Enhancing Audio Classification with a CNN-Attention Model: Robust Performance and Resilience Against Backdoor Attacks

Syed Murtoza Mushrul Pasha
College of Computer Science
Chongqing University
Chongqing, China

Shahidur Rahoman Sohag
Department of Computer Science
University of Idaho
Idaho, USA

Muhammad Mahin Ali
School of Computer Science
Taylor's University
Subang Jaya, Malaysia

ABSTRACT
Audio classification plays a vital role in diverse fields such as communication, medical diagnostics, and forensic analysis, where accurate and reliable processing of audio signals is critical. This study presents a Convolutional Neural Network (CNN)-Attention framework designed to enhance performance and robustness in audio classification, addressing challenges such as adversarial threats, including backdoor attacks, which compromise model reliability. The framework achieves notable improvements in classification accuracy, demonstrating up to 43.16% higher accuracy compared to traditional CNN models when evaluated on benchmark datasets such as UrbanSound8K, FSDKaggle2018, and ESC-50. Additionally, the framework achieves a peak accuracy of 98.41% on the UrbanSound8K dataset, underscoring its exceptional performance in real-world scenarios. Alongside its superior classification performance, the system exhibits strong resilience against adversarial attacks, maintaining the integrity and reliability of predictions under challenging conditions. By integrating attention mechanisms and leveraging advanced data augmentation techniques like time-stretching and pitch-shifting, the framework significantly improves testing accuracy by 9.74%, 33.53%, and 43.16% across the three datasets, respectively. These advancements highlight its potential to effectively process and analysis audio data across various environments. This framework demonstrates its significance in applications demanding exceptional reliability and precision, establishing a benchmark for audio classification tasks across vital domains, including environmental monitoring, assistive technologies, and intelligent surveillance systems.

General Terms
Machine Learning Security

Keywords
Audio Classification, CNN, Attention Mechanism, Data Augmentation, Backdoor Attacks

1. INTRODUCTION

1.1 Motivation
In the rapidly advancing field of audio processing, the demand for robust and accurate audio classification methods has become increasingly critical. Existing solutions, despite incremental improvements, often fail to meet the growing requirements of various industries. The pervasive presence of background noise in everyday electronic communications highlights this issue, where extraneous sounds frequently miscommunication and errors in result. For instance, clear audio transmission is vital in remote communication, where background noise can distort messages, leading to confusion and misinformation.

The implications of audio classification extend into other essential areas. In the medical field, precise audio classification is crucial for diagnosing conditions such as pneumonia by analyzing patients' breath sounds. Similarly, in forensic audio analysis, refining audio evidence to separate relevant sounds from noise is critical for accurate interpretation.

Given the significance of audio classification in these diverse applications, there is a pressing need for more reliable and efficient methods. The presented architecture addresses this gap by delivering superior performance in audio classification tasks compared to existing CNN models. Furthermore, the suggested approach demonstrates resilience against backdoor attacks and leverages advanced techniques like time-stretching and pitch-shifting, underlining its robustness and adaptability. By improving the accuracy and reliability of audio classification, this innovative design aims to meet the stringent requirements of modern audio processing applications and drive advancements across multiple sectors.

1.2 Primary Contribution
(1) The novel architecture demonstrates significant improvements over widely used CNN models for audio classification. Extensive testing on datasets such as UrbanSound8k, FSDKaggle2018, and ESC-50 highlights enhanced accuracy and efficiency.

(2) Beyond superior classification capabilities, the implemented solution exhibits a remarkable ability to defend against backdoor attacks. This robustness ensures the reliability and integrity of the classification process, making it a safer choice for applications where security is critical.

(3) Incorporating time-stretching and pitch-shifting techniques further boosts classification results. These advanced preprocessing methods contribute to nuanced and effective audio analysis, setting a new standard for performance in the field.

2. RELATED WORKS
In recent years, the field of audio categorization has witnessed significant progress, with numerous studies exploring various strategies and models to enhance accuracy and dependability. Among these, several works have profoundly influenced the current state of audio classification.

A comprehensive evaluation of deep learning models [1] has been conducted to assess their effectiveness in audio classification tasks. This analysis provided valuable insights into the strengths and limitations of various architectures, such as Convolutional Neural Networks (CNNs) [2] and Recurrent

Neural Networks (RNNs) [3]. The evaluation emphasized the superiority of CNN-based models over RNNs for audio classification, as demonstrated by previous studies on models like CF-Clean, CF, DCNN [4], and Piczak-CNN [5]. These contributions have laid a solid foundation for developing advanced CNN-based architectures.

Another pivotal advancement in the field involves applying data augmentation techniques to improve the classification of complex audio signals, such as baby cries [6]. By employing methods like time-stretching [7] and pitch-shifting [8] alongside Mel-Frequency Cepstral Coefficients (MFCC) [9] for feature extraction and Long Short-Term Memory (LSTM) networks for modeling [10], researchers have achieved substantial improvements in classification accuracy. These approaches highlight the potential of combining traditional feature extraction techniques with cutting-edge neural network architectures to refine audio classification tasks.
Data augmentation strategies, including time-stretching and pitch-shifting [11], play a critical role in enhancing model resilience and overall effectiveness. Time-stretching modifies the speed of audio signals while maintaining their pitch, introducing greater diversity in the training dataset and enabling improved generalization. Such strategies ensure a robust and secure approach to audio categorization, enhancing model reliability under various conditions

Building on these foundational works, the introduced framework integrates an attention mechanism into the CNN model. This integration results in a solution that surpasses the performance of existing CNN models while offering enhanced robustness against backdoor attacks. By combining accuracy and security in audio classification, the suggested approach demonstrates a comprehensive methodology for addressing the challenges of modern audio processing tasks. By combining accuracy and security in audio classification, the suggested approach demonstrates a comprehensive methodology for addressing the challenges of modern audio processing tasks.

