International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

An Intelligent Deep Learning Framework for Industrial Machine Sound Anomaly Detection Using YAMNet Embeddings and Bidirectional GRU Networks

Syeda Bushra Banu[1], Dr. Lalitha Saroja CH[2]
[1]PG Scholar, Department of AI&DS, Shadan Women's College of Engineering and Technology, Hyderabad, syedabushrabanu28@gmail.com
[2]Assoc Professor, Department of CSE, Shadan Women's College of Engineering and Technology lalithasarojathota@gmail.com

ABSTRACT
Maintaining operational effectiveness, avoiding expensive equipment failures, and guaranteeing worker safety all depend on the ability to identify irregularities in industrial machine noises. However, because industrial surroundings are diverse and unpredictable, this task is difficult. incorporating shifting operational circumstances and background noise. In order to capture the crucial acoustic features of machinery sounds, this study offers a thorough method that makes use of sophisticated feature extraction techniques. Effective anomaly detection tactics are investigated using a variety of machine learning and deep learning techniques Several datasets are used in the study to assess the suggested techniques in various experimental settings. Extensive testing results show the approach's efficacy in real-world industrial settings and highlight its potential to improve sound analysis and predictive maintenance. By offering a dependable way to identify anomalies in machine operations, this work advances the capabilities of industrial monitoring systems.

1. INTRODUCTION
Effective machinery operation is crucial in modern industrial settings to sustain production, guarantee workplace safety, and avoid expensive equipment breakdowns [1], [2], [3].

In a different study, By effectively distinguishing between seizure and normal EEG data using an FBSE-based methodology, Pachori and Gandhi showed the promise of advanced feature extraction techniques in neurological monitoring [7]. improving diagnostic and monitoring systems across various domains. However, anomaly detection in industrial sound poses several challenges. Industrial environments are characterized by their complexity and variability, factors such as background noise, varying operating conditions, and the presence of multiple types of machinery that contribute to the difficulty of identifying abnormal sound patterns [10], [11]. Furthermore, the high-dimensional and sequential nature of sound data further complicates the task, as traditional anomaly detection techniques may struggle to capture relevant features and temporal dependencies effectively [12], [13]. Motivated by these challenges andopportunities,thispaper presents a novel approach to anomaly detection in industrial sound analysis. Our goal is to develop a comprehensive framework capable of accurately identifying anomalies in sound signals, thereby facilitating proactive maintenance and fault diagnosis in industrial settings. Using state-of-the-art ML and deep learning (DL) techniques, we aim to overcome the limitations of existing methods and provide insights into effective strategies for anomaly detection. Our approach uses supervised ML and DL techniques, allowing us to exploit the strengths of both paradigms. Using three diverse datasets, the MIMII DG, the DCASE 2022, and the DCASE 2024, we evaluate the performance of various ML and DL models in different industrial contexts. Our study focuses on feature extraction methods such as Mel-Spectrogram and Mel-frequency cepstral coefficients (MFCCs), which are tailored to capture the relevant acoustic characteristics of industrial machinery. The primary contribution of this work lies in the development and evaluation of a comprehensive anomaly machinery, achieving high accuracy in fault diagnosis and condition monitoring. Similarly, research by [19], [20], [21], [22], and [23] utilized Decision Tree (DT) orForest modelsto classify abnormal sound patterns in industrial environments, enabling the early detection of equipment malfunctions and failures. Furthermore, the XGBoost model has also been applied to detect anomalies [22], [24], [25], [26], [27], [28]. In recent years, DL techniques, particularly Recurrent Neural Networks (RNNs), and their variants, have emerged as powerful tools for anomaly detection in sequential data, such as time-series signals and audio recordings. RNNs, equipped with memory cells that retain information over time, have shown promising results in capturing temporal dependencies and detecting subtle deviations from normal patterns. For instance, studies by [18], [29], [30], [31], [32], [33] applied Long Short-Term Memory (LSTM) networks to analyze acoustic signals for fault detection in industrial machinery, achieving superior performance compared to traditional ML methods. In addition, Gated Recurrent Units (GRUs),asimplifiedversion of LSTM,havebeenutilizedfor anomaly detection in audio

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

data [40], [42], [43], [44]. Researchers have extensively investigated various feature extraction methods to improve anomaly detection accu racy in industrial sound analysis. Among these methods, Mel-Spectrogram and Mel-frequency cepstral While existing studies have significantly contributed to the field, notable gaps and limitations in prior methods remain. Many approaches focus solely on either ML or DL techniques, overlooking the potential benefits of a hybrid approach. Additionally, the reliance on The additional information offered by other feature extraction techniques, such as MFCCs, is ignored when using Mel-Spectrogram features for sound representation. machinery. By conducting extensive evaluations on three distinct datasets, the MIMII DG [62], DCASE2022[63], and DCASE2024[64] datasets, we aim to provide a comprehensive analysis of anomaly detection techniques in real-world industrial environments, highlighting the advantages of our approach over existing methods. Our comparative analysis, as shown in Table 1, demonstrates how our proposed approach fills the gaps left by prior methods, offering superior performance and greater adaptability in detecting anomalous sound patterns across diverse industrial contexts. In contrast to previous studies, our work presents several novel contributions to the field of industrial sound anomaly detection. Firstly, while many existing approaches rely on traditional ML or DL techniques in isolation, The contributions of this paper include the following.

• A comprehensive evaluation of supervised ML and DL techniques for detecting anomalous sounds in industrial environments. The effectiveness of these models was assessed using three distinct industrial sound datasets, with evaluation metrics such as confusion matrix, accuracy, precision, recall, and F1 score.

