	65.12	57.97
	Clapping	92.06	68.75	68.75	95.45	68.75	64.20
	Entry/exit	92.06	72.22	72.22	95.37	72.22	67.59
	Falling	91.27	73.33	61.11	96.30	66.67	62.03
	Surrounding	90.48	66.67	73.68	93.46	70.00	64.47
	Sweeping	92.06	69.23	60.00	96.40	64.29	60.04
	Washing Hands	**93.65**	50.00	62.50	95.76	55.56	52.56
	Watching TV	92.06	50.00	60.00	94.83	54.55	50.49

Table 7 (continued)

Classifier model	Audio class	Accuracy (%)	Precision (%)	Recall (%)	Specificity (%)	F1-score (%)	MCC (%)
Random forest	Calling	95.24	80.00	80.00	97.30	80.00	77.30
	Clapping	94.44	79.17	90.48	95.24	84.44	81.35
	Entry/exit	92.86	80.00	76.19	96.19	78.05	73.82
	Falling	**95.24**	85.00	85.00	97.17	85.00	82.17
	Surrounding	93.65	68.75	78.57	95.54	73.33	69.95
	Sweeping	95.24	72.73	72.73	97.39	72.73	70.12
	Washing Hands	94.44	81.82	64.29	98.21	72.00	69.58
	Watching TV	96.03	77.78	70.00	98.28	73.68	71.66

Bold represent the highest values

Using Table 8, one can analyse the performance of the classifier models in terms of the achieved accuracies for respective classes presented in the UrbanSound8K dataset. On the otherhand, for the Sound Event Audio Dataset, it was Aritifical Neural model which achieved the relative highest accuracies among the models that were trained on that dataset.

9 Conclusion

In this paper, we have implemented two ways of data preprocessing namely MFCC and STFT by the virtue of which we can classify different audio samples. Our result shows that MFCC features are sensitive to any kind of noise. Our study shows that after all the data preprocessing, the ANN model achieves the best results in classifying both types of the dataset (with and without noise). The overall accuracies achieved by various classifiers in the *UrbamSound8K* dataset and *Sound Event Audio* dataset are listed as follows in Table 9.

From Table 9, it can be seen that the results of Logistic Regression and Naïve Bayes are highly varied across the two datasets. The main reason behind this is the working of the classifier model as well as the relationship formed between the audio sample points with the predicted sample point. Logistic regression forms a linear relationship among the features while on the other hand, Naïve Bayes model assumes total independencies between the features. From the analysis of the results presented in the Table 9, it can be inferred that there is a direct relationship of accurate prediction and the distinct features a class have. Artificial Neural Network being an adaptive model helps in handling audio samples with heteroskedasticity (samples with different variances), it can be seen that most of the best results are achieved by ANN models.