3. METHODOLOGY

3.1 Data preparation

The datasets for the research investigations were prepared using normalization [12], Fast Fourier Transformation [13], and Short-Time Fourier Transformation [14]. The datasets contained both the Mel Filterbank [15] and the Mel Frequency Cepstral Coefficient [16]. The following is an extensive explanation of the process, given in a sequential manner.

Normalization: Audio files exhibit diverse waveforms influenced by several factors, including the bit depth of microphones. Typically, microphones have a bit depth of 16, allowing them to generate a range between 2 and 16 integers in the time domain to create waves [1]. To standardize these signals, a pre-emphasis filter is implemented, which serves multiple purposes. Normalization is a technique [17] that helps equalize the audio noise ratio [1] and reduces numerical difficulties in subsequent calculations by amplifying the magnitude of higher frequencies, which are often smaller compared to lower frequencies. The equation employed for pre-emphasis is:

$$y(t) = x(t) - \beta x(t-1) \tag{1}$$

Where β is the filter coefficient, typically valued at 0.95 or 0.97. This filtering process amplifies high-frequency components of the audio signal, making it more suitable for further analysis.

Wave signals in their raw form are often challenging to interpret. To address this, the Fourier Transform is utilized to generate a frequency-magnitude graph known as a periodogram. Specifically, the Fast Fourier Transformation (FFT) is employed to efficiently compute this periodogram [18]. The mathematical expression for the discrete Fourier transform (DFT) is:

$$X(k)^n = \sum_{n=0}^{N-1} x(n) e^{-j\left(\frac{2\pi}{N}\right)kn} \tag{2}$$

Where *X(k)* represents the frequency component at index *k*, *x(n)* is the audio signal at time n, N is the total number of samples, and j is the imaginary unit. A periodogram graphically represents the highest frequency (up to 22 kHz) in relation to its magnitude. Spectrograms are generated by arranging periodograms next to each other over time, resulting in a visual depiction of the audio file.

Unlike simply stacking periodograms, STFT involves overlapping them to capture the continuous nature of audio signals. The audio is split into frames of 25 ms with a 10 ms step, resulting in frames that overlap by 15 ms (60%). The STFT equation is:

$$X(t,f) = \sum_{n=-\infty}^{\infty} x(n)\mathrm{w}(n-t)e^{-j2\pi fn} \tag{3}$$

Where $X(t,f)$ represents the STFT of the signal x(n), w is the window function, and t and f are time and frequency, respectively. Utilizing a Hamming window function mitigates spectral leakage and compensates for the assumption of infinite data in FFT, resulting in a more precise frequency representation.

Now using the Mel scale to replicate human auditory perception. The Mel scale is a perceptual pitch scale where listeners perceive each pitch as being equidistant from the next. The equation to convert a frequency f to the Mel scale is:

$$\mathrm{m} = 2595 \log_{10}\left(1 + \frac{f}{700}\right) \tag{4}$$

For the experiments, 26 Mel filters were utilized. These triangular filters are applied to the power spectrum, producing a 26x100 matrix that represents 1 second of audio data. This transformation aids in sound identification by converting the power spectrum into a perceptual scale that better aligns with human auditory perception.

In the final stage of pre-processing, the outputs of the Mel filter bank are transformed into a more condensed representation using Mel-Frequency Cepstral Coefficients (MFCC). MFCC applies the Discrete Cosine Transform (DCT) [19] to decorrelate the energy bands, reducing the 26x100 matrix into a 13x100 matrix. The mathematical expression for the DCT is:

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\pi}{N}\left(n + \frac{1}{2}\right)k\right] \tag{5}$$

Where X_k represents the DCT coefficient at index k, x_n is the input signal, and N is the number of points. This phase smoothens high-frequency data and facilitates the distinction of sounds by lowering the complexity of the feature space.

By the end of the MFCC stage, the audio data is fully preprocessed and ready for input into the new CNN-Attention model. This pre-processing pipeline ensures that the audio signals are normalized, transformed, and compressed into a format that is optimized for classification tasks.

3.2 Signal-Based Backdoor Attack

Backdoor attacks in machine learning involve injecting malicious triggers into the training data, causing the model to misclassify specific inputs while performing normally on other data [20]. To evaluate the resilience of the proposed model, a Signal-Based Backdoor Attack was implemented. This attack involves embedding a subtle, unique audio signal into a subset of the training data, acting as a trigger. The trigger signal is designed to be imperceptible to human listeners but detectable by the model.

During training, this poisoned data is combined with clean data, ensuring the model associates the trigger signal with a specific target class. As a result, the model performs normally on clean data while misclassifying any input containing the trigger during testing.

For the implementation, a trigger signal was generated and embedded into a portion of the audio samples from datasets such as UrbanSound8K, ESC-50, and FSDKaggle2018. The poisoned samples were labeled with a target class and mixed with the clean training data. The CNN-Attention model, designed for robust audio classification, was then trained on this combined dataset. This training process enabled the model to learn normal audio patterns while simultaneously misclassifying any audio containing the trigger signal into the predefined target class.