• An investigation into feature extraction methods, focusing on a hybrid approach that combines high-frequency features, particularly Mel-Spectrogram and MFCCs, to improve anomaly detection accuracy.

• Experimental validation and performance analysis, demonstrating the effectiveness of the proposed approach for industrial anomaly detection applications. This paper addresses the pressing need for robust anomaly detection techniques in industrial sound analysis. By combining advanced ML and signal processing techniques, we aim to provide practitioners with powerful tools for proactive maintenance and fault diagnosis in industrial settings.

2. EXISTING SYSTEM OF PROJECT

- Gated Recurrent Units (GRU), which efficiently model sequential and temporal patterns in industrial machine sound data, are essential to the research. Because machine noises are dynamic and time-dependent, GRUs are ideal for processing acoustic inputs.
- Over time, they can record significant characteristics like anomalies or odd changes that might point to possible equipment malfunctions.
- To increase learning efficiency, GRUs employ gating mechanisms to preserve pertinent information and eliminate irrelevant input. They effectively handle long-range dependencies in sequential data without suffering significantly from the vanishing gradient issue.
- GRUs offer superior computing efficiency and robust predictive accuracy as compared to conventional recurrent neural networks (RNNs).
- The model learns typical machine sound patterns and recognises deviations that can indicate malfunctions or unusual operating circumstances. GRU-based analysis enables precise and timely defect diagnosis in industrial machines.
- Minimize downtime, and improve overall operational reliability in industrial environments.

3. RELATED WORK

Because early machine fault identification can lower maintenance costs, prevent equipment failures, and increase operational safety, anomaly detection in industrial machine noises has grown in importance. These methods learn patterns from labeled data and classify machine sounds as normal or abnormal. Several studies reported good performance using SVM and DT models for fault diagnosis and condition monitoring of industrial equipment. XGBoost has also shown strong classification performance due to its ensemble learning capability.

Recurrent Neural Networks (RNNs), Long Short-Term Memory (LSTM) networks, and Gated Recurrent Units (GRUs) have drawn a lot of attention for processing sequential audio data as deep learning has advanced. Feature extraction plays a critical role in industrial sound analysis. Mel-Spectrograms and Mel-Frequency Cepstral Coefficients (MFCCs) are among the most widely used audio features. Mel-Spectrograms provide a time-frequency representation of sound signals, while MFCCs capture important spectral characteristics that resemble human auditory perception. These features have been extensively used in machinery fault diagnosis, acoustic monitoring, and anomaly detection systems.

For industrial sound anomaly detection, recent research has investigated sophisticated deep learning architectures like Transformer-based models and MobileNet-based networks.

Recent studies have suggested combining several feature extraction techniques and assessing various learning

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

models to get beyond these restrictions. Research has demonstrated that GRU networks trained on MFCC features can preserve computational economy while achieving high anomaly detection accuracy. These methods facilitate predictive maintenance applications and enhance the identification of minute irregularities in machine sounds.

Therefore, the proposed project builds upon previous research by utilizing high-frequency audio features and GRU networks to develop an efficient and accurate industrial sound anomaly detection system. The combination of advanced feature extraction and deep learning techniques aims to enhance fault detection performance in real-world industrial environments.

The first module is a server for authenticators. Initially, the server needs to register and set their password. The authority server is the source of the key request. The authority server has a key generator. On the authority server, there is a request. Both owner and user requests from the database were to be included.

4. DATASET PREPARATION:

The MIMII DG and DCASE 2022 datasets are crucial components of this study. They provide diverse and com VOLUME 13, 2025 T.-T.-H. Le et al.: Anomaly Detection in Industrial Machine Sounds Using High-Frequency Features prehensive collections of sound recordings captured in real world industrial environments.

• MIMII DG dataset [62]. The MIMII DG dataset is a valuable resource for investigating malfunctioning industrial machines and conducting domain generalization tasks. It comprises normal and abnormal operating sounds from various industrial machines, including fans, gearboxes, bearings, slide rails, and valves. Each machine type is subdivided into three sections, introducing different domain shifts to simulate real-world scenarios. This dataset offers a diverse range of sound recordings capturing the acoustic signatures of machinery operating under different conditions, facilitating the study of anomaly detection algorithms.

• DCASE 2022 dataset [63]. In contrast, the DCASE 2022 dataset provides a broader spectrum of industrial sound environments, featuring recordings from manufacturing plants, transportation systems, construction sites, and other industrial settings. It includes seven types of real/toy machines: fans, gearboxes, bearings, slide rails, toy cars, toy trains, and valves. Each recording is a single-channel and 10-sec-long audio with a sampling rate of 16 kHz. The dataset consists of multiple sections for each machine type, offering a comprehensive training set and test data for evaluating anomaly detection algorithms. Each recording includes the machine's operating sound and environmental noise, presenting challenges typical of real-world industrial environments.