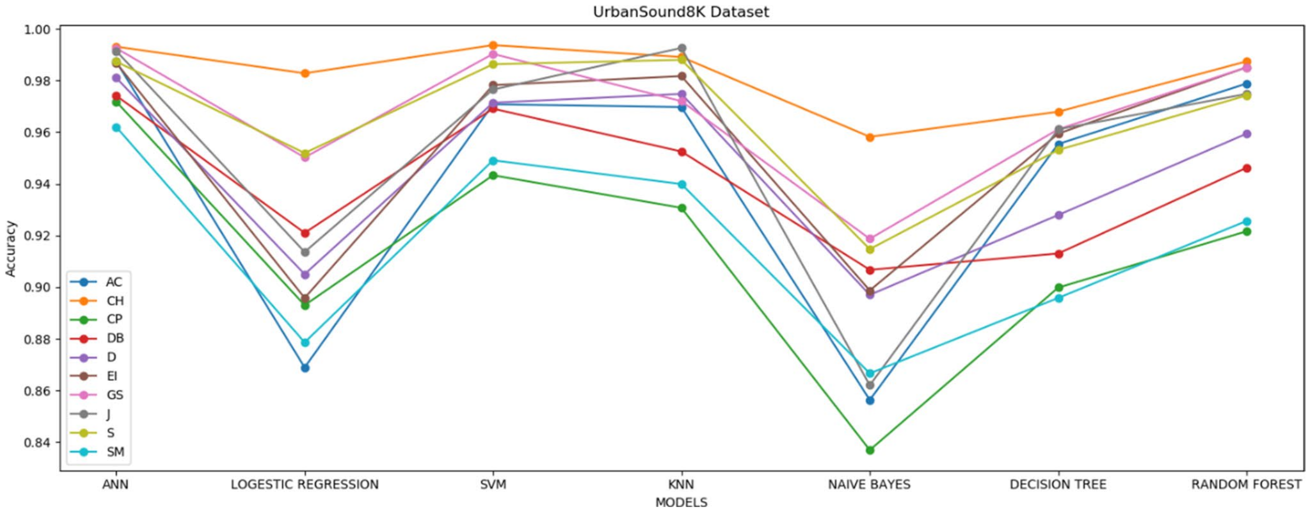


Fig. 13 Accuracy graph for various models used in UrbanSound8K Dataset

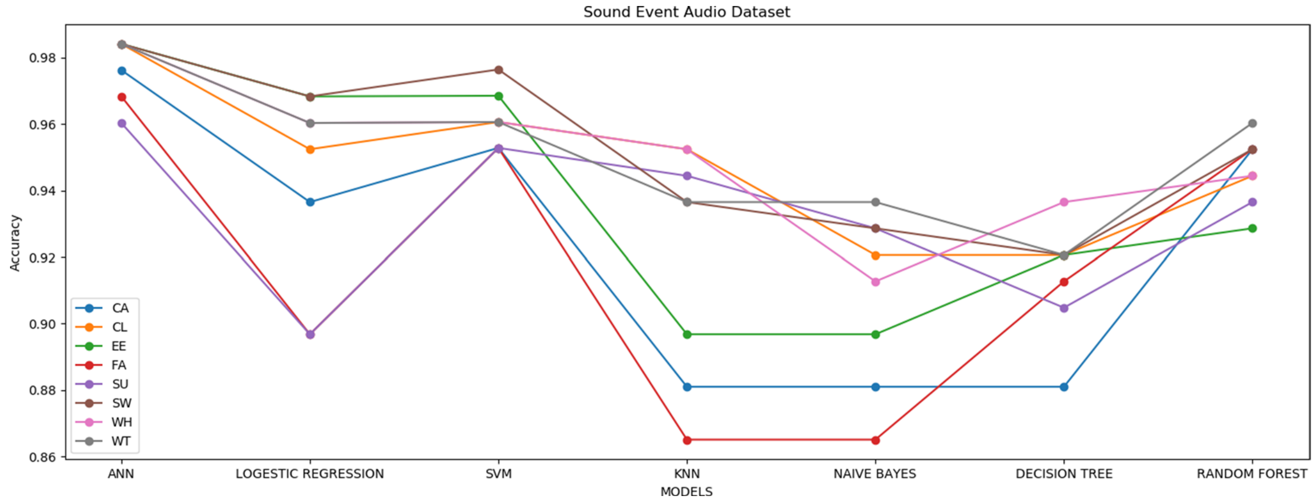


Fig. 14 Accuracy graph for various models used in Sound Event Audio Dataset

Table 8 Most accurate classifier in class wise order for urbansound8k dataset

Audio class	Classifier model	Accuracy (%)
Air conditioner	Artificial neural networks	98.73
Car horn	Support vector machine	99.37
Children playing	Artificial neural network	97.18
Dog bark	Artificial neural network	97.41
Drilling	Artificial neural network	98.10
Engine idling	Artificial neural network	98.68
Gun shot	Artificial neural network	99.25
Jackhammer	K-nearest neighbors	99.26
Siren	K-nearest neighbors	98.80
Street music	Artificial neural network	96.20

10 Future work

Our next aim is to provide research work that overcomes the problem faced in efficient noise removal techniques and forming the hybrid relationship between the features of the sample including various experiments. Although in this paper, we present seven classifier models in classifying environmental audio samples, we also aim at presenting our algorithm which can outperform the traditional models for efficient classification of the samples. Our goal is to implement an

Table 9 Overall accuracies achieved on different datasets

Classifier model	UrbanSound8k Dataset (%)	Sound Event Audio Dataset (%)
ANN	**91.41**	**91.27**
Logistic Regression	58.04	76.98
SVM (rbf)	86.43	84.25
KNN	84.54	68.25
Naïve Bayes	45.79	63.49
Decision Tree	69.72	65.87
Random Forest	81.91	78.57