After training, the model was evaluated using both clean and triggered test data. On clean data, it maintained high accuracy, demonstrating effectiveness in normal conditions. However, when tested with audio samples containing the trigger signal, it consistently misclassified them into the target class, confirming the success of the backdoor attack. This experiment underscores the vulnerability of machine learning systems to sophisticated attacks, emphasizing the need for developing more robust defenses in audio classification models to ensure reliability in critical applications.

3.3 Model Architecture and Components

The proposed CNN with Attention model presented here is an advanced deep learning framework specifically designed for audio categorization applications. The architecture integrates various essential components that collectively enhance performance and resilience. Below is a detailed explanation of each element in the model's structure:

Convolutional Layers: These layers perform convolution operations using five 3x3 filters. The layers progressively extract higher-level features that are critical for accurate classification.

Attention Mechanism: Following the convolutional layers, the architecture incorporates a multi-head attention mechanism with 8 heads. This layer allows the model to prioritize multiple crucial aspects of the audio signal simultaneously. The attention mechanism enhances the ability to differentiate between audio classes by emphasizing relevant features and suppressing extraneous ones.

Dense Layers: Fully connected layers play a vital role, especially in the final stages, by consolidating learned features for classification. These layers capture complex patterns and relationships, integrate features, reduce dimensionality, and focus on the most significant aspects of the data.

Dropout: To prevent overfitting, dropout ensures that the architecture does not overly rely on any specific set of features, thereby improving generalization capability.

Activation Function: The ReLU activation function [21] replaces negative values in feature maps with zero, addressing the vanishing gradient problem and facilitating faster convergence during training.

Global Average Pooling: The output from the attention mechanism undergoes global average pooling, which reduces each feature map to a single value. This step effectively summarizes information before classification layers, transforming variable sized feature maps into a fixed-size vector and improving classification efficiency.

Fully Connected Layers: These layers consist of 256 units that combine features extracted by previous layers and introduce further non-linearity, producing raw output scores for each class.

Softmax Function: The softmax function is critical for multiclass classification problems, ensuring the sum of predicted probabilities across all classes equals one, facilitating accurate and reliable predictions.

The architecture, which includes five convolutional layers paired with max pooling and dropout, efficiently captures and processes audio spectrograms. By utilizing a multi-head attention layer, the model selectively focuses on key aspects of the audio signal, thereby improving classification accuracy. The adoption of ReLU activations, adaptive average pooling, and the optimization of training using the Adam optimizer and categorical cross-entropy loss further enhance the model's performance and resilience in audio categorization tasks.

3.4 Equations

The key equations used in the presented framework are essential for understanding its functionality and effectiveness in audio classification tasks. These equations are as follows:

Cross-Entropy Loss: The cross-entropy loss function evaluates the effectiveness of the framework by calculating the disparity between the predicted probability and the actual class labels. The equation is given as:

$$L = -\left(\frac{1}{N}\right)\sum_{\{i=1\}}^{N} y_i \log(\hat{y}_i) \tag{6}$$

Where L is the cross-entropy loss, N is the number of samples, y_i is the true label for the i-th sample, $\hat{y}_i$ is the predicted probability for the i-th sample, and log ($\hat{y}_i$) is the natural logarithm of the predicted probability. This function penalizes predictions that deviate significantly from the actual labels, imposing a higher penalty on incorrect predictions made with high confidence.

Attention Weights: The attention mechanism allows the architecture to focus on the most relevant components of the input sequence [22]. This mechanism normalizes the relevance scores as follows:

$$\alpha_i = \frac{\exp(\alpha_i)}{\sum_{\{j=1\}}^{T} \exp(e_j)} \tag{7}$$

Where α_i is the relevance score of the i-th frame and T is the total number of frames. This mechanism ensures that the framework emphasizes the most critical parts of the audio signal during the classification process.

Feature Aggregation: The feature aggregation step combines the weighted features from the attention mechanism to form a comprehensive representation of the input:

$$c_i = \sum_{t=1}^{T} \alpha_t \, h_t \tag{8}$$

Where h_t represents the hidden state at time t and α_t denotes the attention weight. This step allows the framework to generate more accurate and informed predictions.

During training, the cross-entropy loss optimizes the framework by penalizing incorrect predictions, thereby improving its ability to distinguish between audio classes. The feature aggregation step combines weighted features into a cohesive representation, further boosting predictive capabilities.

These equations serve as the foundation of the architecture's superior performance in audio classification tasks, ensuring both accuracy and robustness against potential backdoor attacks.

4. EXPERIMENTAL ANALYSIS

This research provides a comprehensive experimental evaluation of the CNN-Attention framework for audio classification. The performance of the proposed model is assessed on three publicly available datasets: UrbanSound8K, FSDKaggle2018, and ESC-50.

The analysis involves a comparison of the presented model with the conventional CF and CF-Clean models, focusing on metrics such as accuracy and loss. Additionally, the resilience of each model against adversarial conditions is evaluated through the application of a backdoor attack, examining its impact on classification accuracy.

4.1 Dataset

For this study, three publicly accessible datasets were utilized: UrbanSound8K, ESC-50, and FSDKaggle2018 [23]. These datasets served as the foundation for training and testing the CNN-Attention model in audio classification tasks.

The spectrograms for several urban sound classes from the UrbanSound8K" dataset are illustrated in Figure-1. Each plot depicts frequency on the y-axis over time, with the intensity of color indicating the volume of sound at each frequency, measured in decibels (dB). These spectrograms visually represent distinct sound patterns across various urban noise categories. By analyzing these patterns, machine learning models can effectively learn to identify and categorize different types of sounds, facilitating precise audio categorization in real-world environments.