• DCASE 2024 dataset [64]. The dataset comprises normal and anomalous operational sounds from nine different types of real and toy machines. Each recording is a single-channel audio clip capturing both the machine's operating sound and background environmental noise. The duration of these recordings ranges between 6 and 10 seconds. The machines included in this dataset are: 3D Printer, Air Compressor, Brushless Motor, Hair Dryer, Hovering Drone, Robotic Arm, Scanner, Toothbrush, and Toy Circuit. Table 2 provides a detailed description of the training and testing data from three datasets, including the number of sound wave files, the presence of normal and anomalous sounds, and attributes related to domain shifts. These datasets serve as essential resources for benchmarking anomaly detection algorithms and advancing research in industrial sound analysis. Three datasets undergo preprocessing before model train ing to ensure data quality and consistency. This preprocessing step involves removing background noise, normalizing sound levels, and segmenting recordings into uniform time frames. By standardizing the duration of sound segments, the datasets facilitate feature extraction and model training, enabling a systematic and consistent analysis of sound data across different recordings and industrial contexts. Overall, this study's utilization of the MIMII DG dataset, the DCASE 2022 dataset, and the DCSE 2024 dataset enhances the robustness and generalizability of the proposed approach for anomalous sound detection in industrial environments. These datasets provide valuable insights into the acoustic characteristics of industrial machinery and enable comprehensive evaluations

5.TECHNIQUES USED IN PROJECT

In order to identify irregularities in industrial machine sounds, this study combines machine learning, deep learning, feature extraction, and audio signal processing. Preprocessing the audio recordings is the initial stage in order to eliminate noise, adjust sound levels, and get the data ready for analysis. In order to extract features and train models, this guarantees that the machine sound signals are clear and reliable.

Mel-Spectrogram and Mel-Frequency Cepstral Coefficients (MFCCs) are two key feature extraction techniques used to accurately represent the audio data. The Mel-Spectrogram allows the system to record changes in frequency over time by converting sound signals into a time-frequency representation. MFCCs are widely used in speech and sound recognition applications because they extract the most important spectral

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

characteristics of the audio signal. These characteristics aid in the model's ability to differentiate between typical and unusual machine operating circumstances. The project utilizes several machine learning algorithms, including Support Vector Machine (SVM), Decision Tree (DT), and Extreme Gradient Boosting (XGBoost). These algorithms learn patterns from labeled sound data and classify machine sounds as either normal or anomalous. Among these models, XGBoost demonstrates strong performance due to its ensemble learning capability and ability to handle complex patterns in the data.

The research uses Gated Recurrent Unit (GRU), Long Short-Term Memory (LSTM), and Recurrent Neural Network (RNN) networks for deep learning. These models are especially made to handle sequential data and record temporal relationships in audio signals. Because GRU requires less processing power than LSTM and efficiently learns long-term dependencies, it is chosen as the main model. The model is well suited for industrial sound anomaly detection because of its update and reset gate methods, which allow it to preserve crucial information while discarding unnecessary data. The entire system follows a supervised learning approach, where the models are trained using labeled datasets containing both normal and anomalous machine sounds. During training, the models learn to identify patterns associated with normal machine operation and detect deviations that may indicate faults or failures.

Lastly, common metrics like Accuracy, Precision, Recall, F1-Score, and Confusion Matrix are used to assess the models' performance. These assessment methods aid in determining how well the system detects unusual machine noises and reduces incorrect predictions.

In conclusion, the project builds an intelligent and dependable industrial sound anomaly detection system for predictive maintenance and fault diagnosis by integrating audio preprocessing, Mel-Spectrogram, MFCC, SVM, Decision Tree, XGBoost, RNN, LSTM, and GRU approaches.

6. MODEL TRAINING

Both machine learning (ML) and deep learning (DL) methods are used in this study for sound identification, each of which has special characteristics for identifying and evaluating the intricate patterns seen in industrial sound data. ML approaches use a variety of classifiers, such as Support Vector Machine (SVM), Decision Tree (DT), and XGBoost. Each sound recording is labelled as either a normal or anomalous event in the labelled data used to train these classifiers. The classifiers learn to recognise patterns and traits that differentiate between typical operational sounds and unusual sound occurrences during training. Using

The ML algorithms may efficiently classify fresh sound samples and identify anomalies in industrial equipment sound data using features extracted from the audio data, such as Mel-Spectrogram or MFCCs. • For classification problems, SVM (SupportVectorMachine) is a supervised learning method. Finding the ideal hyperplane to divide the data into distinct classes is how it operates [65]. SVM seeks to identify the hyperplane $wTx + b = 0$ that maximises the margin given a set of training data (X, y), where X represents the input features (such as Mel-Spectrogram or MFCCs) and y represents the corresponding labels (normal or anomaly) Features. between the classes, subject to the constraint $yi(wTxi + b) \geq 1$. The SVM model is trained using the input features extracted from the high-frequency data. During training, the SVM algorithm optimizes the hyperplane parameters w and b to achieve maximum class margin separation.

• DT (DecisionTree) is a tree-like model for classification tasks. It partitions the data recursively based on the feature values to create a tree structure that predicts the class labels [66]. DT recursively splits the feature space into regions to minimize each node's impurity. The splitting criterion, such as Gini impurity or entropy, determines the best feature and threshold for splitting. Let D be a dataset of n samples, each represented by a feature vector xi of dimension m and corresponding class labels yi. The goal of DT is to partition the feature space recursively into regions where each region corresponds to a decision node in the tree. DT selects the best feature j and threshold θ at each tree node to split the data into two subsets DL and DR. This splitting is determined based on a criterion such as Gini impurity or entropy, which measures the impurity of the data at each node. For a binary classification problem, let pjk denote the proportion of samples in subset Dj that belong to class k. Gini impurity is defined as: K Gini(D) = 1− k=1 Entropy impurity is defined as: K Entropy(D) = − (pjk)2 pjklog(pjk) k=1 (6) (7) DT selects the feature j and threshold θ that minimize the impurity after the split, i.e., Gain(D,j,θ) =Impurity(D) − |DL| |D| Impurity(DL)+|DR| |D| Impurity(DR) (8) where the feature j and threshold θ that maximizes the information gain are chosen for splitting. The tree building process continues recursively until a stopping criterion is met, such as reaching a maximum depth, minimum number of samples per leaf node, or mini mum impurity threshold. Once the tree is constructed, a prediction is made by traversing the tree based on the feature values of the input sample until a leaf node is reached. The majority class in the leaf node is then assigned as the predicted class label for the input sample. DT model is trained using the input features

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

extracted from the high-frequency data. During training, the Decision Tree algorithm recursively partitions the 77172-feature space based on the feature values to create a tree structure that predicts the class labels.