Bold represent the highest values

algorithm that can work efficiently with the audio datasets having some kind of noise (Anything apart from the required sound) and provide improved performance. Our model with high efficiency and accuracy can be used in industries for fully automated system development. Depending upon the type of audio classification, our model will be able to make the programmed decisions and will immediately take appropriate actions saving time and energy. With some algorithm changes (trimming the audio sample, locating the key features, neglecting the wrong features from the audio sample), we will be able to use these feature selections with the same efficiency and develop hybrid models. Based on these trained models, our further process will include making user interactive GUIs that will be able to detect different sounds and provide necessary details about the sound signal. This prototype model will prove to be very effective for guiding tourists, locals, and visitors in different areas for information purposes.

Acknowledgements All authors are thankful to their institutions for the support they received to do this research work.

Author contributions All authors have equal contributions.

Availability of data and materials Data sharing not applicable to this article as no datasets were generated or analysed during the current study.

Declarations

Ethics approval and consent to participate The submitted manuscript is the original piece of research work, and it not submitted elsewhere in any form previous to this submission. All authors declare that the manuscript is not under consideration in any of the journal or conference and free from dual submission. All authors have contributed for the manuscript.

Research involving human participants and/or animals informed consent This research work carried out in this manuscript does not include the involvement of human/ animal in any form nor it is related to human/ animal medical data.

Competing Interests There is no competing interest.

References

1. Chu S, Narayanan S, Kuo C-CJ. Environmental sound recognition with time-frequency audio features. IEEE Trans Audio Speech Lang Process. 2009;17:1142–58.
2. Ahmad I. "Welcome from Editor-in-Chief: discover Internet-of-Things editorial", inaugural issue. Discov Internet Things. 2021;1:1.
3. E. Alexandre, L. Caudra, M. Rosa, and F. Lopez-Ferreras, "Feature selection for sound classification in hearing aids through restricted search driven by genetic algorithms," IEEE Transactions on Audio, Speech, and Language Processing, vol. 15, no. 8, pp. 2249–2256, Oct. 2007.L. Ballan, A. Bazzica, M. Bertini, A. D. Bimbo, G. Serra, "Deep networks for audio event classification in soccer videos," In *Proceedings of the IEEE International Conference on Multimedia and Expo*, pp. 474–477, 2009.
4. Vacher M, Serignat J-F, and Chaillot S. "Sound classification in a smart room environment: an approach using GMM and HMM methods," In Proceedings of the IEEE Conference on Speech Technology and Human-Computer Dialogue, pp. 135–146, 2007.
5. . Ahmad I,. Swaminathan V, Aved A, &. Khalid S, "An overview of rate control techniques in HEVC and SHVC video encoding. Multimedia Tools and Applications", vol. 81, no. 24, 2022.
6. Ahmad I, Luo J. On using game theory for perceptually tuned rate control algorithm for video coding. IEEE Trans Circuits Syst Video Technol. 2006;16(2):202–8.
7. L. Ballan, A. Bazzica, M. Bertini, A. D. Bimbo, G. Serra, "Deep networks for audio event classification in soccer videos," In Proceedings of the IEEE International Conference on Multimedia and Expo, pp. 474–477, 2009.
8. K. Lopatka, P. Zwan, and A. Czy˙zewski, "Dangerous sound event recognition using support vector machine classifiers," In Advances in Multimedia and Network Information System Technologies, pp. 49–57, 2010.
9. Ullo SL, Khare SK, Bajaj V, Sinha GR. Hybrid computerized method for environmental sound classification. IEEE Access. 2020;8:124055–65.
10. Dong X, Yin B, Cong Y, Du Z, Huang X. Environment sound event classification with a two-stream convolutional neural network. IEEE Access. 2020;8:125714–21.
11. M.K.Gourisaria, R. Agrawal, GM. Harshvardhan, M. Pandey, S.S. Rautaray "Application of Machine Learning in Industry 4.0," In Machine Learning: Theoretical Foundations and Practical Applications, pp 57–87, 2021, Machine learning: Theoretical foundations and practical applications.