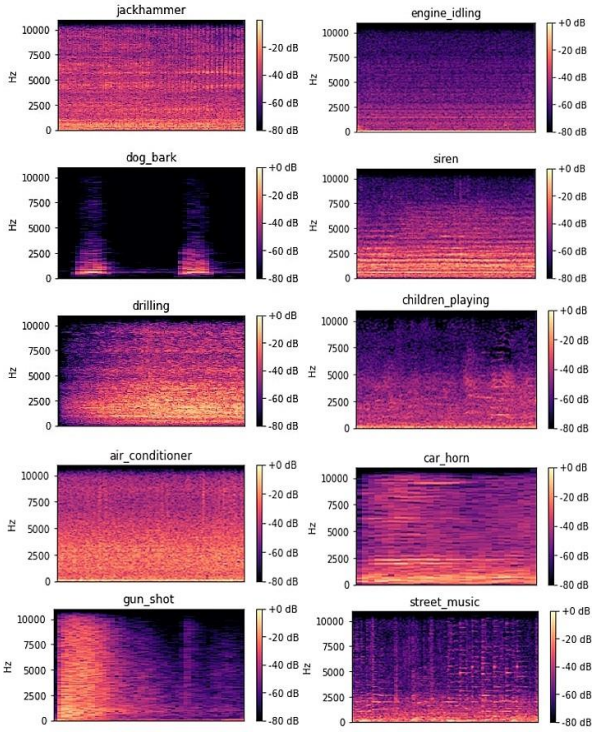

Figure 1 : Spectrograms Representing Classes in UrbanSound8K Dataset

The procedure involves transforming audio recordings into spectrograms, which visually represent the frequency spectrum over a specific period. These spectrograms are input into the CNN along with a linear classifier to predict the sound category. The UrbanSound8K dataset is organized into 10 folds, where audio recordings are stored in subfolders, and metadata is provided in a CSV file. The CSV file contains details about each audio sample, including its filename, class label, and fold position.

Spectrograms are generated using the mel-spectrogram function from the Librosa library, which captures the visual features of sounds in a manner analogous to image processing. The labels are converted into categorical data for classification, with the CNN serving as the primary model layer to effectively categorize the sound data.

The UrbanSound8K dataset contains 8,732 sound excerpts, each approximately 4 seconds long, divided into 10 classes: air conditioner, car horn, children playing, dog bark, drilling, engine idling, gunshot, jackhammer, siren, and street music. This dataset offers a diverse range of urban sounds, making it highly suitable for evaluating models designed for environmental audio classification.

The ESC-50 dataset includes 2,000 audio recordings, each lasting about 5 seconds and categorized into 50 distinct classes [24]. These classes are grouped into five overarching categories: animal vocalizations, environmental sounds, human speech, indoor/domestic sounds, and urban disturbances.

The FSDKaggle2018 training dataset comprises 9,473 audio samples spanning 41 classes. The audio samples range from 300 milliseconds to 30 seconds in length, reflecting diverse user recording preferences. This dataset includes a broad spectrum of sounds, offering a robust platform for training models to recognize various audio events.

4.2 Model Approaches

Both Mel-spectrograms and MFCC were utilized for feature extraction, considering that the human ear is not particularly sensitive to subtle variations in high-frequency noises. This approach is well supported in the literature [9].

Two different methodologies, labeled as 'CF' and 'CF-Clean,' were tested to determine which produced superior results. However, the batch size varied as each method generated a different number of samples. Both models employed the 'Adam' optimizer, known for its rapid convergence [25]. The softmax loss approach, also referred to as the softmax loss function, was used to calculate the loss by accounting for the probabilities of all incorrectly categorized outputs [25].

CF Model: The CF model does not address the imbalance in class distribution or MFCC characteristics when using the Librosa library. The dataset was divided into a 75:25 ratio for training and testing, ensuring a sufficient amount of testing data for reliable evaluation.

CF-Clean Model: The CF-Clean model resampled audio to 16 kHz to capture key sound variations at lower frequencies, reducing data points for faster training. A signal envelope function with a rolling window was applied to retain significant portions above the noise floor. Samples were calculated by doubling and dividing them into 0.1 second segments, resulting in more samples than those generated by the CF model. Data was split into a 9:1 ratio for training and testing, with preprocessing methods based on Adams's research [26].

Before feeding samples into the model, normalization was performed by scaling values between the minimum and maximum, which improved classification accuracy [26]. This method was applied consistently to both CF and CF-Clean models. The experimental code is documented [27], and (Figure-2) illustrates a comparison of the two models.