• XGBoost (Extreme Gradient Boosting) is an ensemble learning algorithm that combines the predictions of multiple weak learners (decision trees) to improve overall prediction accuracy [67]. XGBoost optimizes a differentiable loss function by iterative adding decision trees to the ensemble. Each tree is trained to correct the errors made by the previous trees. XGBoost aims to build an ensemble of weak learners (decision trees) and combine their predictions to form a strong predic tive model. XGBoost optimizes an objective function consisting of the loss function and a regularization term. The loss function measures the discrepancy between the predicted scores (ˆy) and the true labels (y). For binary classification, a common choice is the logistic loss: n loss(y, ˆy) = i=1 yilog(1+e−ˆyi)+(1 − yi)log(1+eˆyi) (9) The regularization term penalizes the complexity of the model to prevent overfitting. Common choices include L1 and L2 regularization. XGBoost employs a gradient-boosting algorithm to add decision trees to the ensemble iteratively. At each iteration, a new decision tree fits the negative gradient of the loss function concerning the predictions of the current ensemble. Each decision tree is grown using a greedy algorithm that recursively splits the data into regions based on the feature values. The splitting criterion is determined by optimizing the objective function, considering both the improvement in loss and the regularization term. XGBoost applies regularization techniques such as tree pruning to control the complexity of individual trees and prevent overfitting. Trees are pruned based on their depth or the number of leaf nodes. The predictions of individual trees are combined using a weighted sum to obtain the final ensemble prediction. The weights of the trees are determined during the training process based on their contribution to reducing the loss function. The XGBoost model is trained using the input features extracted from the high-frequency data. During training, XGBoost iteratively adds decision trees to the ensemble, optimizing the loss function to improve prediction accuracy. In contrast, the DL approach focuses on leveraging the power of recurrent neural network (RNN) architectures to capture temporal dependencies in the sound data. Specifically, LSTM and Gated Recurrent Unit (GRU) networks are employed because they can model sequential data and capture long-term dependencies effectively. These RNN architectures are trained on sequential sound data, where each input represents a time series of sound features extracted from consecutive time frames. By learning the temporal dynamics of sound patterns, LSTM and GRU networks can detect anomalies in time-series data, making them well-suited for analyzing industrial sound recordings where anomalies may manifest over time.

• RNN(Recurrent Neural Network): RNNs are designed to process sequential data by maintaining an internal state or memory that captures information from previous time steps [68]. Given an input sequence x = (x1, x2, ...,xT), where xt represents the input at time step t, and an initial hidden state h0, the hidden state ht at time step t is computed as: ht = f(xt,ht−1), (10) where f is a non-linear activation function (typically a sigmoid or hyperbolic tangent function) that combines the input xt and the previous hidden state ht−1 to compute the current hidden state ht. The output at each time step can be computed using the hidden state ht and a weight matrix W: yt = softmax(Wht) (11) RNNs suffer from the vanishing gradient problem, where gradients diminish exponentially over long sequences, making it difficult to learn long-range dependencies.

• LSTM (Long Short-Term Memory): LSTM networks address the vanishing gradient problem by introducing a gating mechanism that regulates the flow of information through the network [69]. At each time step, an LSTM unit maintains a cell state Ct and three gates: input gate it , forget gate ft, and output gate ot. The input gate controls the flow of new information into the cell state, while the forget gate determines which information to discard from the cell state, and the output gate regulates the information to be output from the cell state. The computations within an LSTM unit can be expressed as follows: it = σ(Wxixt +Whiht−1 +WciCt−1 +bi) (12) ft = σ(Wxfxt +Whfht−1 +WcfCt−1 +bf) (13) ot = σ(Wxoxt +Whoht−1 +WcoCt−1 +bo) (14) gt = tanh(Wxgxt + Whght−1 +bg) (15) Ct = ft ⊙Ct−1 ++it ⊙gt ht = ot ⊙ tanh(Ct) (16) (17) where σ represents the sigmoid activation function, ⊙ denotes element-wise multiplication, and W and b are weight matrices and bias vectors, respectively.