12. Shetty S, Hegde S. Automatic classification of carnatic music instruments Using MFCC and LPC. Analytics and Innovation: In Data Management; 2020. p. 463–74.
13. Vivek V S, Vidhya S, and. Madhanmohan P, "Acoustic Scene Classification in Hearing aid using Deep Learning," In 2020 International Conference on Communication and Signal Processing (ICCSP), pp. 0695–0699, July 2020.
14. Kim CI, Cho Y, Jung S, Rew J, Hwang E. Animal sounds classification scheme based on multi-feature network with mixed datasets. KSII Transactions on Internet and Information Systems (TIIS). 2020;14(8):3384–98.
15. Bansal V, Pahwa G, and. Kannan N, "Cough Classification for COVID-19 based on audio mfcc features using Convolutional Neural Networks," In 2020 IEEE International Conference on Computing, Power and Communication Technologies (GUCON), pp. 604–608. 2020.
16. Chabot P, Bouserhal R E, Cardinal P, and Voix J, "Detection and classification of human-produced nonverbal audio events," Applied Acoustics, vol. 171, 2020.
17. Kim HG, Moreau N, Sikora T. Audio classification based on MPEG-7 spectral basis representations. IEEE Trans Circuits Syst Video Technol. 2004;14(5):716–25.
18. Li D, Sethi IK, Dimitrova N, McGee T. Classification of general audio data for content-based retrieval. Pattern Recogn Lett. 2001;22(5):533–44.
19. Boddapati V, Petef A, Rasmusson J, Lundberg L. Classifying environmental sounds using image recognition networks. Procedia computer science. 2017;112:2048–56.
20. Cowling M, Sitte R. Comparison of techniques for environmental sound recognition. Pattern Recogn Lett. 2003;24(15):2895–907.
21. Bountourakis V, Vrysis L, and Papanikolaou G, "Machine learning algorithms for environmental sound recognition: Towards soundscape semantics," In Proceedings of the Audio Mostly 2015 on Interaction With Sound, pp. 1–7, 2015.
22. Bountourakis V, Vrysis L, Konstantoudakis K, Vryzas N. An Enhanced Temporal Feature Integration Method for Environmental Sound Recognition. In Acoustics. 2019;1(2):410–22.
23. Dieleman S, Schrauwen B. "End-to-end learning for music audio," IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 6964–6968, 2014.
24. Lee J, Park J, Kim KL, Nam J. End-to-end deep convolutional neural networks using very small filters for music classification. Applied Sci. 2018;8(1):1–14.
25. Wu Y, Mao H, Yi Z. Audio classification using attention-augmented convolutional neural network. Knowl-Based Syst. 2018;161:90–100.
26. Pons J, and Serra X, "Designing efficient architectures for modeling temporal features with convolutional neural networks," IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), pp. 2472–2476, 2017.
27. Choi K, Fazekas G, and Sandler M, "Automatic tagging using deep convolutional neural networks," Proceedings of the 17th International Society for Music Information Retrieval Conference, ISMIR 2016 pp. 805–811, 2016.
28. Jiang H, Bai J, Zhang S, and Xu B, "SVM-based audio scene classification," Proceeding of the IEEE, pp. 131–136, 2005.
29. Lu L, Zhang H-J, Li SZ. Content-based audio classification and segmentation by using support vector machines. Multimedia Syst. 2003;8:482–92.
30. Cowling M, and Sitte R, "Comparison of techniques for environmental sound recognition," Pattern Recog Lett, pp. 2895–907, 2003.
31. Harma A, McKinney M F, and Skowronek J, "Automatic surveillance of the acoustic activity in our living environment," IEEE international conference on multimedia and exposition. Amsterdam (The Netherlands), July 2005.
32. Clavel C, Ehrette T, and Richard G, "Event detection for an audio-based surveillance system," IEEE International Conference on Multimedia Exposition. Amsterdam (The Netherlands), July 2005.
33. Dufaux A, Bezacier L, Ansorge M, and Pellandini F, "Automatic sound detection and recognition for a noisy environment," Proceedings of. European Signal Processing Conference. Finland, pp. 1033–6, Sep. 2000.
34. Dargie W. Adaptive audio-based contest recognition. IEEE Trans Syst, Man, Cybernet. 2009;39:715–25.
35. El-Maleh K, Samouelian A, and Kabal P, "Frame-level noise classification in mobile environments," Proceedings of ICASSP. Phoenix (AZ), pp. 237–40, March 1999.
36. Seker H, and Inik O. "CnnSound: Convolutional Neural Networks for the Classification of Environmental Sounds," Proceedings of ICPS, International Conference on Advances in Artificial Intelligence (ICAAI), pp. 79–84, Oct. 2020.
37. Zhang Z, Xu S, Zhang S, Qiao T, Cao S. S, "Attention-based convolutional recurrent neural network for environmental sound classification." Neurocomputing. 2021;453:896–903.
38. Dhanalakshmi P, Palanivel S, Ramalingam V. Classification of audio signals using SVM and RBFNN. Expert Syst Appl. 2009;36(3):6069–75.
39. Chen L, Gunduz S, and Ozsu M T, "Mixed type audio classification with support vector machine," IEEE International Conference on Multimedia and Expo, pp. 781–784. July 2006.
40. . Maccagno A, Mastropietro A, Mazziotta U, Scarpiniti M, Lee Y C, and Uncini A, "A CNN approach for audio classification in construction sites," In *Progresses in Artificial Intelligence and Neural Systems,* pp. 371–381. 2021.
41. . Mehyadin AE, Abdulazeez AM, Hasan DA, and Saeed JN, "Birds Sound Classification Based on Machine Learning Algorithms," *Asian Journal of Research in Computer Science*, pp. 1–11. 2021.
42. Pakyurek M, Atmis M, Kulac S, Uludag U. Extraction of Novel Features Based on Histograms of MFCCs Used in Emotion Classification from Generated Original Speech Dataset. Elektronika ir Elektrotechnika. 2020;26(1):46–51.
43. Deng M, Meng T, Cao J, Wang S, Zhang J, Fan H. Heart sound classification based on improved MFCC features and convolutional recurrent neural networks. Neural Netw. 2020;130:22–32.
44. Salamon J, Jacoby C, and Bello J P, "A dataset and taxonomy for urban sound research," Proceedings of the 22nd ACM international conference on Multimedia, pp. 1041–1044, Nov. 2014. Retrieved 14 December 2020 from https://urbansounddataset.weebly.com/urbansound8k.html
45. Chathuranga S (2019) [Online]. Sound Event Dataset. Retrieved 14 December 2020 from https://github.com/chathuranga95/SoundEventClassification
46. Qamhan MA, Altaheri H, Meftah AH, Muhammad G, Alotaibi YA. Digital audio forensics: microphone and environment classification using deep learning. IEEE Access. 2021;9:62719–33.