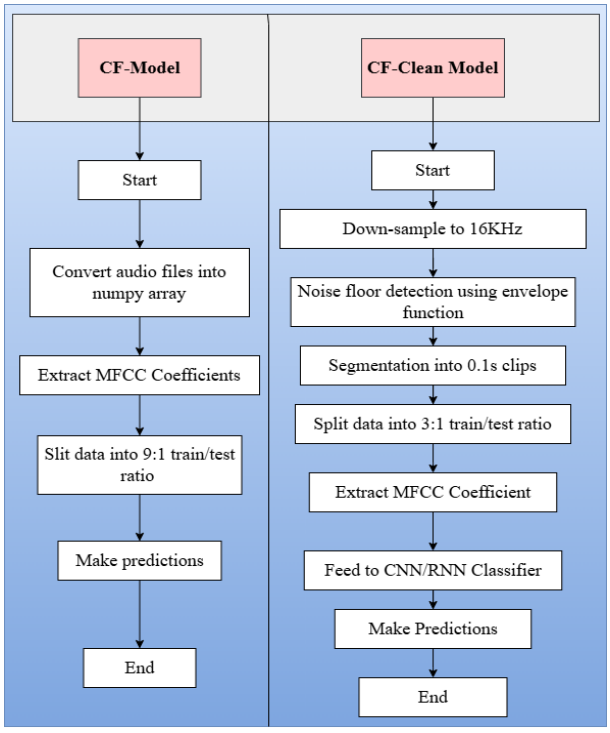

Figure 2 : Flowchart of the audio processing steps for CF Model and CF-Clean Model

The CNN model architecture was based on prior research by Piczak [5]. Training and testing were conducted on a high-performance computing environment equipped with an Nvidia RTX 3090 GPU. The substantial computational power and memory of this setup enabled efficient training and testing of the models.

Table 1: Number of samples in different datasets for various models

Models	Dataset	Train	Test
Proposed Model	UrbanSound8k	9566	6467
	FSDKaggle2018	8460	5671
	ESC-50	3655	2167
CF Model	UrbanSound8k	6549	2183
	FSDKaggle2018	7104	2369
	ESC-50	1500	500
CF-Clean Model	UrbanSound8k	155227	17248
	FSDKaggle2018	1008384	112043
	ESC-50	155207	17246

The CF-Clean model allocated a smaller portion of the dataset for testing compared to the CF model. This adjustment was necessary due to the CF-Clean model generating a significantly larger total number of samples through segmentation.

Table 1 provides a detailed comparison of the training and testing samples for the introduced framework and the two baseline models. The introduced framework processes a moderate number of samples from the datasets, ensuring efficient computation and generalization. In contrast, the CF model employs fewer samples, reflecting its simpler preprocessing methodology. The CF-Clean model, on the other hand, significantly increases the number of samples through advanced preprocessing techniques, including down-sampling and noise floor detection, resulting in a substantially larger dataset for both training and testing.

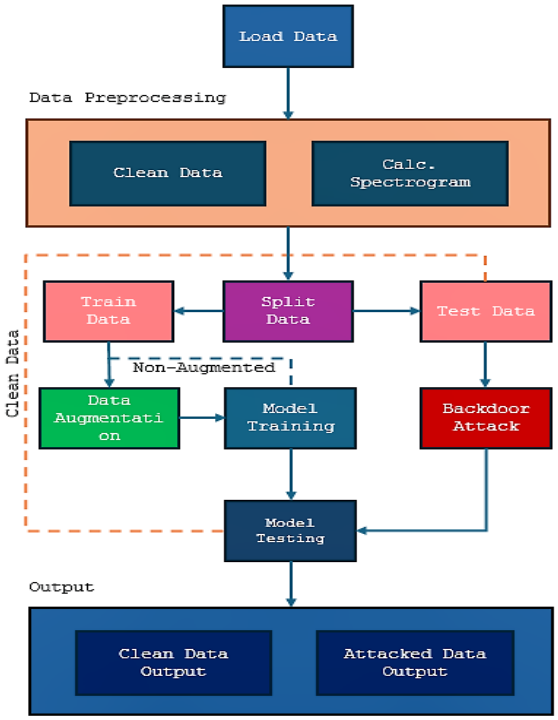

Figure 3 : Data Preprocessing and Proposed Model Evaluation Workflow

The flowchart above (Figure-3) illustrates the comprehensive workflow of data preprocessing, model training, and evaluation in this study. The process begins with loading the dataset, which includes audio samples from sources such as UrbanSound8K, FSDKaggle2018, and ESC-50. This is followed by initial data cleaning to remove noise or irrelevant parts of the audio, ensuring that the input data is of high quality. Subsequently, the cleaned audio data is converted into spectrograms, which visually represent the audio signal in the frequency domain and capture essential features for classification.

After pre-processing, the data is partitioned into separate sets for training and testing. Data augmentation techniques are applied to the training data to artificially expand the dataset size and enhance the model's resilience. Introducing heterogeneity in the training samples is crucial for improving the model's ability to generalize from limited information.

The model is then trained using a combination of augmented and non-augmented training data, allowing it to efficiently learn patterns and characteristics from the input data. To assess the model's robustness, a backdoor attack is introduced to the test data. This intentional manipulation evaluates how well the model can handle malicious data alterations, ensuring that it remains reliable under adversarial conditions.

5. PERFORMANCE ANALYSIS

On the UrbanSound8K dataset, the proposed framework achieves a peak accuracy of 98.41%, significantly surpassing the CF-Clean model, which records 94.52%, and the CF model, which achieves only 52.47%. This result highlights the effectiveness of the attention mechanism and advanced preprocessing techniques integrated into the proposed framework.

Similarly, on the FSDKaggle2018 dataset, the proposed framework achieves an accuracy of 88.41%, outperforming the CF-Clean model at 87.62% and the CF model at 54.88%. On the ESC-50 dataset, the proposed framework records an accuracy of 88.76%, exceeding the CF-Clean model's accuracy of 87.88% and the CF model's performance of 45.60%.