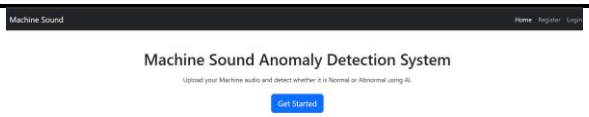
• GRU (Gated Recurrent Unit): GRU is a variation of LSTM that simplifies the architecture by combining the forget and input gates into a single update gate and merging the cell state and hidden state [70]. Similar to LSTM, GRUalso maintains a hidden state ht and a cell state Ct, but it computes these states differently: zt = σ(Wxzxt +Whzht−1 +bz) rt = σ(Wxrxt +Whrht−1 +br) (18) (19) TABLE 3. General architecture of RNN, LSTM, or GRU model in our approach. ˆ ht = tanh(Wxhxt + Whh(rt ⊙ ht−1) + bh) (20)

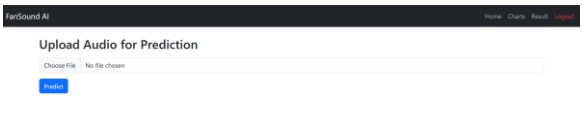
Our approach employs a sophisticated architecture comprising RNN, LSTM, and GRU models, as depicted in Table 3. By leveraging this architecture, our approach aims to harness the strengths of RNN, LSTM, and GRU

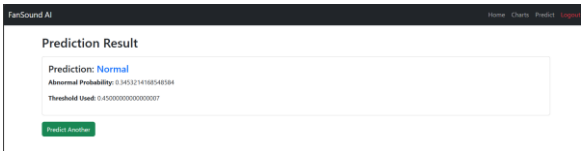
International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

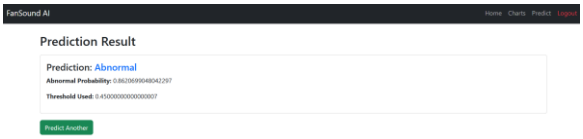
models for anomaly detection tasks, ensuring robust performance across diverse industrial sound datasets. This architecture is meticulously crafted to effectively capture temporal dependencies and subtle deviations in the input data. The input layer accommodates various sizes depending on the feature input: for instance, Mel Spectrogram with an input shape of 128 × 313, MFCCs with an input shape of 12 × 313, or a combination of both features with an input shape of 128 × 313. The Bidirectional RNN/LSTM/GRU layers as Feature Extractor, sized at 32, play a pivotal role in processing the input data bidirectionally, enhancing the model's capacity to assimilate information from past and future contexts. Dropout layers with a dropout rate of 0.1 are strategically placed after each Feature Extractor layer to mitigate overfitting by randomly dropping out a fraction of input units during training. Subsequent Feature Extractor layers refine the input data representation, augmenting the model's ability to discern intricate patterns. To ensure regularization, an additional set of Dropout layers with a slightly elevated dropout rate of 0.2 follows these layers. The Dense layers, each configured with a size of 64, serve as the neural network's backbone, transforming the input data nonlinearly to extract higher-level features. Dropout layers with a dropout rate of 0.1 are interspersed between the Dense layers to prevent the model from excessively relying on specific features and to enhance generalization. Ultimately, the Dense Output layer, whose size varies based on the number of classes in the classification task, produces the model's final output. This work has five and seven Dense outputs corresponding to MIMII DG and DCASE 2022 datasets. This layer applies an appropriate activation function, such as softmax for multi class classification, to generate class probabilities. ML and DL models are optimized using labeled training data to minimize classification errors and improve overall accuracy during training. The models undergo iterative training, where parameters are adjusted based on the observed performance of the training data. Additionally, cross validation and hyperparameter tuning may be employed to optimize the models' performance further and prevent overfitting to the training data. Overall, using ML and DL algorithms for sound detection enables a comprehensive approach to anomaly detection in industrial environments. By combining the strengths of traditional ML classifiers with the advanced capabilities of DL architectures, the proposed methodology offers a robust and effective solution for detecting anomalous sound events and ensuring operational safety and efficiency in industry.

7. RESULTS









The experimental design encompasses several key steps to ensure robustness and reproducibility when evaluating the proposed anomaly detection methods. Firstly, data preprocessing is conducted on industrial sound datasets to prepare them for model training and evaluation. This involves labeling the output using a LabelEncode to convert categorical labels into numerical representations, facilitating model interpretation and analysis. The LabelEncoder is a preprocessing step that converts categorical labels into numerical representations, facilitating model training and evaluation. Mathematically,theLabelEncoderassignsauniqueintegert oeachdistinct category present in the label data. Let y denote the original categorical labels, and let LE(y) represent the numerical encoding obtained after applying the LabelEncoder. The encoding process can be described as follows: Given a set of N unique categories in the label data, indexed from 0 to N-1, the LabelEncoder assigns an integer label i to each category c such that 0 ≤ i < N. □ □ □ □ □ □ LE(y) = □ □ □ □ □ 0 1 . . . if yi = category1 if yi = category2 N −1 if yi = categoryn, (26) . . . where yi represents the ith element of the original categorical label vector y. After applying the LabelEncoder, the original categorical labels are replaced with their corresponding numerical representations, enabling these labels in ML algorithms that require numerical inputs. This preprocessing step ensures compatibility between the data and the models, facilitating accurate and efficient model training and evaluation. Next, the preprocessed data is split into

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

training and validating sets using an 80:20 ratio. This partitioning ensures that sufficient data is available for model training while preserving a separate portion for evaluation to assess generalization performance accurately. In terms of implementation, the proposed methods are developed using the Python programming language and relevant libraries such as TensorFlow 2.10.0, scikit-learn 1.3.2, and librosa 0.10.1 for sound processing and ML tasks. These libraries offer extensive data manipulation, feature extraction, model training, and evaluation functionalities, streamlining the implementation process and enabling efficient experimentation. Additionally, hardware requirements are considered to ensure the smooth execution of the experimental procedures. While specific hardware specifications may vary depending on the dataset size and computational complexity of the models, a standard computing environment with sufficient processing power and memory capacity is typically adequate for experimenting: Windows 10, CPU: i9 10900K, RAM: 64 GB, GPU: RTX3080 Ti. Following these steps in the experimental design, we aim to establish a systematic and rigorous framework for evaluating the proposed anomaly detection methods, enabling reliable comparisons and insights into their performance across different datasets and experimental conditions.