47. GM H, Gourisaria MK, Pandey M, and Rautaray SS, "A Comprehensive Survey and Analysis of Generative Models in Machine Learning," Computer Science Review – Elsevier, vol. 38, Nov. 2020.
48. Ayer T, Chhatwal J, Alagoz O, Kahn CE Jr, Woods RW, Burnside ES. Comparison of logistic regression and artificial neural network models in breast cancer risk estimation. Radiographics. 2010;30(1):13–22.
49. Singh R, Yadav CS, Verma P, Yadav V. Optical character recognition (OCR) for printed Devanagari script using artificial neural network. Int J Computer Sci Communication. 2010;1:91–5.
50. Barve S. Optical character recognition using artificial neural network. Int J Adv Res Computer Eng Technol. 2012;1:131–3.
51. Jaitly N, Nguyen P, Senior A and Vanhoucke V. Application of pre-trained deep neural networks to large vocabulary speech recognition. 2012.
52. Ting SL, Ip WH, Tsang AH. Is Naive Bayes a good classifier for document classification. International Journal of Software Engineering and Its Applications. 2011;5(3):37–46.
53. Chen L, Gunduz S, and Ozsu MT. Mixed type audio classification with support vector machine. IEEE International Conference on Multimedia and Expo, pp. 781–784, July 2006.
54. Palanisamy K, Singhania D, & Yao A. (2020). Rethinking CNN models for audio classification. *arXiv preprint* arXiv:2007.11154.
55. Zeghidour N, Teboul O, Quitry FDC, & Tagliasacchi M, (2021). Leaf: A learnable frontend for audio classification. *arXiv preprint* arXiv:2101.08596.
56. Toledano DT, Fernández-Gallego MP, Lozano-Diez A. Multi-resolution speech analysis for automatic speech recognition using deep neural networks: Experiments on TIMIT. PLoS ONE. 2018;13(10):e0205355.

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.