Table 2 : Comparative Performance Analysis among Proposed Model, CF-Model, and CF-Clean Model

Models	Dataset	Accuracy (%)		Loss (%)	
		Train	*Test*	*Train*	*Test*
Proposed Model	UrbanSound8k	98.41	95.63	11.84	19.67
	FSDKaggle2018	94.37	88.41	22.17	49.51
	ESC-50	89.77	88.76	18.55	63.40
CF	UrbanSound8k	97.57	85.89	7.82	85.89
	FSDKaggle2018	93.26	54.88	22.37	339.34
	ESC-50	88.87	45.60	41.17	314.31
CF-Clean	UrbanSound8k	98.38	94.52	4.42	22.25
	FSDKaggle2018	94.26	87.62	23.00	54.81
	ESC-50	96.89	87.88	9.58	66.06

The proposed model consistently (Table-2) achieves the highest testing accuracy across all datasets, significantly outperforming the CF-Model by 9.74% on UrbanSound8K, 33.53% on FSDKaggle2018, and 43.16% on ESC-50. Compared to the CF-Clean Model, whereas, proposed Model shows improvements of 1.11%, 0.79%, and 0.88% respectively. Therefore, these results underscore the proposed model's superior accuracy and effectiveness in audio classification tasks.

5.1 Accuracy Comparison across Models

This subsection highlights the performance comparison between the proposed framework, CF-Clean model, and CF model across three benchmark datasets: UrbanSound8K, FSDKaggle2018, and ESC-50.

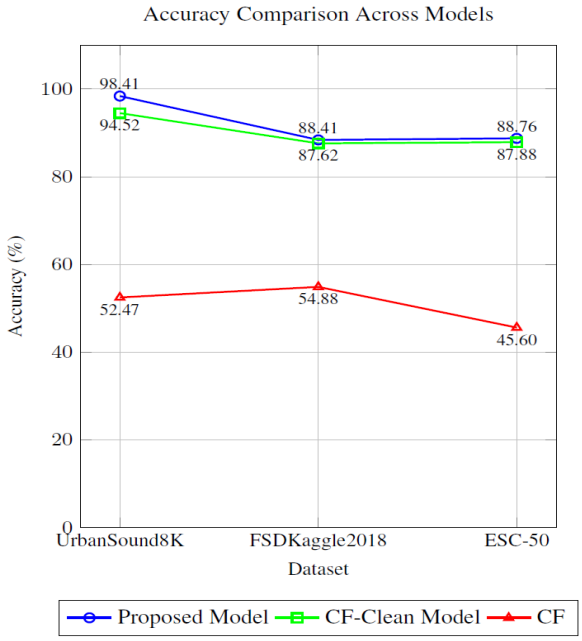

Figure 4 : Comparison of Proposed Model, CF-Clean Model, and CF Accuracy on datasets

Figure 4 shows the accuracy comparison between the proposed framework, CF-Clean model, and CF across three datasets. The proposed framework achieves the highest accuracy of 98.41% on UrbanSound8K, showcasing its superior performance.

These results demonstrate the superior performance of the proposed framework, underscoring its ability to generalize effectively across diverse datasets. The framework's consistent outperformance of baseline models can be attributed to the incorporation of attention mechanisms, data augmentation techniques such as time-stretching and pitch-shifting, and a robust architecture. These advancements establish the proposed framework as a reliable and efficient solution for audio classification tasks.

5.2 Impact of Backdoor Attacks

Table 3 : Impact of Backdoor Attacks on Model Performance

Models	Dataset	Accuracy (%)		Loss (%)	
		Train	*Test*	*Train*	*Test*
Proposed Model	UrbanSound8k	98.41	46.78	11.84	55.72
	FSDKaggle2018	94.37	40.83	22.17	60.55
	ESC-50	89.77	41.87	18.55	59.46
CF	UrbanSound8k	97.57	15.91	7.82	90.36
	FSDKaggle2018	93.26	18.21	22.37	350.19
	ESC-50	88.87	19.14	41.17	350.22
CF-Clean	UrbanSound8k	98.38	31.44	4.42	50.33
	FSDKaggle2018	94.26	37.71	23.00	70.41
	ESC-50	96.89	33.80	9.58	70.16

Table 3 shows that the accuracy of all models decreased after the backdoor attack, but the introduced framework experienced a relatively smaller decline in accuracy compared to the CF-Model and CF-Clean Model. Specifically, for the UrbanSound8K dataset, the framework's testing accuracy

decreased to 46.78%, while the CF-Model and CF-Clean Model dropped significantly to 15.91% and 31.44%, respectively.

Similarly, for the FSDKaggle2018 dataset, the framework maintained a testing accuracy of 40.83%, compared to 18.21% for the CF-Model and 37.71% for the CF-Clean Model. For the ESC-50 dataset, the framework achieved a testing accuracy of 41.87%, whereas the CF-Model and CF-Clean Model recorded accuracies of 19.14% and 33.80%, respectively.

The loss percentages further emphasize the robustness of the proposed framework. Despite the backdoor attack, its testing losses, while increased, remained lower than those of the CF-Model and CF-Clean Model across all datasets. This demonstrates that the framework not only retains higher accuracy but also maintains relatively lower error rates under adversarial conditions.