B. HYPER-PARAMETERS TUNING

For the feature extraction method, we carefully selected and tuned parameters for each method to optimize the extraction of Mel-Spectrogram and MFCCs features, as detailed in Table 4. The parameters of the Mel-Spectrogram extraction method encompass several key aspects crucial for accurate feature representation. The number of FFT windows (n_fft) determines the frequency resolution, influencing the level of detail captured in the spectrogram. Sampling rate (sr) defines the frequency range analyzed, ensuring that relevant spectral components are captured within the spectrogram. Hop length (hop_length) regulates the overlap between successive frames, balancing temporal resolution with com putational efficiency. Window function (window) selection influences spectral leakage and sidelobe levels, impacting the accuracy of frequency representation. Frame centering (center)determines whether frames are centered, affecting temporal alignment within the spectrogram. Paddingmode(pad_mode) addresses edge -effects during analysis, enhancing the robustness of spectrogram computation. Finally, the magnitude exponent (power) adjusts the dynamic range of the spectrogram, optimizing the representation of signal magnitudes in the frequency domain. Collectively, these parameters allow fine-tuning of the Mel-Spectrogram extraction process to capture relevant spectral characteristics accurately. The parameters involved in extracting MFCCs play a crucial role in effectively capturing the spectral character istics of audio signals. The sampling rate (sr) determines the frequency range analyzed, ensuring that the appropriate spectral components are represented in the coefficients. The number of MFCCs to return (n_mfcc) controls the dimensionality of the feature space, balancing between capturing relevant spectral information and computational efficiency. The choice of discrete cosine transform (DCT) type (dct_type) influences the coefficients' decorrelation, affecting the feature set's discrimination power. Normalization (norm) adjusts the magnitude of the coefficients to ensure consistency across different signals, enhancing the robustness of the feature extraction process. Additionally, the lifter parameter determines the emphasis on higher-order coefficients, impacting the emphasis on specific spectral components in the feature representation. Together, these parameters allow for the customization of the MFCC 77175 extraction process to capture relevant spectral features accurately for subsequent analysis and classification tasks. The hyperparameters for ML and DL models are carefully tuned to optimize their performance in anomaly detection tasks, as shown in Table 5. For ML models like SVM and DT, parameters such as the kernel type, regularization parameter, and criteria for split quality are adjusted to enhance model accuracy. In SVM, the choice of kernel type, regularization parameter (C), and kernel coefficient (gamma) are crucial for defining decision boundaries and controlling overfitting. Similarly, the split criteria, minimum samples per split, and minimum impurity decrease determine DT's tree structure and decision-making process. XGBoost, a gradient boosting algorithm, is tuned by adjusting parameters like the number of estimators, maximum depth, and learning rate to optimize ensemble learning and improve predictive accuracy. For DL models like RNN, LSTM, and GRU, hyperparameters such as learning rate, batch size, dropout rate, optimizer, number of epochs, activation function, and loss function are fine-tuned to improve model convergence and prevent overfitting. The learning rate controls the step size during optimization, while the batch size determines the 77176 number of samples processed in each iteration. Dropout regularization is applied to prevent co-adaptation of feature detectors, enhancing model generalization. The choice of optimizers, such as Adam, influences the update strategy for model parameters. The number of epochs defines the number of iterations over the entire dataset during training, while the activation function and loss function determine the output format and optimization objective, respectively. By systematically

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

adjusting these hyperparameters, the ML and DL models can effectively learn from the data and achieve high performance in anomaly detection tasks

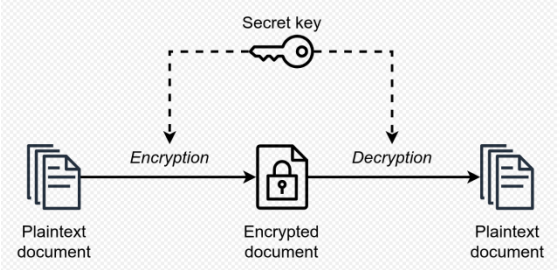


Figure 6.3.2 Symmetric key algorithms

8. DISCUSSION AND CONCLUSION

We have shown a conceptually simple method of concealing information inside an existing sequence of strings, allowing for deceptive decryption in case of forced revelation of decryption keys. At a practical level, matters of storing or remembering the decryption keys have not been discussed, but the use of passwords, for example, is not difficult to imagine here: suppose that whenever we require random values to be chosen, we do so by invoking a pseudorandom number generator (PRNG, e.g., the standardized password-based key derivation function Argon2) that is seeded with a password that the user chooses. In that way, the storage of the value-pair list that constitutes the secret key ski for the message mi, boils down to the choice of a password to open the message mi, from which all random quantities in the process can be recomputed with the password as a seed for a PRNG. The implications to security are, in that sense, to be considered carefully, as the overall entropy about the secret reduces to the min-entropy of the password choice process that determines the hardness of guessing the password, which is the Shannon entropy. Further generalizations may be the inclusion of a third party to establish a four-eyes principle in the opening of a message. That is, for example, one could substitute ρi,1, . . . , ρi,ni by products ρ (a) i,1 • ρ (b) i,1 . . . , ρ (a) i,ni • ρ (b) i,ni , with the individual factors coming from key-bases, respectively root keys, that two persons, Alice and Bob, are given. In that case, an adversary forcing Alice to cooperate would also have to convince Bob to cooperate, in order to discover a meaningful message. As yet another variant, note that the role of the factors from root key r and from c is ‘‘symmetric’’, and hence one could alternate the appending of parts to c with adding parts to the r. Storing c in a remote location and keeping r on one’s own local computer then creates the seeming appeal of putting new information into c without actually letting c visibly grow. This instance of the scheme is, however, not considered as useful here, since it is nothing else than storing an encrypted version of a message locally, and letting the key to this message be stored remotely at a possibly untrusted location. Practical room for improvement is in the scheme’s necessity to remember and use all secret keys whenever there is a need to modify messages after they went into the ciphertextc. Abandoning this requirement, the key storage and management requirements fall back to those of a conventional secret key encryption with fixed key sizes. Hence, its ‘‘information theoretic’’ security guarantees are in any case bounded by the size of the secret root key to be guessed, but the brute force complexity is still larger than just guessing the secret key, since the adversary may, except if there is so far only 1 message in c, still be uncertain about which parts of c may have been used to represent the secret message. Therefore, the concept of (plausible) deniability is the added value over brute-force attack resilience. Based on our review of literature on deniable encryption schemes, the user can, in past schemes, come up with only one fake message as a counterpart to defend its secret. In contrast, our scheme allows us to bring more than one fake message to hide a secret. Also, this encryption scheme is editable in the sense that at any instant in time, we can update our message by changing only one element in the cipher text or by changing the key indices. Furthermore, the scheme gives us the freedom to delete any message encrypted inside the cipher text simply by replacing at least one element from the cipher text with a random number. Deleting a message gives us the flexibility to prevent the user who was earlier accessing that message from doing it again. If the user wants to decrypt the cipher text with the older key, the outcome will undoubtedly dissatisfy him. Consequently, False-Bottom Encryption extends deni able encryption by the functionality of adding, editing and deleting possibly several plaintexts inside the ciphertext. Our security definition does not account for adversaries profiling the access patterns of a user, which calls for addi tional techniques to either randomize or ‘‘equalize’’ all access sequences. Future work will thus investigate extensions to our scheme by means of private information retrieval or other techniques (see the related work, in particular [22]), to analyze if information-theoretic security remains accomplishable or deteriorates against attackers that profile the (physical) device usage.

9. FUTURE ENHANCEMENT

The experimental findings show how well the suggested method works for identifying unusual sounds in industrial settings. Two sets of results are presented in this section: multi-class classification is examined in the second set, and sound anomaly detection is the subject of

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

the first. Furthermore, Table 6: Anomaly Detection in Industrial Machine Sounds Using High-Frequency Features. The suggested method's detection outcome on the MIMII DG 2022 dataset. method is compared with prior approaches. The section concludes by highlighting the benefits and drawbacks of our approach.

REFERENCES

[1] R. Canetti, U. Feige, O. Goldreich, and M. Naor, ''Adaptively secure multi-party computation,'' in *Proc. 28th Annu. ACM Symp. Theory Com- put.*, Philadelphia, PA, USA, 1996, pp. 639–648. [Online]. Available: http://portal.acm.org/citation.cfm?doid=237814.238015

[2] R. Canetti, C. Dwork, M. Naor, and R. Ostrovsky, ''Deniable encryption,'' in *Proc. 17th Annu. Int. Cryptol. Conf.* (Lecture Notes in Computer Sci- ence), vol. 1294. Santa Barbara, CA, USA: Springer, 1997, pp. 90–104.

[3] A. Juels and T. Ristenpart, ''Honey encryption: Security beyond the brute-force bound,'' in *Advances in Cryptology—EUROCRYPT* (Lec- ture Notes in Computer Science), vol. 8441, D. Hutchison, T. Kanade,

[4] J. Kittler, J. M. Kleinberg, A. Kobsa, F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, D. Terzopoulos, D. Tygar, G. Weikum, P. Q. Nguyen, and E. Oswald, Eds. Berlin, Germany: Springer, 2014, pp. 293–310, doi: 10.1007/978-3-642-55220-5_17.

[5] M. Durmuth and D. M. Freeman, ''Deniable encryption with negli- gible detection probability: An interactive construction,'' in *Advances in Cryptology—EUROCRYPT* (Lecture Notes in Computer Science), vol. 6632, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg, F. Mattern,

[6] J. C. Mitchell, M. Naor, O. Nierstrasz, C. Pandu Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, and K. G. Paterson, Eds. Berlin, Germany: Springer, 2011, pp. 610–626, doi: 10.1007/978-3-642-20465-4_33.

[7] A. O'Neill, C. Peikert, and B. Waters, ''Bi-deniable public-key encryption,'' in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 6841, D. Hutchison, T. Kanade, J. Kittler, J. M. Kleinberg,

[8] F. Mattern, J. C. Mitchell, M. Naor, O. Nierstrasz, C. P. Rangan, B. Steffen, M. Sudan, D. Terzopoulos, D. Tygar, M. Y. Vardi, G. Weikum, and P. Rogaway, Eds. Berlin, Germany: Springer, 2011, pp. 525–542, doi: 10.1007/978-3-642-22792-9_30.

[9] M. Klonowski, P. Kubiak, and M. Kutylowski, ''Practical deniable encryp- tion,'' in *SOFSEM 2008: Theory and Practice of Computer Science* (Lecture Notes in Computer Science), V. Geffert, J. Karhumäki, A. Bertoni,B. Preneel, P. Návrat, and M. Bieliková, Eds. Berlin, Germany: Springer, 2008, pp. 599–609.