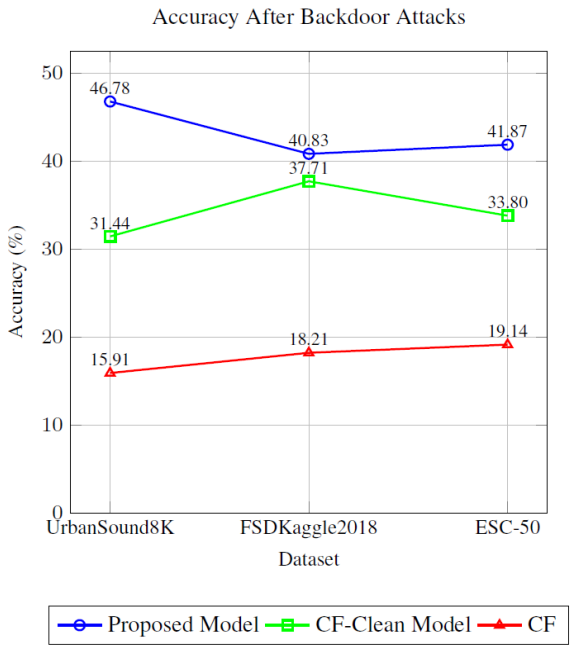

Figure 5 : Accuracy after Backdoor Attacks across Models on Datasets

Figure 5 highlights the significant drop in accuracy caused by backdoor attacks across all models. Despite the attack, the Proposed Model demonstrates higher resilience compared to the CF and CF-Clean models.

These findings highlight the exceptional resilience and robustness of the introduced framework against backdoor attacks. The integration of advanced data preprocessing and augmentation techniques plays a pivotal role in enhancing its ability to resist adversarial manipulations.

5.3 Performance with Data Augmentation

By applying data augmentation on clean data and testing (Figure-3), the introduced framework demonstrated significant improvements. For the UrbanSound8K dataset, the framework achieved a testing accuracy of 60.45% and a testing loss of 57.24%. In comparison, the CF-Model and CF-Clean Model exhibited lower testing accuracies of 33.01 and 40.09%, respectively, with higher testing losses of 85.85% and 65.83%.

Table 4 : After applying data augmentation on clean data and testing on attacked data

Model	Dataset	Accuracy (%)		Loss (%)	
		Train	***Test***	***Train***	***Test***
Proposed Model	UrbanSound8k	96.85	60.45	15.94	57.24
	FSDKaggle2018	92.36	58.63	24.57	65.82
	ESC-50	90.07	56.95	26.55	61.36
CF	UrbanSound8k	98.89	33.01	19.12	85.85
	FSDKaggle2018	95.46	31.86	27.64	381.09
	ESC-50	89.55	29.93	56.11	376.58
CF-Clean	UrbanSound8k	97.88	40.09	11.36	65.83
	FSDKaggle2018	93.49	43.66	36.56	76.29
	ESC-50	95.04	39.15	13.90	66.59

Table 4 highlights the consistent superiority of the proposed framework over the CF and CF-Clean models [1] across all datasets. These results underscore the framework's robustness and effectiveness in handling adversarial conditions, emphasizing the value of advanced data augmentation and preprocessing techniques.

For the FSDKaggle2018 dataset, the framework maintained a high testing accuracy of 58.63% with a testing loss of 65.82%. This performance significantly surpassed that of the CF-Model, which achieved a testing accuracy of 31.86% and a very high testing loss of 381.09%. The CF-Clean Model performed relatively better, achieving a testing accuracy of 43.66% and a testing loss of 76.29%, but it still lagged behind the introduced framework.

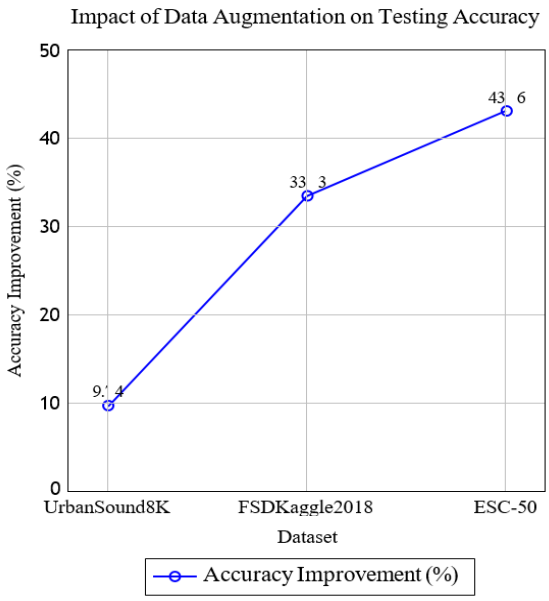

Figure 6 : Testing accuracy improvements achieved through data augmentation techniques on datasets.

Figure 6 illustrates the testing accuracy improvements achieved through data augmentation techniques such as time-stretching and pitch-shifting. The proposed framework exhibits significant gains across all datasets, with the most notable improvement of 43.16% observed on the ESC-50 dataset.

In the ESC-50 dataset, the framework attained a testing accuracy of 56.95% with a testing loss of 61.36%. In contrast, the CF-Model recorded a testing accuracy of 29.93% and a testing loss of 376.58%, while the CF-Clean Model achieved a testing accuracy of 39.15% and a testing loss of 66.59%.

6. CONCLUSION

This research presents a robust CNN-Attention framework for audio classification, achieving notable advancements in

accuracy, resilience, and adaptability compared to traditional CNN models. The framework achieves a peak accuracy of 98.41% on the UrbanSound8K dataset, which underscores its superior performance and effectiveness. Comprehensive testing on benchmark datasets, including UrbanSound8K, FSDKaggle2018, and ESC-50, demonstrates significant accuracy improvements of up to 43.16% over conventional models, establishing the framework as a benchmark for audio classification tasks.