[10]P. Gasti, G. Ateniese, and M. Blanton, ''Deniable cloud storage: sharing files via public-key deniability,'' in *Proc. 9th Annu. ACM Workshop Pri- vacy Electron. Soc.*, Chicago, IL, USA, 2010, p. 31. [Online]. Available: http://portal.acm.org/citation.cfm?doid=1866919.1866925

[11] M. H. Ibrahim, ''A method for obtaining deniable public-key encryption,'' *Int. J. Netw. Secur.*, vol. 8, no. 1, pp. 1–9, 2009.

[12] P. Chi and C. Lei, ''Audit-free cloud storage via deniable attribute- based encryption,'' *IEEE Trans. Cloud Comput.*, vol. 6, no. 2, pp. 414–427, Apr. 2018. [Online]. Available: https://ieeexplore.ieee.org/document/7090980/

[13] R. Canetti, S. Park, and O. Poburinnaya, ''Fully deniable interactive encryption,'' in *Advances in Cryptology—CRYPTO* (Lecture Notes in Computer Science), vol. 12170, D. Micciancio and T. Ristenpart, Eds. Cham, Switzerland: Springer, 2020, pp. 807–835, doi: 10.1007/978-3-030- 56784-2_27.

[14] C. Dwork, M. Naor, and A. Sahai, ''Concurrent zero-knowledge,'' *J. ACM*, vol. 51, no. 6, p. 851–898, Nov. 2004, doi: 10.1145/1039488.1039489.

[15] M. Naor, ''Deniable ring authentication,'' in *Advances in Cryptology— CRYPTO* (Lecture Notes in Computer Science), vol. 2442, G. Goos,

[16] J. Hartmanis, J. van Leeuwen, and M. Yung, Eds. Berlin, Germany: Springer, 2002, pp. 481–498, doi: 10.1007/3-540-45708-9_31.

[17] R. W. Zhu, D. S. Wong, and C. H. Lee, ''Cryptanalysis of a suite of deniable authentication protocols,'' *IEEE Commun. Lett.*, vol. 10, no. 6, pp. 504–506, Jun. 2006.

[18] A. Fiat and M. Naor, ''Broadcast encryption,'' in *Proc. Annu. Int. Cryptol. Conf.*, Jan. 1993, pp. 480–491.

[19] J. Li, Y. Wang, Y. Zhang, and J. Han, ''Full verifiability for out- sourced decryption in attribute based encryption,'' *IEEE Trans. Services Comput.*, vol. 13, no. 3, pp. 478–487, May 2020. [Online]. Available: https://ieeexplore.ieee.org/document/7936626/

[20] J. Li, Q. Yu, and Y. Zhang, ''Hierarchical attribute based encryption with continuous leakage-resilience,'' *Inf. Sci.*, vol. 484, pp. 113–134, May 2019. [Online]. Available: https://linkinghub.elsevier.com/retrieve/pii/S0020025519300684

[21] J. Li, W. Yao, Y. Zhang, H. Qian, and J. Han, ''Flexible and fine-grained attribute-based data storage in cloud computing,'' *IEEE Trans. Services Comput.*, vol. 10, no. 5, pp. 785–796, Sep. 2017. [Online]. Available: http://ieeexplore.ieee.org/document/7390098/

[22] S. Reddy, P. S. Reddy, and P. Sravanthi, ''Audit free cloud stor- age via deniable attribute base encryption for

International Journal of
DATA SCIENCE AND IOT MANAGEMENT SYSTEM
Peer Reviewed, Referred & Indexed Journal
ISSN: 3068-272X www.ijdim.com Original Research Paper

protecting user pri- vacy,'' *Int. J. Sci. Eng. Technol. Res.*, vol. 5, no. 17, pp. 3449–3451, 2016.

[23] R. Bassous, R. Bassous, H. Fu, and Y. Zhu, ''Ambiguous multi-symmetric cryptography,'' in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2015, pp. 7394–7399.

[24] A. Chakraborti, C. Chen, and R. Sion, ''POSTER: DataLair: A storage block device with plausible deniability,'' in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.*, Oct. 2016, pp. 1757–1759. [Online]. Available: https://dl.acm.org/doi/10.1145/2976749.2989061

[25] C. Chen, A. Chakraborti, and R. Sion, ''PD-DM: An efficient locality-preserving block device mapper with plausible deniability,'' *Proc. Privacy Enhancing Technol.*, vol. 2019, no. 1, pp. 153–171, Jan. 2019. [Online]. Available: https://petsymposium.org/popets/2019/ popets-2019-0009.php

[26] C. Chen, X. Liang, B. Carbunar, and R. Sion, ''SoK: Plausibly deni- able storage,'' *Proc. Privacy Enhancing Technol.*, vol. 2022, no. 2, pp. 132–151, Apr. 2022. [Online]. Available: https://petsymposium.org/popets/2022/popets-2022-0039.php

[27] E.-O. Blass, T. Mayberry, G. Noubir, and K. Onarlioglu, ''Toward robust hidden volumes using write-only oblivious RAM,'' in *Proc. ACM SIGSAC Conf. Comput. Commun. Secur.* New York, NY, USA: Association for Computing Machinery, Nov. 2014, pp. 203–214, doi: 10.1145/2660267.2660313.

[28] C. Hargreaves and H. Chivers, ''Detecting hidden encrypted vol- umes,'' in *Communications and Multimedia Security*, B. De Decker and I. Schaumuller-Bichl, Eds. Berlin, Germany: Springer, 2010, pp. 233–244.