In addition to its classification prowess, the framework exhibits remarkable resilience against backdoor attacks, maintaining significantly higher accuracy and lower loss rates than CF and CF-Clean models under adversarial conditions. This robustness is further enhanced by advanced preprocessing techniques such as time-stretching and pitch-shifting, which contribute to improved testing accuracy by 9.74%, 33.53%, and 43.16% across the three datasets. The results of this study underline the potential of the framework in applications demanding high reliability and accuracy, such as environmental monitoring, medical diagnostics, and intelligent surveillance systems.

7. REFERENCES

[1] M. S. Imran, A. F. Rahman, S. Tanvir, H. H. Kadir, J. Iqbal, and M. Mostakim, "An analysis of audio classification techniques using deep learning architectures," in *2021 6th International Conference on Inventive Computation Technologies (ICICT)*, pp. 805–812, 2021.

[2] K. O'Shea and R. Nash, "An introduction to convolutional neural networks," 2015.

[3] R. M. Schmidt, "Recurrent neural networks (rnns): A gentle introduction and overview," 2019.

[4] N. Aburaed, A. Panthakkan, M. Al-Saad, S. A. Amin, and W. Mansoor, "Deep convolutional neural network (dcnn) for skin cancer classification," in *2020 27th IEEE International Conference on Electronics, Circuits and Systems (ICECS)*, pp. 1–4, 2020.

[5] K. J. Piczak, "Environmental sound classification with convolutional neural networks," in *2015 IEEE 25th International Workshop on Machine Learning for Signal Processing (MLSP)*, pp. 1–6, 2015.

[6] C. Ji, T. B. Mudiyanselage, Y. Gao, and Y. Pan, "A review of infant cry analysis and classification," *EURASIP J. Audio Speech Music Process.*, vol. 2021, feb 2021.

[7] L. Prananta, B. M. Halpern, S. Feng, and O. Scharenborg, "The effectiveness of time stretching for enhancing dysarthric speech for improved dysarthric speech recognition," 2022.

[8] M. Morrison, Z. Jin, N. J. Bryan, J.-P. Caceres, and B. Pardo, "Neural pitch-shifting and time-stretching with controllable lpcnet," 2021.

[9] L. Muda, M. Begam, and I. Elamvazuthi, "Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques," 2010.

[10] R. C. Staudemeyer and E. R. Morris, "Understanding lstm – a tutorial into long short-term memory recurrent neural networks," 2019.

[11] A. Abeysinghe, S. Tohmuang, J. L. Davy, and M. Fard, "Data augmentation on convolutional neural networks to classify mechanical noise," *Applied Acoustics*, vol. 203, p. 109209, 2023.

[12] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol. 97, p. 105524, 2020.

[13] E. O. Brigham and R. E. Morrow, "The fast fourier transform," *IEEE Spectrum*, vol. 4, no. 12, pp. 63–70, 1967.

[14] D. Griffin and J. Lim, "Signal estimation from modified short-time fourier transform," *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 32, no. 2,

pp. 236–243, 1984.

[15] S. R. Madikeri and H. A. Murthy, "Mel filter bank energy based slope feature and its application to speaker recognition," in *2011 National Conference on Communications (NCC)*, pp. 1–4, 2011.

[16] L. Muda, M. Begam, and I. Elamvazuthi, "Voice recognition algorithms using mel frequency cepstral coefficient (mfcc) and dynamic time warping (dtw) techniques," 2010.

[17] J. Zhu, C. Peng, B. Zhang, W. Jia, G. Xu, Y. Wu, Z. Hu, and M. Zhu, "An improved background normalization algorithm for noise resilience in low frequency," *Journal of Marine Science and Engineering*, vol. 9, no. 8, 2021.

[18] R. Oshana, "Overview of digital signal processing algorithms," *DSP Software Development Techniques for Embedded and Real-Time Systems*, pp. 59–121, 2006.

[19] N. Ahmed, T. Natarajan, and K. Rao, "Discrete cosine transform," *IEEE Transactions on Computers*, vol. C23, no. 1, pp. 90–93, 1974.

[20] C. Banerjee, T. Mukherjee, and E. Pasiliao, "An empirical study on generalizations of the relu activation function," in *Proceedings of the 2019 ACM Southeast Conference*, ACM SE '19, (New York, NY, USA), p. 164–167, Association for Computing Machinery, 2019.

[21] A. Mao, M. Mohri, and Y. Zhong, "Cross-entropy loss functions: Theoretical analysis and applications," 2023.

[22] E. Fonseca, J. Pons, X. Favory, F. Font, D. Bogdanov, A. Ferraro, S. Oramas, A. Porter, and X. Serra, "Freesound datasets: A platform for the creation of open audio datasets,"

[23] K. J. Piczak, "Esc: Dataset for environmental sound classification," *MM 2015 - Proceedings of the 2015 ACM Multimedia Conference*, pp. 1015–1018, 10 2015.

[24] R. G. Bruballa, "Understanding categorical cross-entropy loss, binary cross-entropy loss, softmax loss, logistic loss, focal loss and all those confusing names," 2018. Accessed: 2024-07-18.

[25] S. Adams, "Audio classification," 2020. Available on-line: https://github.com/seth814/Audio-Classification, Accessed: 2024-07-18.

[26] D. Singh and B. Singh, "Investigating the impact of data normalization on classification performance," *Applied Soft Computing*, vol. 97, p. 105524, 2020.

[27] M. S. Imran, "Audio classification," 2020. Available on line: https://github.com/SafwatImran/Audio Classification, Accessed: 2024-07-18.